

AFRL-RI-RS-TR-2023-120

THE TESTING, EVALUATION, AND CONTROL OF HETEROGENEOUS LARGE SCALE SYSTEMS OF AUTONOMOUS VEHICLES (TECHLAV) CENTER

NORTH CAROLINA AGRICULTURAL &
TECHNICAL STATE UNIVERSITY

JUNE 2023

FINAL TECHNICAL REPORT

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED

STINFO COPY

AIR FORCE RESEARCH LABORATORY
INFORMATION DIRECTORATE

■ **AIR FORCE MATERIEL COMMAND** ■ **UNITED STATES AIR FORCE** ■ **ROME, NY 13441**

NOTICE AND SIGNATURE PAGE

Using Government drawings, specifications, or other data included in this document for any purpose other than Government procurement does not in any way obligate the U.S. Government. The fact that the Government formulated or supplied the drawings, specifications, or other data does not license the holder or any other person or corporation; or convey any rights or permission to manufacture, use, or sell any patented invention that may relate to them.

This report is the result of contracted fundamental research deemed exempt from public affairs security and policy review in accordance with SAF/AQR memorandum dated 10 Dec 08 and AFRL/CA policy clarification memorandum dated 16 Jan 09. This report is available to the general public, including foreign nations. Copies may be obtained from the Defense Technical Information Center (DTIC) (http://www.dtic.mil).

AFRL-RI-RS-TR-2023-120 HAS BEEN REVIEWED AND IS APPROVED FOR PUBLICATION IN ACCORDANCE WITH ASSIGNED DISTRIBUTION STATEMENT.

FOR THE CHIEF ENGINEER:

/ S /
JESSICA DORISMOND
Work Unit Manager

/ S /
JULIE BRICHACEK
Chief, Information Systems Division
Information Directorate

This report is published in the interest of scientific and technical information exchange, and its publication does not constitute the Government's approval or disapproval of its ideas or findings.

REPORT DOCUMENTATION PAGE

1. REPORT DATE	2. REPORT TYPE	3. DATES COVERED	
		START DATE	END DATE
JUNE 2023	FINAL TECHNICAL REPORT	APRIL 2015	DECEMBER 2022

4. TITLE AND SUBTITLE

THE TESTING, EVALUATION, AND CONTROL OF HETEROGENEOUS LARGE SCALE SYSTEMS OF AUTONOMOUS VEHICLES (TECHLAV) CENTER

5a. CONTRACT NUMBER	5b. GRANT NUMBER	5c. PROGRAM ELEMENT NUMBER
FA8750-15-2-0116	N/A	61228D
5d. PROJECT NUMBER	**5e. TASK NUMBER**	**5f. WORK UNIT NUMBER**
		R1NZ

6. AUTHOR(S)

Dr. Abdollah Homaifar

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

North Carolina Agricultural and Technical State University
1601 East Market Street
Greensboro NC 27411-0002

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)	11. SPONSOR/MONITOR'S REPORT NUMBER(S)
Air Force Research Laboratory/RISA 525 Brooks Road Rome NY 13441-4505	AFRL/RI	AFRL-RI-RS-TR-2023-120

12. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for Public Release; Distribution Unlimited. This report is the result of contracted fundamental research deemed exempt from public affairs security and policy review in accordance with SAF/AQR memorandum dated 10 Dec 08 and AFRL/CA policy clarification memorandum dated 16 Jan 09.

13. SUPPLEMENTARY NOTES

14. ABSTRACT

The Testing, Evaluation, and Control of Heterogeneous Large Scale Systems of Autonomous Vehicles (TECHLAV) Center, led by North Carolina Agricultural and Technical State University (N.C. A&T), in collaboration with partners at the University of Texas at San Antonio (UTSA) and the Southwestern Indian Polytechnic Institute (SIPI), are conducting research activities to address two fundamental grand challenges:

1. Teaming and Cooperative Control of Large Scale Autonomous Systems of Vehicles (LSASVs) integrated with human operators

2. Testing, Evaluation, Validation, and Verification of LSASVs These challenges are met through three interleaved research thrusts:

• Thrust 1: Modeling, Analysis and Control of LSASVs

• Thrust 2: Resilient Control and Communication for LSASVs

• Thrust 3: Testing, Evaluations and Verification of Large-scale Autonomous Vehicles (TEVLAV)

15. SUBJECT TERMS

TECHLAV, LSASVs, TEVLAV

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES
a. REPORT	b. ABSTRACT	C. THIS PAGE		
U	U	U	SAR	443

19a. NAME OF RESPONSIBLE PERSON	19b. PHONE NUMBER *(Include area code)*
JESSICA DORISMOND	N/A

PREVIOUS EDITION IS OBSOLETE.

STANDARD FORM 298 (REV. 5/2020)
Prescribed by ANSI Std. Z39.18

Table of Contents

1 Summary and Introduction....1
Educational Activities: 10
Outreach Activities: 10
2 Project Information....11
2.1 Administrative Information 11
2.2.1 Project Description 12
2.2.1.1 Research Objectives 12
2.2.1.2 Public Problem Description 12
2.2.2 Methods 14
2.2.2.2 Comparison with Current Technology/Assumptions and Procedures 15
2.2.3 Schedule and Milestones....16
2.2.3.2 Detailed Description of the Tasks in Thrust 1....16
2.2.3.3 Detailed Description of the Tasks in Thrust 2....19
2.2.3.4 Detailed Description of the Tasks in Thrust 3....22
2.2.4 Deliverables Description 25
2.2.5 Technology Transition and Technology Transfer Targets and Plans....26
3 Results and Discussion....27
3.1 Project Progress and Project Plans for Thrust 1: Modeling, Analysis and Control of Large-scale Autonomous Vehicles (MACLAV) 27
3.1.1 Project Progress for Task T1-1 (Mathematical Modeling of LSASVs)....27
3.1.1.2 General Description of Task T1-1....27
3.1.1.3 Progress Against Planned Objectives in Task T1-1 27
3.1.1.4 Modeling Testbed Systems 28
3.1.1.5 Summary....36
3.1.2 Project Progress for Task T1-2 (Data-and Knowledge-Based Modeling of LSASVs) 38
3.1.2.2 General Description of Task T1-2....38
3.1.2.3 Progress Against Planned Objectives in Task T1-2 39
3.1.2.4 Technical Accomplishments in Task T1-2....39
3.1.3 Project Progress for Task T1-3 (Qualitative and Quantitative Analysis of LSASVs) 41
3.1.3.2 General Description of Task T1-3....42
3.1.3.3 Progress Against Planned Objectives in Task T1-3 42
Software Implementation of Position Estimation of UGV 51
2.1.3.4 Summary....66
3.1.4 Project Progress for Task T1-4 (Cooperative Localization and Navigation for LSASV) 67
3.1.4.2 General Description of Task T1-4....67
3.1.4.3 Technical Accomplishments in Task T1-4....68
Investigation Into Real-Time Appearance-Based Mapping....68
3.1.5 Project Progress for Task T1-5 (Hierarchical Hybrid Cooperative Control of LSASV) 100
3.1.5.2 General Description of Task T1-5....100
3.1.5.3 Objectives for Task T1-5 over the Course of Project....101
3.1.5.4 Progress Against Planned Objectives in Task T1-5 102
3.1.5.5 Technical Accomplishments in Task T1-5....102
Hybrid Controller Design:....110

3.1.6 Project Progress for Task T1-6 (Cloud-Based Control of LSASV) 112
3.1.6.2 General Description of Task T1-6 112
3.1.6.3 Progress Against Planned Objectives in Task T1-6 112
3.1.6.4 Technical Accomplishments in Task T1-6 113
3.1.6.5 Summary 145
3.2 Project Progress and Project Plans for Thrust 2: Resilient Control and Communication of Large-scale Autonomous Vehicles (RC2LAV) 145
3.2.1 Project Progress for Task T2-1 (Developing a decentralized fault detection mechanism) 145
3.2.1.2 General Description of Task T2-1 145
3.2.1.3 Progress Against Planned Objectives in Task T2-1 136
3.2.1.4 Technical Accomplishments in Task T2-1 136
3.2.2 Project Progress for Task T2-2 (Fault Diagnostics and Prognostics through Data Analytic Approaches) 139
3.2.2.2 General Description of Task T2-2 139
3.2.2.3 Objectives for Task T2-2 over the Course of Project 139
3.2.2.4 Progress Against Planned Objectives in Task T2-2 140
3.2.2.5 Technical Accomplishments in Task T2-2 140
3.2.3 Project Progress for Task T2-3 (Developing a decentralized fault accommodation mechanism) 141
3.2.4 Project Progress for Task T2-4 (Internet of Things (IoT) 5G Networks with Game Theory Applied to Resource Allocation and IoT Security) 142
3.2.4.2 Progress Against Planned Objectives in Task T2-4) 142
3.2.4.3 Technical Accomplishments in Task T2-4 142
3.2.5 Project Progress for Task T2-5 (Enhancing network performance utilizing the cognitive features of the agents) 162
3.2.5.2 General Description of Task T2-5 162
3.2.5.3 Objectives for Task T2-5 over the Course of Project 162
3.2.5.4 Progress Against Planned Objectives in Task T2-5 163
3.2.5.5 Technical Accomplishments in Task T2-5 163
3.2.6 Project Progress for Task T2-6 (Delay-tolerant and loss-tolerant consensus in networks of agents) 164
3.2.6.2 General Description of Task T2-6 165
3.2.6.3 Technical Accomplishments in Task T2-6 165
3.2.7 Project Progress for Task T2-7 (Wireless Scale Free Topologies for Resiliency and Jamming Immunity) 168
3.2.7.2 General Description of Task T2-7 168
3.2.7.3 Objectives for Task T2-7 over the Course of Project 169
3.2.7.4 Progress Against Planned Objectives in Task T2-7 169
3.2.7.4.14 Appendix: K-User MIMO Interference Alignment Mathematics for K=3 179
3.2.7.4.19 Proof of Theorem 1 182
3.2.7.4.25 System Model 188
3.2.7.5 Review of the 5G Extended MOPRO Scheme 190
Step 1: Alice to Bob 190
Step 2: Bob to Alice 190
Step 3: Alice to Bob 191
3.2.7.6 Simulation procedure and results 193
3.2.7.7 Technical Accomplishments in Task T2-7 194
3.3 Project Progress and Project Plans for Thrust 3: Testing, Evaluations and Verification of Large-scale Autonomous Vehicles (TEVLAV) 196

3.3.1 Project Progress for Task T3-1 (Motion planning and control framework for coordinated collective motion of robotic swarms) 196
3.3.1.2 General Description of Task T3-1 196
3.3.1.3 Objectives for Task T3-1 over the Course of Project 198
3.3.1.4 Progress Against Planned Objectives in Task T3-1 198
3.3.1.5 Technical Accomplishments in Task T3-1 199
3.3.2 Project Progress for Task T3-2 (Developing a compositional verification approach for LSASV) 211
3.3.2.2 General Description of Task T3-2 211
3.3.2.3 Objectives for Task T3-2 over the Course of Project 212
3.3.2.4 Progress Against Planned Objectives in Task T3-2 212
3.3.2.5 Technical Accomplishments in Task T3-2 212
3.3.3 Project Progress for Task T3-3 (Developing Formal Methods for Human-robot Collaboration) 220
3.3.3.2 General Description of Task T3-3 220
3.3.3.3 Objectives for Task T3-3 over the Course of Project 221
3.3.3.4 Progress Against Planned Objectives in Task T3-3 221
3.3.3.5 Technical Accomplishments in Task T3-3 222
Table 22: Human Cognitive Performance Simulation Parameters 229
3.3.4 Project Progress for Task T3-4 (Formulation of LCS to Learn Emergent Behaviors) 230
3.3.4.2 General Description of Task T3-4 230
3.3.4.3 Objectives for Task T3-4 over the Course of Project 230
3.3.4.4 Progress Against Planned Objectives in Task T3-4 231
3.3.4.5 Technical Accomplishments in Task T3-4 231
3.3.5 Project Progress for Task T3-5 (Formulation of FLS to Handle Uncertainty) 236
3.3.5.2 General Description of Task T3-5 236
3.3.5.3 Objectives for Task T3-5 over the Course of Project 237
3.3.5.4 Progress Against Planned Objectives in Task T3-5 238
3.3.5.5 Technical Accomplishments in Task T3-5 239
3.3.6 Project Progress for Task T3-6 (Train and Test LSASVs using PIE) 240
3.3.6.2 General Description of Task T3-6 240
3.3.7 Project Progress for Task T3-7 (Test uncertainty in human perception of system states) 261
3.3.7.2 General Description of Task T3-7 261
3.3.7.3 Progress Against Planned Objectives in Task T3-7 272
3.3.7.4 Technical Accomplishments in Task T3-7 273
3.3.8 Project Progress for Task T3-8 (Evaluate human trust and its calibration in human-machine interaction) 273
3.3.8.2 General Description of Task T3-8 273
3.3.8.3 Objectives for Task T3-8 over the Course of Project 273
3.3.8.4 Progress Against Planned Objectives in Task T3-8 274
3.3.8.5.1 EEG Band related to Cognitive Process 282
3.3.8.5.2 Experiment I: Human Trust and EEG 283
3.3.8.5.3 Experiment I with words 284
3.3.8.5.4 Hypotheses 284
3.3.8.5.5 Recruitment of participants 285
3.3.8.5.6 Participants demographics 285
3.3.8.5.7 Instrumentation 285

3.3.8.5.8 EEG procedure 285
3.3.8.5.9 Experimental design 287
3.3.8.5.10 Analysis 287
3.3.8.5.11 Results 288
3.3.8.5.12 Comparison of Intraindividual and Average Differences 289
3.3.8.5.13 Comparisons of trust and distrust 290
3.3.8.5.14 Comparison of stimulated brain regions to the baseline, trust, and mis- trust situations 290
3.3.8.5.15 Detailed comparison of average and exceptional topography maps in the trust and mistrust situations 292
3.3.8.5.16 Conclusion 292
3.3.8.5.17 Discussion 293
3.3.8.5.19 Trust and Decision Making 294
3.3.8.5.24 EEG Recording 296
3.3.8.5.25 Results 296
3.3.8.5.27 Comparison of Intraindividual and Average Differences 297
3.3.8.6 Technical Accomplishments in Task T3-8 299
3.3.9 Project Progress for Task T3-9 (Evaluation of visualization of uncertainty dynamically and intuitively) 302
3.3.9.2 General Description of Task T3-9 302
3.3.9.3 Objectives for Task T3-9 over the Course of Project 302
3.3.9.4 Progress Against Planned Objectives in Task T3-9 303
3.3.9.6 Methods 306
3.3.9.7 Results 310
3.3.9.8 Discussion 311
3.3.9.9 Conclusion 313
3.3.9.10 Technical Accomplishments in Task T3-9 313
4 Deliverables 313
4.1 Publications 313
4.2 Meetings and Presentations 343
4.3 TECHLAV Seminar Series 356
4.4 Conducted Dissertations and Thesis 357
4.5 Claimed Intellectual Properties 371
4.6 Financially Supported and Involved Students 377
5 Education and Outreach Plans 399
5.1 Student Support-Training Pathway to Next Generation of STEM Leaders 399
5.2 Collaborative Curriculum Development 399
5.3 Organizing a workshop 399
5.4 Other educational activities 400
5.5 Outreach Activities 401
References 402
LIST OF SYMBOLS, ABBREVIATIONS, AND ACRONYMS 415

List of Figures

Figure 1: Project Milestones 16
Figure 2: Modular simultaneous modeling and evaluation of 23
Figure 3: LCS and FLS Based PIE for System Modeling at Learning Mode. 24
Figure 4: Representation of the TurtleBot2 Land Rover in the Coordinate System 29
Figure 5: The effect of increasing the speed on the left wheel, (a) Angle-Longitudial axis of the robot with respect to y-axis, (b) Earth coordinate of the land rover 30
Figure 6: The effect of increasing the speed on the right wheel, (a) Angle-Longitudial axis of the robot with respect to y-axis, (b) Earth coordinate of the land rover 31
Figure 7: The effect of increasing and reducing the speed of the wheels 31
Figure 8: The system dynamics and coordinates for a quad-copter, (a) Coordinate system, (b) System dynamics and body forces 33
Figure 9: Hardware results of the PID controller with and without state estimation 35
Figure 10: Figure 10: Simulation of UGV & UAV with No Delay. 36
Figure 11: Simulation of UGV & UAV with a 1 second delay from UAV to UGV. 37
Figure 12: Simulation of UGV & UAV with a 1 second delay from UGV to UAV. 37
Figure 13: Simulation of UGV & UAV with a 1 second delay from UGV to UAV and from UAV to UGV. 38
Figure 14: Modeled Kobuki Ground Rover 43
Figure 15: Equilibrium position of the Formation 43
Figure 16: Views of the robot URDF model in RVIZ 44
Figure 17: Analysis of the kinematic model of a bicycle 46
Figure 18: Views of the robot URDF model in RVIZ: left two images without the main chassis, right image with the main chassis. 48
Figure 19: Block Diagram of LQR state feedback control of the UAV 49
Figure 20: Trajectories taken by UGV to maneuver between P 1 and P 2 with different communication delays. 50
Figure 21: Error bar plot showing error distribution for different communication delays. 51
Figure 22: Recorded pose of manually controlled UGV pattern of driving 52
Figure 23: Y-Axis Estimation 52
Figure 24: Z-Axis Estimation 53
Figure 25: Observed and Estimated 53
Figure 26: Metric Rules 54
Figure 27: ROS Network System 55
Figure 28: VL53L0X ToF sensor layout on small Ackermann steered vehicle. The arrow marked Center points towards the front of the vehicle 56
Figure 29:Photo of Simulated Agent(car) in an Environment. 57
Figure 30: Photo of old model on left and new Ackerman steered platform on right with AM-L03 sensor package. 57
Figure 31: Recorded pose of manually controlled UGV pattern of driving in circles for approximately 4s followed by a pattern of three point turns for another 6.5s 58
Figure 32: Physical Environment and Virtual Environment. 58
Figure 33: Physical UGV and Virtual UGV 59
Figure 34: Testing Model on Physical UGV 60
Figure 35: Neural Network Structure of Second Generation Genetic Algorithm Agent. 61
Figure 36:Image of DQN with GA Simulated Agent and Running Reward Graph Learning at 1000 Epochs 62
Figure 37: Image of DQN Simulated Agent and Running Reward Graph Learning at 1000 Epochs 62
Figure 38: Image of DQN with GA Simulated Agent and Running Reward Graph Learning at 580

Epochs on Track Map. 63
Figure 39: Image of DQN Simulated Agent and Running Reward Graph Learning at 580 Epochs on Track Map. 63
Figure 40: VL53L0X ToF sensor layout on small Ackermann-steered vehicle. The arrow marked "Center" points towards the front of the vehicle. 64
Figure 41: Previous Results on DQN Simulated Model. 64
Figure 42: Python simulation environment for training a vehicle using reinforcement learning 65
Figure 43: GUI Application for Testing Pre-trained Reinforcement Learning Navigation Models 66
Figure 44: Virtual Reality Testing Environment 66
Figure 45: A map captured through RTAB ROS 70
Figure 46: The database viewer showing the same image 70
Figure 47: The database viewer showing different images 71
Figure 48: Preliminary simulation results for single agent mapping in RViz environment by using GMapping package with default settings. 72
Figure 49: Preliminary experimental trials for cooperative mapping from Gmapping package with RGB-D camera feedback. It is easy to determine the lack of merging capability before the modifications. 73
Figure 50: Experimental trials for cooperative mapping from Gmapping package with RGB-D camera feedback after some modifications on default settings. Improvements can be seen in the solid lines which indicate the shapes of the walls and corridors. 73
Figure 51: Cooperative mapping from Gmapping package with RGB-D camera feedback after finalizing the modifications on default settings and solving the networking issues. It is easy to observe higher accuracy obtained **through** cooperative mapping. A better global map was obtained through increased amounts of overlapping areas and better feature matching by sensor reading improvements. The particular noises can be seen in the figure that are related to the office environment, which is mostly a result of the tables, chairs and the location of the camera on the UGV. 74
Figure 52: Map generated from slam_gmapping. 75
Figure 53: Landmark annotated version of Figure 52 76
Figure 54: General representation of communication behavior on ROS, built on Publisher-Subscriber relationship. Sensory readings are illustrated by the Topics, messages are the type of data and/or information that available through the entire system. 77
Figure 55: MATLAB Robotic Toolbox provides a global node for MATLAB to communicate to gather any ROS topics and parse the messages throughly. 77
Figure 56: Selected outputs of YOLO demonstrating correct and incorrect classifications of objects in an image frame. 78
Figure 57: Sample sets of tf2 coordinate transforms generated from testing the AM-L03 board in various poses. Coordinate transforms are visualized in RVIZ (ROS visualization program). 80
Figure 58: Localization sensing package installed on Mapper Agents. The sensing package uses a custom designed AM-L03 interface to TS3633-CM1 light sensors. 80
Figure 59: Top: Positioning of HTC Lighthouse in Room, Bottom: RVIZ Showing Multiple Vehicle Agents' Transforms Pointing Back to the Lighthouse Frame and Vehicles on Ground 82
Figure 60: Identification of a cell phone using darknet_ros 83
Figure 61:Top: Computing the Centroid of the Cell Phone, Bottom: Mapping the Centroid Coordinate to its Respective 3D Point Cloud Coordinate and Computing the Magnitude of the Position Vector 84
Figure 62: Top: RVIZ Point Cloud Visualization of a Cell Phone Including Accurate Point Measurement, Bottom: Testbed to Verify Calibration of The Camera 85
Figure 63: Positioning of the HTC Vive Lighthouse; Left: Brace for lighthouse post, Middle: Lighthouse on post, Right: Lighthouse on front instead of the post. 86
Figure 64: RVIZ showing vehicle positions identified relative to the scale Jeep vehicle. Left: vehicle

positions without tf data displayed; Right: with tf data displayed 87
Figure 65: Alignment of the HTC Vive Lighthouse in the URDF at the front position to the red colored grid representing the ground plane. Top: Bad case where tracked vehicles protrude through ground plane, Middle: Tracked vehicles not protruding as much through ground plane, Middle: Tracked vehicles not protruding as much through ground plane, Bottom: Vehicles on ground plane. 88
Figure 66: Output of multi-image source YOLOv3 operating on ZED Camera and Raspberry Pi Camera images. 89
Figure 67: Scenes from a multi-vehicle simulation in Unity with vehicles driving in a line formation. 90
Figure 68: Intersection of two vectors coming from two camera frames. The intersection marks the estimated location of a detected object's centroid. 92
Figure 69: Improved simulator performance at eliminating steady state error in the heading. 93
Figure 70: : Scene from RVIZ showing an instantaneous view of multi-robot simulation cooperative detection of multiple objects. The top image shows the location of 6 detectable objects, visualized as red pillars. The bottom image shows the objects as red pillars, various orientation estimates as long blue arrows originating from the robot pose, detected object pose estimates as light blue cylinders (multiple pillars are overlapped), "Mapper Agent" headings as short green arrows, and camera detection range as thin green outlined arc shaped areas. 94
Figure 71: Scene from RVIZ showing an instantaneous view of object detections where two agents are currently detecting one object. The remainder of pillar objects have either one or no current detections 95
Figure 72: Image from "Mapper Agent" showing multiple detections using "darknet-ros" package for YOLO V3 from two separate agents' viewpoints. The left image shows the view from the robot on the right looking towards the left, which shows one positive identificat 95
Figure 73: Scene from RVIZ showing an instantaneous view of object detections where two agents are currently co-detecting one object (vectors marked "1" and "2" and tracking a false-positive object with a vector marked "3". 96
Figure 74: Image from RVIZ showing a view from the simulation where red cylinders are true object positions, dark blue arrows are vectors towards detected objects, light blue cylinders are object detection candidates (detection object vector intersections). 97
Figure 75: 3D histogram of recorded data of all vector intersections recorded where x and y axes have unit in meters. 98
Figure 76: Detection and intersection vectors for four simulated agents over two time steps 98
Figure 77: Configuration of an example of multi-target dynamic reach-avoid scenario. The targets are in regions P23, P35 and P57, and the attacker and the defender are initially in regions P11 and P41, respectively. 105
Figure 78: Some parts of the Synthesized automaton G for the attacker in Example 1. The nodes show the true attacker sensor, which means the position of the attacker 111
Figure 79: Containerized environment to test various cloud models. 115
Figure 80: High level overview of cyber-physical system used. 117
Figure 81: Overview of Fuzzy Application Placement Controller 119
Figure 82: Graph of simulated network 120
Figure 83: Gazebo simulation starting configuration, with 4 agents located at (5,3), (9,7), (1,8), ... 122
Figure 84: Plots of the runtime over each iteration for the batch and speed desired position algorithms for the Control, Lambda, and Kappa architectures 124
Figure 85: Boxplots from the runtimes of the batch and speed layer modeling for the Control, Lambda, and Kappa architectures 125
Figure 86: Comparison of prediction on linearized model versus nonlinear model 126
Figure 87: Drivepoint scenario for UGV subject to 80% Packet loss 127

Figure 88: Comparison of models produced from original data set vs. PCA reduced data set with 2 eigenvalues.... 129
Figure 89: Temperature schedule plots for exponential, linear, logarithmic, and quadratic functions. 130
Figure 90: Box plot of the number of points removed by the exponential, linear, logarithmic, and quadratic schedules for 5, 10, and 25 epochs and trials. The y-axis is the number of points removed and the x-axis is the number of epochs and trials.... 132
Figure 91: Comparison of exponential, linear, logarithmic, and quadratic schedule's error score for 5, 10, and 25 epochs and trials.... 133
Figure 92: Run-time of exponential, linear, logarithmic, and quadratic schedules with 5, 10, and 25 epochs and trials.... 134
Figure 93: Box plots of points removed from 50, 100, 200, and 400 points by the exponential and linear schedules with 10 epochs and 5 or 25 trials.... 135
Figure 94: Comparison of original (left) and reduced models (right) from mobile sensor optimization simulation.... 136
Figure 95: Example of an intelligent transportation system, showing the communication between vehicles, RSUs, and a cloud server. V2V communication is in green, V2I communication is in blue, infrastructure to infrastructure is in red, and finally infrastructure to cloud is in black.... 140
Figure 96: Example CNN input.... 142
Figure 97: Process for generating a sample from OpenStreetMaps.... 143
Figure 98: The mean absolute error of the CNN model as it trains over 200 epochs.... 144
Figure 99: Four example outputs from CNN the error between the estimated and actual solutions for optimal RSU placement.... 146
Figure 100: Level flight and banked turn.... 143
Figure 101: DES model of normal aircraft flight maneuvers.... 143
Figure 102: Flight commands for Left Aileron Stuck Down, f_{LD}.... 144
Figure 103: Flight Commands for Right Aileron Stuck Up.... 144
Figure 104: Flight commands for Left and Right Aileron simultaneously stuck down and stuck up, f_{LDRU}.... 145
Figure 105: DES model of a Fixed Wing aircraft including faulty and normal behaviors during the bank turn.... 145
Figure 106: Interpolated A∗ path around given obstacle with initial condition at (0, 0) and target at (35, 35). 2mx2m cost map that is segmented into a 40x40 grid. White: traversable, Black: obstacle. 146
Figure 107: UGV trajectory subject to fault conditions, Green: No Fault, Red: Left Suspension Fault, Blue: Right Suspension Fault.... 147
Figure 108: Plots of summary statistics, Left: (skewness, mean) Center: (skewness, kurtosis) Right:(variance, furthest value from 0).... 148
Figure 109: Plots of spectral clustering results on 1: (skewness, mean), 2: (skewness, kurtosis), 3:(variance, furthest value from 0).... 149
Figure 110: UGV trajectory subject to fault conditions, Left: No Fault, Center: Left Suspension Fault, Right: Right Suspension Fault.... 150
Figure 111: Results of spectral clustering on UGV data, 1: (skewness,mean), 2: (skewness,kurtosis), 3: (variance,furthest value from 0).... 151
Figure 112: Optimized Fault Noise Functions, Left: No Fault, Center: Left Suspension Fault, Right: Right Suspension Fault.... 152
Figure 113: Left: Simulated path of UGV under healthy conditions using optimized PID controller. Blue: Desired Path, Red: Simulated Path, Right: Genetic algorithm convergence plot under healthy conditions. Converges to 14.39.... 153
Figure 114: Left: Simulated path of UGV with the left suspension disabled using optimized PID

controller. Blue: Desired Path, Red: Simulated Path, Right: Genetic algorithm convergence plot with the left suspension disabled. Converges to 33.56 153
Figure 115: Left: Simulated path of UGV with the right suspension disabled using optimized PID controller. Blue: Desired Path, Red: Simulated Path, Right: Genetic algorithm convergence plot with the right suspension disabled. Converges to 21.37 154
Figure 116: Accuracy convergence plot for LSTM given 1900 epochs, training accuracy: 99.08%, testing accuracy: 98.13% 156
Figure 117: Loss convergence plot for LSTM given 1000 epochs, training loss: 0.0197, testing accuracy: 0.0474 157
Figure 118: Training accuracy spread from five trial runs from five considered neural network archi- tectures. 158
Figure 119: Average required transmit power versus users' density. 159
Figure 120: Average required transmit power versus UAVs altitude. 160
Figure 121: Inverse of average transmit power versus altitude. 160
Figure 122: K-User MIMO X network Example for K = 3 170
Figure 123: K-User MIMO X network showing the application of precoders at the access points as well as stage 1 and stage 2 beamformers (Equations (50)- (52)) 172
Figure 124: Example showing the channel estimation overhead for n K-User Groups (UG). The bandwidth allocated for estimation is divided among the n K-User Groups. Each group transmits pilot signals in its allocated band. 174
Figure 125: : Comparison of our very high throughput K-User MIMO X spectral efficiencies against related technologies such as 5G-NR Massive MIMO. Spectral Efficiencies are for a 100m cell in an Indoor A channel scenario. The curve for Massive MIMO is based on the formula shown in [47]. The number of antennas is $M = K(K - 1)$ 175
Figure 126: Cumulative distribution functions of spectral efficiencies in bits/sec/Hz for K 3 in single hexagonal cells of radius 50m, 100m and 500m for Indoor A channel scenarios [49]. Results are shown both with realistic channel estimation using ML/MMSE and als 176
Figure 127: Comparison of channel estimation error variance between ML/MMSE and the Cramér-Rao Lower Bound (CRLB) 178
Figure 128: Relative frequency distribution of the singular value power sum for K 3. The upper bound for this sum is given by equation (63) 183
Figure 129: OSI 5-Layer model showing Physical Layer Security for privacy without higher layer ciphers. 186
Figure 130: System model showing a new mode for securing 5G communication through a secrecy plane. [68] 187
Figure 131: An example realization of the system geometry based on a Poisson Point Process in a 5 x 5 sq. km area 189
Figure 132: 5G Extended MOPRO signalling time schedule showing transmission of rotated reference signals. 190
Figure 133: Signalling Procedure of the 5G Extended MOPRO compatible with LTE like OFDM-reference signals. In each step, the reference signals r1 and r2 are on different frequency sub-bands. 191
Figure 134: Key Bit Error Rate vs Base Station density 194
Figure 135: CDF of Key Bit Error Rate for various Base Station densities (λBS) and codebook sizes. 195
Figure 136: (a) Robot reference frames and definition of virtual point Pv, (b) Swarm reference frames and the abstract shape that circumscribes the region occupied by swarm of robots. 203
Figure 137: (a) The communication graph of the swarm of robots in the conducted simulation. (b) Navigation of swarm of robots along a desired trajectory ζ. The robots are initially in a square formation. Their formation evolves to rectangular and parallelogram shapes along the road while

tracking ζ. 208
Figure 138: The abstract shape parameters trajectory: (a) Swarm position tracking, (b) Swarm orientation tracking, (c) swarm width tracking, (d) swarm length tracking. 209
Figure 139: Inputs to the robots: (a) the driving velocity of each robot, (b) the steering angle velocity of each robot. 209
Figure 140:(a) The heading angle of each robot, (b) the steering angle of each robot in the swarm. 210
Figure 141: Synthesizing a BT for a UAV to search and deliver an object to a particular position 219
Figure 142: Task assignment along a timeline 220
Figure 143: A human-robot collaboration (HRC) setting. 224
Figure 144: A human-robot collaboration scenario: robot transfers heavy objects from the "source" conveyor belt to the "destination" conveyor belt, and the human operator guides the robot to successfully complete the task. 227
Figure 145: The change of physical performance (blue line), cognitive workload (red line), robot performance (green line), cognitive performance (black line), and utilization factor (cyan line) are represented over time for the task complexity, $C(t) = 0.1$ 228
Figure 146: Figure 146: The change of physical performance (blue line), cognitive workload (red line), robot performance (green line), cognitive performance (black line), and utilization factor (cyan line) are represented over time for the task complexity, $C(t) = 0.4$ 228
Figure 147: Figure 147: The change of physical performance (blue line), cognitive workload (red line), robot performance (green line), cognitive performance (black line), and utilization factor (cyan line) are represented over time for the task complexity, $C(t) = 0.7$. 228
Figure 148: Complete PIE system diagram. 242
Figure 149: Block diagram of prediction model generation sub-system. 243
Figure 150: A sample report for a simulation experiment (here, Hold = 1.0, Takeoff = 2.0, Hover 244
Figure 151: : Visual representation of the scenario and the state transition diagram of the scenario. 245
Figure 152: 3DR IRIS (left) and SOLO (right) quadcopter. 245
Figure 153: Effect of partitioning data on model accuracy (left for simulation data and right for real data). The accuracy is averaged over 20 trials. 248
Figure 154: Unnormalized (left) and normalized (right) confusion matrix of the trained model. 248
Figure 155: Modified Intel Aero Ready to Fly Drone (top-left), Parrot AR.Drone 2.0 (top-right) and VICON Motion capture system (bottom). 249
Figure 156: Unnormalized (top) and normalized (bottom) confusion matrix of the trained model for real data. 251
Figure 157: The developed UAV which is used to implement the scenarios and tested in the lab environment. 251
Figure 158: Illustration of the fire detection sensor. 252
Figure 159: Proposed search pattern. The dotted line shows the boundary of each grid cell. 254
Figure 160: Graphical representation of the Creeping Line (CL-search), Expanding Squares (ES-search) and the proposed Sandclock search patterns within each grid cell. 254
Figure 161: Survey procedure. 258
Figure 162: Continuation of the survey procedure for two UAV s. The red lines are the trajectories of the two UAV s during the survey procedure. 258
Figure 163: A sample scenario while multiple UAVs are surveying multiple fire zones. 261
Figure 164: ent of stimuli identified as "leebish" versus the percent likelihood indicated by the machine estimate in the second experiment. The four different stimulus conditions represent varying levels of uncertainty. In the most uncertain condition of untrained stimulus/untrained view, the responses are closest to the percentage value given by the automated estimate. 265
Figure 165: The mean accuracy of responses for each type of machine estimate in experiments 1

and 2. 266
Figure 166: The display interface. 269
Figure 167: Reliance on the machine estimate in the present experiment. Participants' decisions closely reflect the machine estimates of 25%, 50%, and 75%. 270
Figure 168: Reliance on the machine estimate in the second experiment. Participants' decisions show increased reliance on the machine estimates as the level of uncertainty of the stimuli increases 271
Figure 169: Brain Anatomy. Source: [225] 281
Figure 170: Example of g. GAMMAcap & g.HIamp 286
Figure 171: International 10-20 system for electrode placement 286
Figure 172: Coherence topography map (alpha waves) 290
Figure 173: Coherence topography map (beta waves) 291
Figure 174: Coherence topography map (gamma waves) 291
Figure 175: International 10-20 system for electrode placement 297
Figure 176: Comparison of the trust level and the use of an automatic control 298
Figure 177: Comparisons of the intraindividual differences to the alpha (a), beta (b), and gamma 300
Figure 178: Comparisons of trust and mistrust differences to the alpha (a), beta (b), and gamma 301
Figure 179: Cogntive Fit Theory 304
Figure 180: Hypothese development 306
Figure 181: Contaminants and the maximum contaminant levels 306
Figure 182: Distribution characteristics of generated parameter values (n = 120). 307
Figure 183: Stimuli presented in graphical format 307
Figure 184: Stimuli presented in numerical format 308
Figure 185: Study block design. Conditions were randomized. Depending on the condition, stimuli were presented in graphical or numerical format. 308
Figure 186: 22-channel electrode montage 309
Figure 187: Participant undertaking experiment 309
Figure 188: Subjective ratings by NASA TLX subscale for the graphical and numerical conditions Error bars are standard errors. The asterisk indicates a significant difference (p 0.05). 310
Figure 189: Mean theta, mean alpha power and mean TLI for each participant and all experimental conditions. 311
Figure 190: Mean EEG TLI in each task condition for each participant, and overall graphical vs. numerical group means. Error bars are standard errors. 311

List of Tables

Table 1: Estimation Performances 53
Table 2: 100 Time of Flight Sensor Samples at Close and Long Range 56
Table 3: Agent 1 Model Characteristics Used in DQN With GA Simulation 60
Table 4: Agent 2 Model Characteristics Used in DQN With GA Simulation 61
Table 5: Agent Model Characteristics Used in DQN Simulation 61
Table 6: Trials for Processing Image Features using ORB (Image Size 640x480 69
Table 7: Trials for Processing Image Features using ORB (Image Size 320x240) 69
Table 8: The minimum, maximum, mean, and variance of the tts of 1000 pings between the end device, simulation, edge device, and cloudmin (ms) max (ms) mean (ms) variance (ms) 121
Table 9: Percentage of packets dropped over 1000 pings between end device, simulation, edge device, and cloud 121
Table 10: Comparing the runtime of different prediction algorithms 126
Table 11: Comparison of estimation algorithm runtimes 128
Table 12: Time to calculate for 50, 100, 200, 400 data points with 5 and 25 trials in seconds 136
Table 13: Results of training CNN over 200 epochs with batch size of 64 samples 144
Table 14: System model parameters for key derivations 171
Table 15: Simulation Parameters 177
Table 16: K-User MIMO X Interference Alignment conditions, K = 3 180
Table 17: Simulation Parameters 193
Table 18: Action templates for case study 218
Table 19: Mission tasks expanded using Algorithm 1 218

1 Summary and Introduction

The Testing, Evaluation, and Control of Heterogeneous Large-Scale Systems of Autonomous Vehicles (TECHLAV) Center, led by North Carolina Agricultural and Technical State University (N.C. A&T), in collaboration with partners at the University of Texas at San Antonio (UTSA) and the Southwestern Cooperative Control of Large-Scale Autonomous Systems of Vehicles (LSASVs) Indian Polytechnic Institute (SIPI), are conducting research activities to address two fundamental grand challenges:

1. Teaming and integrated with human operators

2. Testing, Evaluation, Validation, and Verification of LSASVs These challenges are met through three interleaved research thrusts:

- Thrust 1: Modeling, Analysis and Control of LSASVs
- Thrust 2: Resilient Control and Communication for LSASVs
- Thrust 3: Testing, Evaluations and Verification of Large-scale Autonomous Vehicles (TEVLAV)

Following are the main achievements during this reporting quarter of TECHLAV.

Thrust 1 (Modeling, Analysis and Control of Large-scale Autonomous systems of Vehicles [MACLAV]) Thrust 1 developed scalable methodologies to improve modeling, analysis, localization, navigation, and control of Large-scale Autonomous Systems of Vehicles (LSASVs). Thrust 1 has two sub-thrusts: 1) Modeling and analysis of LSASVs and 2) Cooperative localization, navigation and control of LSASVs. The highlights in this thrust for this reporting quarter are listed as follows:

- In the vehicle modeling task we developed kinematic and dynamic models for UGVs and UAVs to represent vehicles in the testbed. With those baseline models, performance of the models were compared to physical UGVs and UAVs from the testbed. Model parameters were estimated and adjusted until the error between the two were reduced to a minimal acceptable range.

- As vehicles in a LSASV can rely on network connected computation for decision making, research was performed in the various types of communication problems present within a networked system. Improvements were made into the physical models of UGVs and UAVs. Simulations and physical experiments were performed to ensure that the developed models were appropriately representing the vehicles. Truetime simulations in MATLAB were used to simulate realistic network scheduling and channel effects. Python code was developed for the physical experiments.

- State-estimation techniques were developed for mitigating time delays in a system. Simula-tions and experiments were performed to determine the model validity using multiple UGVs and a single UAV.

- Proposed a 3D detection system trained based on some hyperparameters that presents a significant performance specifically for classes that have more annotated objects in the dataset. However, an optimized training method can increase the accuracy better while the speed of the detection system is still acceptable.
- Using transfer learning for training the proposed 3D detection system that presents a better performance specifically for classes that have more annotated objects in the dataset. However, we utilized other aspects of transfer learning to see if it is possible to further increase the performance of the detection system.
- Utilizing transfer learning as a feature extraction method for training the proposed 3D detection system that slightly improves its performance specifically for car and pedestrian classes that have more annotated objects in the dataset. However, to see if it is possible to gain better performance for the detection system, we investigated some other training methods for the model as well.
- Using Learning without Forgetting (LwF) method for training the proposed 3D detection system that has better performance than feature extraction method and fine-tuning method, specifically for the new classes added to the system. This method helps adapting the system with new tasks with no need to train it on data for existing tasks.
- Proposed model performed with higher accuracy by updating training method and network parameters according to data. Depending on the data correlation and structure, fittest algorithm can be different. In addition to this results, making network deeper and more complex causes more computational time however having higher performance.
- Developed training methods and updating network parameters according to data. With the improvements model error in terms of MAPE (Mean Absolute Percentage Error) is 0.051 which means around 91% of enhancement in the performance of data-based model compare to the first model.
- With the goal of improving the autonomous navigation capabilities of LSASV's, reinforcement learning has offered a promising opportunity for enhancing the behavior and reliability of these systems. A simulation environment was developed in Python for training a virtual UGV to avoid collisions with surrounding obstacles while navigating towards a goal position.
- Genetic algorithm was studied as a potential improvement for training the network throughout the learning process. More development would be required to make more consistent results with the algorithm.
- To deploy the trained algorithm on a physical vehicle, efforts were made to ensure the safety of the mobile system and prevent costly collisions with physical obstacles in a test environment.
- To assist the evaluation of reinforcement learning strategies using the physical vehicle, a GUI application was developed in Python for accessing sensor data (range sensors, imu, video camera) throughout testing, additionally providing useful features including keyboard teleoperation and reinforcement learning policy selection.

- A new approach was proposed to use virtual reality for tracking the actual vehicle in an empty test space and constructing a virtual environment around it so that collisions between the real vehicle and virtual obstacles could be examined. The virtual reality environment was constructed in Unity, and the previous Python simulator for training the vehicle was adapted to this software using the ML-Agents Unity asset package.

- In an effort to maximize usability of the intelligent mapping algorithms in a variety of environments, research was performed into the different datasets containing indoor and outdoor object classes. A data fusion technique was applied to color, depth and LIDAR data collected from sensors onboard the Kobuki Turtlebot 2 robot. Constrained by the current hardware resources, it was determined that the process for understanding a large number of classes in real-time involves first training a deep network architecture on a variety of different datasets, and then dynamic loading of the trained network weights based on contextual logic. A process was described for mapping of important objects, saving storage space and processing power, while increasing resolution of logged object structures.

- Simulation of multi-vehicle object pose estimation for cooperative localization in mapping was performed using monocular camera models with local coordinate providers. These developments will also be used for correlating object poses to uniquely identify and label objects. Platforms with depth capable cameras will also be used in conjunction with this system to get better estimates of object location. Noise and pixel quantization characteristics were added to the simulation to get a better estimate of realistic issues faced in testing the algorithm in the hardware testbed. Depth enabled camera platforms will add a better sense of depth estimates when fused with the developed monocular algorithm.

- A ROS-compatible Unity simulation environment was developed using "ros-sharp" and custom model files of Mapping Master and Mapper Agent vehicles. The purpose of using this environment is to have a higher level of realism in simulation of visual processes for the cooperative localization and mapping task. Realistic effects of cloud cover, dusty environments, and lighting can all be modeled with this simulation environment. More complex environments will aid in development of vision processes as they better mimic real world conditions.

- A process was developed using the testbed for the intelligent detailed mapping of important objects only, which is beneficial in saving storage space and processing power, while increasing the resolution of detailed object scans.

- Developed several complex algorithms to automatically design trajectories for autonomous vehicles in a dynamic adversarial environment. These algorithms generate paths for the autonomous vehicles in real-time to ensure their safety and reachability to a desired target while respecting the dynamics of the system and its physical limitations. The resulting control signals to implement the winning trajectories are smooth. A comprehensive analysis on cor- rectness and existence of the solutions and complexity analysis of the algorithms are also provided. Several scenarios have been addresses including single target reach-avoid, multi- target reach-avoid.

- Investigated the use of cloud computing in several LSAVS environments, including autonomous boats, mobile sensor networks, and intelligent transportation systems. A cloud computing

based algorithm was developed to control a swarm of autonomous boats tasked with rescuing victoms from a sinking ship. The algorithm is responsible for collecting information on the location of victoms and assigning a boat to rescue them quickly and efficiently. A second algorithm was developed to find the optimal location for a swarm of mobile sensors so that they can best monitor a region. This second algorithm collected data on the environment, modeled the signal using the data, then determined the optimal position for the mobile sensors based on the model. Finally, the use of cloud computing was investigated for the field of intelligent transportation systems. The fields of vehicle platooning and roadside unit placement were focused on, and their implementation was investigated.

- Developed algorithms for controlling these systems utilizing cloud computing with a focus on the architecture of the cloud layer. A cloud-edge architecture was proposed for the above applications, which allowed for cloud computing to have both an accurate and quick solution to the problems. The architecture is able to switch between two algorithms dynamically, one that provides a quick solution using a subset of the data and a second that provides an accurate solution using all the available data. The placement of these two algorithms in a fog computing environment was investigated, allowing for a more refined and better performing system. The cloud-edge architecture was implemented on multiple industry cloud computing systems.

- Robot-in-the-loop simulations were used to test the validity of the control algorithms utilizing industry cloud computing systems. The simulation included physical devices for the robot and edge device. Communication between the two was achieved through a wireless connection, and a wired connection was used between the edge device and the cloud. The network performance was recorded during testing in order to help assess the impact of network communication on the cloud-edge architecture.

Thrust 2 (Resilient Control and Communication of Large-scale Autonomous systems of Vehicles [RC2LAV]) Thrust 2 developed systematic techniques, tools, and algorithms to enhance the reliability and efficacy of the control structure and the communication backbone for Large- scale Autonomous Systems of Vehicles (LSASVs) integrated with human operators in dynamic and uncertain environments such as the battlefield. Thrust 2 has two sub-thrusts: 1) Developing fault-tolerant mechanisms for LSASVs and 2) Developing a reliable distributed communication network for LSASVs. The highlights in this thrust for this reporting quarter are listed as follows:

- Developed asynchronous and semi-asynchronous diagnosers tools which allow for diagnosing a system (faulty, or non-faulty) without requiring the restarting of the system under diagnosis. The developed asynchronous diagnoser is able to be applied to a system during any period of operation, where there is no information about the state and conditions of the syestem understudy. The semi-asynchronous diagnoser, however, can be applied to a system when there is uncertain information about the state and conditions of the system understudy in the form of a set of particular states or modes. The advantage of these developed diagnosers is that the systems under diagnosis are able to continue operating, and are not required to be shut down and re-initialized.

- Developed an active-learning technique to construct the diagnoser which detects and identifies occurred faults and their types by monitoring the observable behaviors of the plant. The

algorithm will actively make two types of queries to an oracle: “the membership queries” and “the equivalence queries.” Receiving the answers to these queries, the algorithm gradually completes a series of observation tables leading to the construction of the diganoser.

- Considered the existing techniques for formulating, developing and modeling fault diagnostics and prognostics framework for single agents. Algorithms of the proposed fault diagnostics and prognostics framework included Multi-Basis Clustering and Optimized Cluster Track- ing. Multi-Basis clustering procedure combined principal component analysis (PCA) based dimensionality reduction with an unsupervised clustering technique. Initially, a single principal component transformation matrix called raw basis was constructed from the signal data. Data points from individual clusters or modes were then identified using sets of indices. A single principal component transformation matrix was then recomputed for each individual mode using the corresponding index set, leading to different mode basis for a distinct operating mode. From this theory, the functionalities of the Diagnostics and Prognostics framework (DPF) was decided to include (i) Multi-Basis Clustering and (ii) Optimized Cluster Tracking. Multi-Basis clustering strategy consolidates principal component analysis (PCA) by decreasing its dimensionality using an unsupervised clustering strategy. The basic techniques followed in DPF are data dimensionality reduction, clustering and forecasting analysis, prognostics, diagnostics, condition monitoring and decision making. The DPF framework assumes that the system accepts the elements got from sensor signal(s), gathered through data acquisition device(s), are accessible as input vectors. Depending upon the kind of sensor utilized, the raw signal itself can be utilized as an element, and subsequently, needs no element extraction process. DPF is essentially an autonomous channel for inconsistency identification that takes highlights/flags and yields demonstrative and prognostic results for decision making.

- Created experimental test bed including two Kobuki Turtlebot 2 robots labeled as "Faulty" and "Helper". In terms of hardware configuration, while Faulty robot only uses wheel encoders that provides information about its movements, Helper robot uses wheel encoders and Kinect sensor. When a failure is detected, two types of diagnostic methods are considered. The first method was called the local method. In this case, Faulty robot detects the fault by using its odometry. The second diagnostic method was called the cooperative method. Sometimes, Faulty robot cannot determine the failure because odometry itself can be source of the failure. Helper robot plays an important role. It uses Kinect for determining Faulty robot’s location visually. This feedback gives Faulty robot to find source of the failure.

- Another objective was reconstructing faulty-noisy signals acquired from a dataset sourced from a "DJI Phantom 3" drone in landing and take off conditions. A Deep Recurrent Neural Networks was used, hosted on Chameleon Cloud Services. In ideal stable flight conditions, all four rotors should be rotating at approximately the same speed. In a simulated faulty case, control signals passed via radio frequencies were received in a noisy or corrupted state. The signal before radio transmission had the shape of a standard PWM, while the received one had two states. In this situation, the reconstruction methods are required to rebuild the faulty signal(s) in order to maintain the stability, and prevent drone deviation. The dataset had provided four original signals without noise which were from the sender’s control device, three noisy signals without fault and one faulty-noisy signal belongs to one of the motors’ drivers.

- Implemented Pioneer 2 development platform in search of the means for detecting whether the tires were not inflated properly. To collect the data, the bot was driven in a straight line with various amounts of tire pressures. The data collected during the experiment was the commanded linear and angular velocities, the wheel encoded estimated position, IMU data collected from a BNO055 IMU sensor, and the tire pressure (measured at the beginning of the experiment).

- A Recurrent Neural Network (RNN) was used on the Pioneer 2 platform. Unlike a traditional deep learning network that makes predictions on single instances of data, the RNN back-propagates through a time-series of data to predict the next step in the sequence. This behavior is almost perfect for the problem at hand. However, simple RNN neuron architecture suffers from a phenomenon called vanishing/exlpoding gradients, where the information the model uses to generalize itself does not allow it to actually learn. To address this, the simple neuron was replaced with a Long Short-Term Memory (LSTM) unit. The LSTM determines what information is relevant or irrelevant at every learning iteration so to avoid the problem of vanishing/exploding gradients.

- Fuzzy logic was employed as a means of mitigating the predicted faults and determining the most detrimental system failures in order to maintain system stability and security. In hopes of capturing system-wide performance metrics, potential faults were analyzed from both hardware and network points of contention. The performance of a single subsystem fault was tested and validated by applying various degrees of tire deflation to a Pioneer UGV. In this experiment, the only decision to be made is an angular velocity required to correct the course of the UGV. The second experiment was limited to DoS style attacks, but this could easily be scaled to apply to many different types of cyber-related anomalies. The experiment consists of a simulated system of UGVs led by a master agent where its constituent agents use position and orientation data from each agent in the system to maintain a formation according to where the leader may go. A DoS attack was simulated by selecting one agent at random whose position and orientation data topic publisher is stalled in such a way that the data transmission delay is modeled to mimic DoS delays. This is done by withholding data according to a uniform random distribution similar to that of corrupted network performance metrics. Using the sequence stamps from each data packet being published from each agent, the probability of whether each respective agent is corrupt or not is evaluated. This statistic is re-evaluated incrementally as the system goes on so until the prediction converges on the corrupted agent. Once the corrupted agent is identified, its communication is severed with its constituent agents so as to retain stability with the rest of the system.

- A large focus was put on detecting unknown faults. In order to address this, the learning scheme was adapted from purely supervised learning to a semi-supervised learning. To test this, the tire inflation data set used in previous experiments was used. The goal was simply to label an unlabeled data set given ideal performance data. This was done by using hierarchical clustering to detect clusters of fault data and assign them to respective feature sets.

- Future algorithm development benefits from a research platform over which we possess more control. Specifically, a platform which is described by more reliably measured variables and that can have faults induced or removed for rapid data collection and subsequent algorithm

training. Unity engine was selected as the simulation environment due to its capabilities in producing training data as well as its potential for integration with physical prototypes.

- Task T2-5 investigated possible solutions in 5G wireless framework to establish and develop reliable communication network among autonomous agents. Since the communication network must be functional without the presence of the communication infrastructure, we selected device-to-device (D2D) communication. However, establishing D2D links requires addressing some technical challenges such as interference management and spectrum sharing. In order address these challenges, and enhance the data transmission speed, D2D links are established in the mmWave band.
- Enabling D2D communication in the mmWave band requires addressing several technical challenge, such as blockage detection, beam alignment, antenna beamwidth optimization and peer association and etc.
- In task T2-5 theses technical challenges are address through proposing novel algorithm and frameworks using stochastic geometry, game theory and machine learning. A summary of these methods are provided in this report.
- In order to minimize the effects of malfunctioning agents on the overall networked systems' performance, trust between agents are incorporated and analyzed using simulation and analytical methods.
- By the optimal search algorithms, it was possible to find a suitable topology for any number of agents to reduce the overall communication time as quickly as possible.
- A neural controller is combined with adaptive control to make the plant output to robustly follow the output of the reference model.
- Developed a new method of Physical Layer Security that ensures privatization while avoiding the costly overhead of traditional cryptographic methods of securing digital data. In 5G, secure wireless communications is an afterthought to the standard but will become necessary in NextG infrastructure.
- Designed a new algorithm for leveraging K-User MIMO technologies for $K \geq 3$. This algorithm overcomes the requirement in K-User MIMO of having perfect interference cancellation, which often limits the transmitter and receiver to only three transmit antennas and three receive antennas.

Thrust 3 (Testing, Evaluation and Verification of Large-scale Autonomous systems of Vehicles [TEVLAV]) Thrust 3 developed and provide technologies and tools for testing, eval- uation, validation, and verification of heterogeneous Large-scale Autonomous Systems of Vehicles (LSASV). Thrust 3 has three sub-thrusts: 1) Development of formal modular verification approaches for LSASVs, 2) Data-driven testing and evaluation of emergent behavior of LSASVs, and 3) Verifi- cation and validation of human-system interactions. The highlights in this thrust for this reporting quarter are listed as follows:

- Developed a motion planning and control framework that leverages local sensory inputs and communication information to transfer a large number of robots from the current location to the desired location in a dynamic environment in the presence of robot drop-outs, communication links failures, and limited communication bandwidth capacity. The framework plans the desired distribution of the swarm and controls individual robots in the group to steer the whole swarm towards the target swarm distribution to accomplish a coordinated collec- tive motion. The proposed distributed control laws are independent of robot permutations and do not assign unique labels to identify individual robots. A comprehensive stability and convergence analysis of the algorithms is also provided.

- Developed an automatic tasking approach for decentralized coordination of a heterogeneous team of autonomous agents. The agents have different capabilities in terms of executing different tasks. In the proposed framework, the collaboration can take place when an agent cannot perform a part of a mission individually but can accomplish the mission in collaboration with other agents which have complementary capabilities. For this purpose, a hierarchical modular coordination algorithm for synthesizing and executing local Behavior Trees (BTs) so that the agents can collectively achieve a set of tasks is developed. Further, a two-level auctioning algorithm is incorporated into the developed framework to assign tasks among the vehicles with lower costs. The developed framework allows a trade-off between the total cost and the duration of the accomplishment of the tasks through an embedded cost function.

- This task discussed three novel models in human-robot collaboration: human cognitive performance model, model of trust in the robot operation, and human-multi-robot performance aware trust modeling. In human cognitive performance model, a novel mathematical representation of dynamical human cognitive performance model was proposed for an Human-robot Collaboration (HRC) framework based on the associated human physical workload, the robot added workload, and the human cognitive workload. Furthermore, considering the research challenge in quantification of trust, a novel time-driven performance-aware mathematical representation of human operator's trust in the robot operation was proposed for an HRC framework based on the associated human physical workload/physical performance, the human cognitive workload/cognitive performance, and the robot added workload/robot performance. Moreover, the proposed two models were extended to multi-robot scenario to identify the impact of human trust in the multi-robot operation considering the effect of individual robot, and the overall robot performance as the added workload to the human operator. Simulation results have been provided to validate all the proposed collaboration models and analyze the effects of variations of the involved parameters.

- In Task 3-4, we reformulated the data-driven emergent behavior analysis problem into an online data stream classification problem. Considering the dynamic and complex environmental conditions, the characteristics of emergent behaviors usually change over time and conventional learning classifier systems can not fully handle this scenario. Therefore, we transformed the problem into an online learning-based classification problem.

- We developed a clustering-based data stream classification framework to learn and classify the non-stationary properties of sensory data streams from autonomous vehicles. It can effectively capture the distributional drifts and occurrence of novel data patterns as data is continuously

generated. Moreover, it is designed with a continual learning capability that can automatically update the knowledge base with a limited amount of human supervision.

- Developed a multi-sensor fusion based method on uncertainty measures. We devised a Dempster-Schafer theory of evidence based algorithms employed in a multi sensor data fusion for classification of targets. Measurements acquired by different sensors for the various attributes of the observed target are represented by belief functions. The belief functions are then combined using a robust fusion algorithm to obtain a final belief function. The resulting final mass is transformed into probability distribution for decision making as to the class of the observed target. The developed method has also been extended to multi-class and multi-label classification problems for reliable target classification.

- Developed trust classification algorithms that effectively differentiate between human-machine trustworthy interactions based on real-time sensing of psychophysiology data. Additionally, we developed machine decision making explainablity that takes into account features that characterize trust propensities of individual operators. The explanability formulations are devised on the basis of Shapley Additive Explanations.

- A new system is proposed for performance evaluation of a UAV during real-time operation and a scenario is developed to verify the proposed system. The proposed system is implemented in laboratory environment and data is collected from external sensor to build the prediction model. The trained model is used for real time performance evaluation of a single UAV.

- Developed a custom drone from scratch to test its performance and capability to carry on a particular mission using perception inference engine. Its performance was originally tested during an autonomous flight in the lab environment. The drone is capable of carrying on 250gm of external payload and has onboard visual sensor kit for 3D sensing with a powerful Intel computer board for high-level navigation algorithm development. The PX4 autopilot for low-level control of the drone is used.

- Analyzed the robustness of previously developed perception inference engine (PIE) by introducing noise in sensor measurement and developed an ambiguous scenario to test the performance of PIE. Two machine learning techniques were compared with decision tree algorithms. The prediction accuracy of the PIE is above 95% for the ambiguous scenario and decision tree outperforms than the other two machine learning technique that have been considered (Naive Bayes & SVM).

- Developed analytical analysis of scalability for the perception inference engine (PIE) for multiple UAVs, proposed two different types of arrangements to test multiple UAVs simultaneously in a space constrained test environment and developed a new state to capture the time-ordered behavior of UAVs during the execution of a particular scenario.

- Developed a mathematical formulation to test multiple UAVs using perception inference engine(PIE), developed seven sub-task to implement the proposed system and a simulation environment using ROS, Gazebo and Pixhawk firmware which will be used to generate a synthetic dataset for PIE.

- Developed scenarios using multi-UAV system to generate synthetic data to train a prediction model which will be used to test the autonomous capability of the UAV agents during the execution of the scenarios. Also, we developed a novel search and survey technique to effectively conduct the wilder monitoring mission.

- Developed a new data-driven testing framework along with seven modes of operation and five scenarios, collected data from the developed simulation environment and used a deep learning based classifier to predict the modes of operation of UAV agents while performing in real mission. Utilized the synthetic data to train the deep-learning model and validate it by using it in the developed simulation environment. Also, deployed the trained model in simulation to evaluate the performance of a UAV while performing in a simulation environment.

- Established the link from sociological construct that play an important role in determining how the human operators make use of automated systems and cognitive process to neuroimaging techniques.

- identified different neuroscientific techniques to collect brain waves to be associated with human trust in automation.

Educational Activities:

- Supported 33 PhD students from NCAT, 6 PhD from UTSA. Also, supported 35 MS from both schools, and 55 undergraduate students from NCAT, and 30 from SIPI.

- Delivered autonomy-related courses including (1) Optimal Linear Control Systems, (2) Machine Learning and Evolutionary Algorithms, (3) Sensation & Perception (4) System Dynamics (5) Introduction to Modern Telecommunications at NCAT and Big data analytic and cloud infrastructure at UTSA, and Engineering Internship at SIPI.

- Developed and offered collaborative courses at both N.C. A&T and UTSA, including (1) Advanced Robotic Systems (2) 4G LTE Wireless Communications. This will assist in sharing the resources and facilitate the exchange of knowledge between the students and instructors at N.C. A&T and UTSA. This will further foster the collaboration amongst students from both campuses to have common advisors, sharing and assisting each other in the experimental setup and writing papers.

Outreach Activities:

- Organized bi-weekly TECHLAV seminar series that promotes sharing of ideas between N.C. A&T, UTSA and SIPI as well as task progress updates. In addition to faculty and students at N.C. A&T, UTSA and SIPI, a total of four guest speakers from industry, government and academia participated in the series. The list of organized seminars is available at "http://techlav.ncat.edu/seminars.html"

- Collaborative faculty advisement of graduate students where faculty members from NCA&T and UTSA mutually serve as the committee members of PhD and Master Students from both campuses.

- Published and submitted 32 journal and 103 conference papers, have 2 patents and 4 under consideration and 4 book chapters.
- Organized the 11th International IEEE Conference on System of Systems Engineering (SoSE 2016).
- Organized a special session at World Automation Congress 2016 in Autonomous Vehicles, Development, Modeling, and Control.
- Continued collaborations and writing joint proposals with other academic units, industry partners, and national laboratories.
- Supported a group of undergraduate and graduate students to attend the 2016 IEEE South-eastCon Hardware Competition.
- Supported a group of undergraduate and graduate students to attend the 2016 American Helicopter Society (AHS) Micro Air Vehicle (MAV) Competition. The team is selected as a finalist team.

2 Project Information

2.1 Administrative Information

Title	Name	Phone	Email
Principal Investigator	Dr Abdollah Homaifar, Director	(336) 285-3709	Homaifar@ncat.edu
Programmatic Reporter/Technical Report	Dr Abdollah Homaifar, Director	(336) 285-3709	Homaifar@ncat.edu
	Dr Ali Karimoddini, Deputy Director and Thrust 2 Lead	(336) 285-3313	akarimod@ncat.edu
Assistance Agreement Officer/Administrative Contact	Sarah Jasche, Director of Office of Sponsored Programs	(336) 334-7995	divofres@ncat.edu
	Mr Tim Linker, Assistant Vice Chancellor for Research	(336) 334-7995	divofres@ncat.edu
Financial Data Reporter	James Watlington, Associat Director of Contract and Grants	(336) 285-3013	jamesw@ncat.edu
	Christopher Ballard, Grant Accountant	(336) 285-3019	crballard@ncat.edu
Recipient Monitor	Dr Abdollah Homaifar, Director	(336) 385-3709	Homaifar@ncat.edu
	Fanta Dorley, Temp Program Assistant	(336) 285-3260	fdorley@ncat.edu
	Dr Ali Karimoddini, Deputy Director and Thrust 2 Lead	(336) 285-3313	akarimod@ncat.edu
Sub-recipient	Dr Mo Jamshidi, University of Texas at San Antonio (UTSA), Cage code 02	(210) 458-7070	mo.jamshidi@utsa.edu
	Dr Milford Muskett, Southwestern Indian Polytechnic Institue (SIPI) Cage Code	(505) 792-4618	millford.muskett@bie.edu

2.2 Programmatic Information

2.2.1 Project Description

This project will establish a multi-disciplinary Center of Excellence in Autonomy to conduct a collaborative and integrated research and education program on Testing, Evaluation and Control of Heterogeneous Large-scale systems of Autonomous Vehicles (TECHLAV). TECHLAV comprises a strong team with expertise in Control, Communication, and Human Cognition from Electrical, Mechanical, Industrial & Systems Engineering programs, and Cognitive Psychology from:

- North Carolina Agricultural and Technical (N.C. A&T) State University (nation's # 1 producer of African American engineers)
- University of Texas at San Antonio (UTSA) (2nd largest Hispanic Serving Institution and 7th producer of Hispanic engineers)

We are also partnering with Southwestern Indian Polytechnic Institute - SIPI (leading engineering institute of education among all Tribal Colleges) to provide and promote education, outreach activities and curriculum development to a larger Native American community.

2.2.1.1 Research Objectives

This center will conduct a collaborative, integrated research and education program with the focus on Large-scale Autonomous Systems of Vehicles (LSASVs) that can be deployed in uncertain and dynamical environments (e.g. a battlefield) through effective interaction with human operators.

The center convenes a diverse and distinguished industrial and academic advisory board to spur innovation through careful design, adaptation, and testing in order to research two fundamental grand challenges:

1. Teaming and Cooperative Control of Large-scale Autonomous Systems of Vehicles (LSASV) integrated with human operators
2. Testing, Evaluation, Validation, and Verification of LSASV

2.2.1.2 Public Problem Description

With advances in technologies, it is now becoming possible to deploy a large team of sophisticated unmanned air, ground, sea surface, and underwater vehicles with different capabilities to collectively accomplish complex missions which often cannot be achieved individually. Such large scale autonomous systems of vehicles could provide great flexibility to achieve tasks which are distributed in time and space. Hence, large scale control systems have emerged as a promising research area with broad military and civilian applications, such as cooperative classification and surveillance, formation flight, coordinated underwater or space exploration, cooperative mapping, mutual defense, cooperative attack and rendezvous.

2.2.1.2.1 Public Research Goals/Contribution TECHLAV aims to achieve the following objectives:

- **Modeling, Analysis and Control of LSASVs:** We will investigate the properties and emergent behaviors of LSASVs through qualitative and quantitative analyses by both model-based and knowledge-based approaches. We will then develop decentralized cooperative control techniques for mission planning, localization, navigation, decision-making and control of vehicles using hybrid supervisory control approaches, task decomposition, and cloud-based control of LSASVs to accomplish the missions assigned by a human operator.
- **Developing Resilient Control and Communication for LSASVs:** We will develop innovative techniques and tools for predicting, detecting, identifying, locating, and isolating different failures in control systems and communication links. Accordingly, to dynamically adapt to operational and situational conditions and to accommodate possible failures, techniques will be developed for recovering, reconfiguring and re-tasking of the system both in the control structure and the communication platform.
- **Testing, Evaluation, and Control of LSASVs:** We will develop formal, modular verifica- tion approaches to ensure the reliability of the developed systems. In addition, uncertainties in model parameters and external disturbances will be handled using data-driven approaches. Furthermore, we will evaluate human interactions with LSASVs by examining human perception of system states, human trust in the system and human judgment of uncertainty.
- **Demonstration, Verification and Integration:** For integrating and evaluating the results, as well as transitioning the technologies to higher TRL levels, the developed techniques and algorithms will be tested through several cooperative scenarios on a team of autonomous vehicles interacting with a human operator.
- **Broad Educational and Outreach Plan:** The TECHLAV will promote a diverse, productive and prepared workforce with more STEM graduates from underrepresented groups across the three campuses. The Center will strengthen planned academic programs at N.C. A&T and will increase multidisciplinary and cross-listed courses across the three campuses. The Center will also have outreach activities to K-12 educators and students, the community, and liberal arts and community colleges in the area to develop a STEM-educated student pipeline and increase public awareness of TECHLAV. Specifically, SIPI has active partnerships with 11 other tribal colleges and will lead the development and the curricular integration of the courses at all of the 11 participating tribal community colleges and three Native American Serving High Schools through collaboration with N.C. A&T and UTSA.

2.2.1.2.2 Expected Impact Reaching higher levels of autonomy and teaming of multi-agent systems requires each agent to accomplish assigned missions autonomously even in a dynamic environment and that each agent should have the capability of autonomous collaboration with other teammates and human operators. To address these challenges, Research Thrust 1 of this proposal considers several issues such as developing capable and scalable models for autonomous collaboration, robust and distributed decision-making, group coordination, planning, and tasking through effective interaction with human operators.

Furthermore, it is important to ensure the developed large scale systems of vehicles is resilient against failures in the system and have the ability to "react to problems in one of its components and still provide the best possible service." To address these issues, Research Thrust 2 of this proposal focuses on reliable and flexible control and communication methodologies and technologies to make the system robust against failures in the control structure and communication platform.

In addition, all the DoD Departments require testing of designed autonomous systems in compliance with safety regulations. Traditional certification practices will not be cost effective and practically applicable to large scale systems. This requires new methods combining software, hardware, and system engineering to modularly validate and verify large scale systems. Thrust 3 of this proposal explores different techniques for testing, verification, and validation of large scale systems of vehicles, particularly in the presence of uncertainty and external disturbances.

2.2.2 Methods

2.2.2.1 Detailed Description of Public Technical Approach

Thrust 1 (Modeling, Analysis and Control of Large-scale Autonomous systems of Vehicles (MACLAV)) will develop scalable methodologies to improve modeling, analysis, localization, navigation, and control of LSASVs. Managing modern large scale systems requires new scalable techniques to integrate communication, control and computation that can interact with humans through many new and emerging modalities. Thrust 1 breaks up this problem into two sub-thrusts: 1) Modeling and Analysis of LSASVs and 2) Cooperative Localization, Navigation and Control of LSASVs. Through these two sub-thrusts, different techniques will be developed for mathematical and data-driven modeling, analysis, localization, navigation, and control of distributed LSASVs. In particular, a systematic hierarchical hybrid, cooperative and scalable control will be developed for decentralized supervisory control and coordination of a team of heterogeneous autonomous vehicles which enables them to cooperatively accomplish an assigned mission under human supervision. To further manage the computational overheads of the system's components and modeling, a cloud-based approach is proposed to keep a virtual copy of the vehicles, which will be available to off-load processing tasks that otherwise would overwhelm the vehicles' real-time processing capabilities. This allows for greater flexibility to include simulated (or virtual) vehicles into a swarm of physical vehicles.

Thrust 2 (Resilient Control and Communication of Large-scale Autonomous systems of Vehicles (RC2LAV)) will develop systematic techniques, tools, and algorithms to enhance the reliability and efficacy of the control structure and the communication backbone for LSASVs integrated with human operators in dynamic and uncertain environments such as a battlefield. This thrust consists of two main sub-thrusts. The first sub-thrust, "*Developing fault tolerant mechanisms for LSASV*," proposes to develop fault tolerant mechanisms for cooperative control of LSASVs. This will be done by first predicting, detecting, identifying, locating, and isolating the occurrence of a fault through model-based and data-driven approaches in both discrete decision-making layers and continuous low layers of the hierarchical control structure proposed in Thrust 1. Then, different fault accommodation strategies will be explored for dynamic reconfiguration of the system towards recovering the faulty system and if necessary and enabled, the developed team has the capability of re-planning and re-tasking in response to changes in the environment and/or mission. In parallel, in sub-thrust 2, "*Developing a reliable distributed communication network for LSASV*," a reliable and flexible communication structure will be developed for a large scale system of vehicles which

is robust against failures in communication links, packet dropouts, network delay, high interference and jamming attacks.

Thrust 3 (Testing, Evaluation and Verification of Large-scale Autonomous systems of Vehicles (TEVLAV)) will develop and provide technologies and tools for testing, evaluation, validation, and verification of heterogeneous Large-Scale Autonomous Systems of Vehicles (LSASV). This will be done through three sub-thrusts. In the first sub-thrust, " *Developing formal modular verification approaches for LSASV*," a runtime formal verification approach will be developed which uses a divide-and-conquer technique to modularly check whether LSASV can satisfy the desired (complex) high level specification. In case of uncertainties in model parameters and external disturbances, in the second sub-thrust, " *Data-driven testing and evaluation of emergent behavior of LSASV*," we will use intelligent data-driven learning mechanisms and Perception Inference Engine (PIE) for testing and evaluation of behavior and emergent properties of LSASVs. Finally, in the third sub-thrust, "Verification and validation of human-system interaction," we will capture uncer- tainty at different levels of the system (environment, machine components, and human-machine interactions). More importantly, human operators' interaction with autonomous systems dealing with unexpected situations will be approached through two important factors: human judgment and decision-making and trust in autonomous systems. Specifically, some ecological models for human operators' judgment and decision-making with trust in autonomous systems will be tested to provide noble guidelines to design a human–system interface.

In addition to the above thrusts, the Center will develop a Demonstration, Implementation, and Integration (DII) component to validate the proposed tools and techniques that support technologies for addressing the growing autonomous system capability, effectively integrated with the human capacity to perform in a high-tempo, complex decision-making environment. The DII platforms will showcase robust execution of challenging tasks and mission scenarios envisioned by DoD such as: cooperative search and coverage to find enemy targets; automatically repair communication breakdown among autonomous systems of vehicles, and assemble systems in cooperation with human operators.

2.2.2.2 Comparison with Current Technology/Assumptions and Procedures

Thorough and efficient management of LSASVs requires development of new techniques for scalable cooperative operation of distributed systems equipped with global communication connections, as well as capabilities for decentralized local actions and distributed resource sharing while effectively interacting with human operators. In addition, new tools and techniques are needed to test, evaluate, validate, and verify LSASVs to ensure their safe and reliable performance. The traditional separation of computing, communications, and control is no longer valid, and new methods that integrate advances from different disciplines are needed. Furthermore, most of the existing solutions involve a centralized resource and task allocation, followed by decentralized execution of the tasks so that each vehicle is responsible for a set of tasks. This strategy is neither scalable nor applicable for analysis, design, and verification of large scale systems. Therefore, this Center brings expertise from different disciplines to develop more modern management of LSASVs by considering the use of cooperative operation of large collections of distributed vehicles, with local computation, global communication connections, and decentralized control actions.

2.2.3 Schedule and Milestones

2.2.3.1 Schedule Graphic

Durations of the proposed tasks have been carefully coordinated to ensure that adequate time and resources are available to investigators.

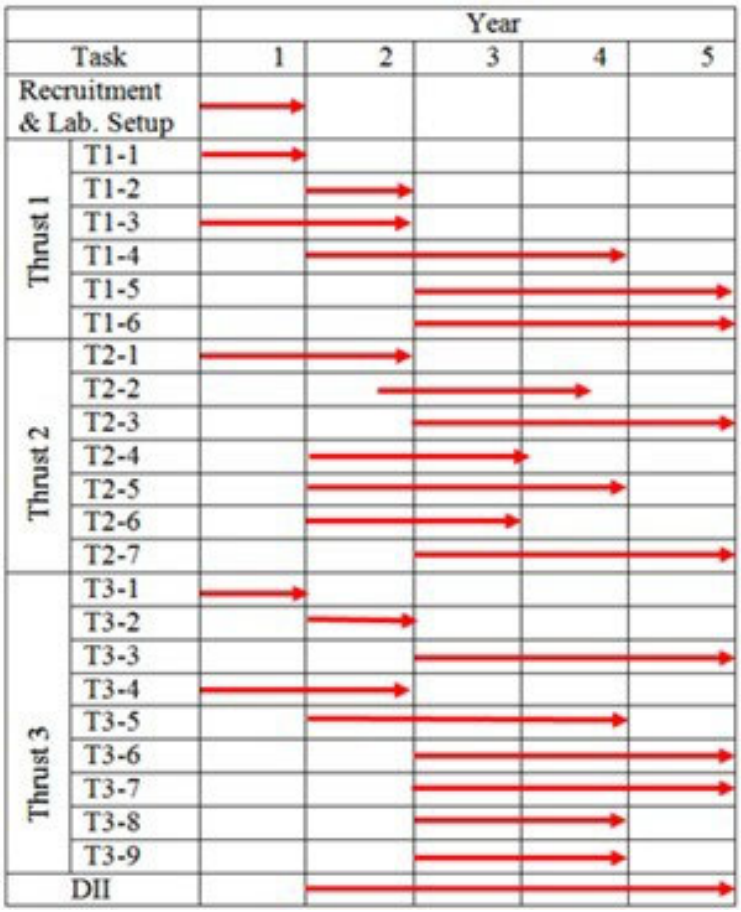


Figure 1: Project Milestones

2.2.3.2 Detailed Description of the Tasks in Thrust 1

Task T1-1. Mathematical Modeling of LSASVs: LSASVs cannot be thought of as a set of in- dependent agents since they usually interact with each other, share information and resources to collaboratively achieve given tasks. Hence, traditional methods which ignore the coupling effects between the system's components cannot be applied to these tightly coupled systems. Developing an augmented model for the overall structure of the system is also too complicated, computation- ally expensive, and impractical. To address these problems, we will develop novel methods using abstraction and model reduction techniques, which help to manage the complexity of the result- ing model. Along with the developed models, communication protocols will also be established to interconnect these systems (machines or assets) together via a wireless network encapsulating the

entire system. We will use a cloud data center to house the models and facilitate access for the LSASV systems. This will also facilitate incorporating additional services into the model such as an Object Recognition Engine and a centralized world map, which will be available to all entities in the connected systems.

Task T1-2. Data-and Knowledge-Based Modeling of LSASVs: One of the key aspects of LSASVs is the exchange and management of large datasets (numerical, textural and image) among the constituents of the system. In some applications, data computation, Bayesian networks accumulation and its security are the key pieces of information. There is a need for employment of a machine intelligence method to enable any vehicle in the system to perform in a team of vehicles. In this task, we propose that a promising approach, called "Data Analytics" be used. Data Analytics utilizes statistical and computational intelligence (CI) tools such as principal component analysis (PCA), clustering, fuzzy logic, neuro-computing, evolutionary algorithms, and data mining to reduce the accumulative large size of the LSASV datasets to a manageable size. We will then apply these tools to a) extract information, b) build a knowledge base using the derived data, c) develop a cooperative teamwork framework through the consensus among members and d) eventually develop a non-parametric cooperative model for LSASVs. This task will construct a bridge between LSASVs and Data Analytics to develop reliable models for resilient cooperative control of such systems, described in Task T2-4. One of the more recent promising data analytic tools is "Deep Learning". Deep learning is the broad term for the recent development and extension of neural networks in the Machine Learning community, which has allowed for state-of-the-art computational results in speech, image, and natural language processing tasks. We will also develop a hierarchical learning approach to extract high order representations from low level data. Data analytic tools and approaches will be further used to filter data in circulation among vehicles by eliminating communication noise, anomalies, and outliers in communicated data in a system of vehicles. This task will feed directly into other Thrusts such as test and evaluation (Thrust 3) and fault diagnostics (Thrust 2). Data Analytics also plays a central role in allowing the vast amount of data created to be converted into actionable information to be used by human operators.

Task T1-3. Qualitative and Quantitative Analysis of LSASVs: Followed by modeling LSASVs, the next step is to analyze the resulting models to investigate the properties and emerging behaviors of LSASVs. Traditional concepts developed for single agents are no longer valid for LSASVs and new notions of stability, controllability, observability, and reachability are required. For the stability analysis of LSASVs, we will focus on the input-to-state stability (ISS) of the system and will employ small gain theory, whose proven compositional properties ease the development of a modular scalable stability analysis method for interconnected large scale systems. We will also formulate the controllability of the LSASVs to realize how to drive the vehicles toward desired positions in the presence of tight interactions with other vehicles. This will be carried out by exploring the reachable space of the abstracted space, and investigating simulation-based structural properties of the system. Communication protocols will be a part of this investigation and maximum communication delays will be determined to guarantee system stability in coordination with Thrust 2.

Task T1-4. Cooperative Localization and Navigation for LSASV: A class of localization and navigation algorithms called Simultaneous Localization and Mapping (SLAM) provides a robust and accurate approach for both tasks. SLAM algorithms that utilize visual information (vSLAM) require large amounts of computational power and storage, which makes it difficult to implement in real-time for LSASV. Of the main bottlenecks in vSLAM, feature identification and matching

across a large database are the most time consuming. And so, in this task, we will implement a novel method to reduce computational time, storage requirements, and network load to perform feature identification and matching components of vSLAM in real-time. The proposed process selects out only the most feature-rich components of visual data to be used in matching and database creation. We will further improve feature detection algorithms using proven feature detectors such as ORB (Oriented FAST and Rotated BRIEF) and SURF (Speeded up Robust Features) on readily available datasets. This will significantly reduce storage requirements and calculation time. The proposed more efficient vSLAM algorithm will be implemented remotely from the LSASV entities themselves (in a cloud datacenter), taking advantage of high performance computing systems and parallel processing of algorithms. The network and cloud access delays will be compensated by predictive approaches to minimize their effects on the overall system performance. Visual information will be fused with known, learned, or sensed physical location references to create a world map. World maps will be kept in a big-data database and made equally available to all required parties. While each vehicle (agent) uses the developed SLAM algorithms to create world maps, on a LSASV level, a fusion of agent world maps is utilized to correct and provide global references to the system. Complex alignment operations to fuse world maps are to be investigated in this task. Complexity in the operations comes from many variables in LSASV, including the pose and perspectives of various sensors on the agents, errors in position data, interference between agents, and dynamic environments. Fusion of this data must be performed with high precision in order to acquire the best estimate of the position of all agents in the system as well as important spatial features within the world map. Registration is one form of fusion in which maps are compared for matching features and transformed to fit together. The registration process can be time consuming, if not performed with high levels of parallel computing. This task therefore develops a real-time registration method for multiple LSASV entities.

Task T1-5. Hierarchical Hybrid Cooperative Control of LSASV: Although a cooperative strategy will enhance the reliability and effectiveness of the team, the overall structure will be a highly complex system with many interacting subcomponents. One of the main sources of the com- plexity is the hybrid nature of the system due to the coexistence of the decision-making mechanism (with event-driven dynamics), and the low-level control of the system (with time-driven dynamics). In such a hybrid structure, the event-driven and time-driven dynamics of the system not only coex- ist but also interact with each other, and their coupling effect cannot be ignored. In addition, this complex structure must include humans in the control loop. Hence, given a team of heterogeneous autonomous systems and humans in the control loop, a challenging problem is how to design and analyze such a hybrid decentralized control structure to comprehensively capture the dynamics of all subcomponents and their interactions. To address this problem and to overcome the complexity of the system, we propose a hierarchical hybrid control structure to distribute the complexity and control tasks among the layers and among the components. Hierarchical control systems have been studied for quite some time; however, considering the concept of hierarchical control within the hybrid modeling and control framework and its application to autonomous systems have not yet been addressed. Moreover, the role of the human in terms of high level supervision and goal setting has not been captured. Recently, we have developed a hierarchical hybrid control structure and applied it to a helicopter. The TECHLAV, however, will leverage our previous work by developing a decentralized hierarchical hybrid structure for control and coordination of a team of heterogeneous autonomous vehicles to cooperatively accomplish an assigned mission under human supervision.

The subcomponents of this control hierarchy would include but not be limited to the vehicle

dynamics, continuous low level controller, planning unit, and supervisory unit. In this hierarchy, each layer by itself can be modeled as a hybrid system. A hybrid system typically consists of a set of discrete states standing for operating modes, each of which has time-driven (continuous or discrete time) dynamics. To develop this control hierarchy, we first will use a formal method to connect the hybrid subcomponents and then synchronize them to collectively form a composed structure. This can be done by properly capturing the input and output of the layers, and by synchronizing and treating the discrete states, discrete transitions and continuous transitions. In this structure, the human operator assigns the mission. The assigned mission given in the form of Linear Temporal Logic (LTL), which is very close to the human language, will be translated to a finite state machine for which we will design and implement a supervisory controller in a decentralized way. This project aims at developing a decentralized hierarchical hybrid cooperative control framework for task allocation and coordination of a team of autonomous vehicles to address more complex tasks. **Task T1-6. Cloud-Based Control of LSASV:** Cloud computing infrastructure (compute, stor- age and network) in cloud datacenters will be the platform for hosting the models and algorithms researched in this Thrust. The computing resources located in cloud datacenters will be used to keep a virtual copy of each vehicle in the LSASV. Storage resources will be used to provide the optimal configuration for Data Analytic research activities (vSLAM, world-maps, etc.). Network resources will be used to create team boundaries for the LSASV swarm performing cooperative mis- sions (shared information). The benefit of using a virtual vehicle will be to offload processing tasks that otherwise would overwhelm the vehicle's local processing capabilities. Since the computational power of remote vehicles is constrained by available battery and processing power and physical characteristics of the vehicle, complex calculations can more optimally be made using the parallel computational facilities offered in the cloud. This approach allows for greater flexibility, such as including multiple simulated (or virtual) vehicles into a swarm of physical vehicles. Large-scale systems analysis can then be performed using as many vehicles as required to verify the correct functionality of the developed models and algorithms. Human participants can also directly use the cloud to interact with the swarm of vehicles and override autonomous behavior as necessary. One of the necessary research challenges of using the cloud as part of the control loop for autonomous vehicles is in assuring adequate bandwidth and availability of a network connection between the LSASV and the cloud. We anticipate that some amount of local processing on each vehicle will be necessary to mitigate mission failures due to increased latency or intermittent outages in the network. After all, this is a key role and advantage of autonomy. Cloud-based robotic control has recently been researched and successfully used to process image data captured by robots. The image point cloud data can then be uploaded and processed in remote cloud data-centers. The result of the image processing can be used to update a common world-map of individual robots in the swarm.

2.2.3.3 Detailed Description of the Tasks in Thrust 2

Task T2-1. Developing a decentralized fault detection mechanism: The first step to handle a fault is to accurately detect the occurrence of the fault, and its nature and location. It is not possible to consider a sensor to detect any fault, as it dramatically increases the overall costs and more, importantly, not all faults are predictable. Instead, it is reasonable to diagnose a failure through system behavior and limited observations. Fault detection and isolation is part of the decision-making unit of the system in which the system behavior has been abstracted to a discrete model and the fault occurrence can be considered an event which changes the operational mode

of the system. As described above, most current research addresses fault detection and isolation for a single agent. To address fault detection for a team of agents, we will first characterize the nature of fault for cooperative tasks. We will then propose a diagnoser for a multi agent system and investigate its decomposability into local diagnosers which form a decentralized fault detection structure in combination with local supervisors. Abstracted data received in the supervisory layer of the control structure will be used to detect a fault. Due to the discrete nature of the supervision layer, we will study fault detection in the context of discrete event systems, automata theory, and supervisory control of discrete event systems.

Task T2-2. Fault Diagnostics and Prognostics through Data Analytic Approaches: The three terms of diagnostics, health monitoring and prognostics are sometimes intertwined and often assumed to be interchangeable. They are related, but not the same [43]. In the former, diagnostics identifies the nature or cause of some phenomenon, while in the latter, health monitoring keeps track of current status systematically with a view to collect information, while prognostics refers to the prediction about how something (such as sensor or actuator failure) will develop in the future. While in the previous task our focus was on the fault detection through general behavior of the system, in this task we are concerned with detection of fault in a system of vehicles through lower layers of the control hierarchy. For this purpose, we are essentially dealing with large amounts of data being exchanged among the vehicles both in simulation and on real-time robotic agents. We will utilize "Big Data" analytic techniques to detect failed sensors, actuators, eliminate noises, or cyber-attacks in data transmissions. Using our developed "Deep Learning", the raw data will be pre-processed and post-processed to create mined data and eventually predict failures of vehicles.

Task T2-3. Developing a decentralized fault accommodation mechanism: After detecting a fault, the next step is to accommodate the detected fault. For a single agent system, the fault-tolerant mechanisms are limited to a few strategies, such as structural redundancy. In a cooperative team of agents, however, we can take advantage of the flexibility of the structure. For example, a functional redundancy can be effectively developed relying on other agents, which have similar capabilities. In an alternative strategy, the agents can be reconfigured to achieve the original assigned task; and if this strategy does not work, the agents can be re-tasked in such a way as to accomplish the maximal possible portion of the assigned task. For re-tasking, we will use On-the-fly controller synthesis techniques combined with DES learning approaches such as L* learning and game theory methods. In our recent results on developing a cooperative control for the formation control of UAVs, the collision alarm has been interpreted as a faulty situation and then, collision avoidance has been handled by our developed decentralized supervisory control algorithm. In this project we will leverage our previous work by developing more sophisticated fault detection and accommodation strategies to address more general and complicated faults.

Task T2-4. Developing an adaptive effective ad hoc communication network: In this task, we will design an efficient and robust communication network for the proposed multi-agent platform that addresses the essential design objectives including high transmission rate, low transmission delay, wide transmission range, compatibility with high-speed mobility of agents, and reconfigurability. These properties allow reliable and continuous interactions among the agents to achieve collaborative tasks performing synchronization, self-healing and self-organizing. The proposed protocol requires being adaptive considering the type of agents, their capabilities versus restrictions, and their ultimate tasks in the network. This means that the agents have the option of selecting the best communication parameters including modulation technique, channel coding, and multiple access technique according to their properties and objectives in terms of desired data

transmission rate, available power, and tolerable transmission delay. The proposed communication system will also be capable of cooperative packet transmission via intermediate agents in the network in order to extend the coverage and connectivity. This requires determining the position of the agents in the network and employing them as routers in case they are not active at the moment or enabling them to provide relaying service while performing their primary tasks. Employing cooperative relaying will improve the network performance and connectivity, decrease the chance of packet transmission failure, and expand the network connectivity. It can also potentially save the limited energy of agents, since they have the option of transmission to their relatively close neighbors rather than consuming higher power to communicate to their far-distance target agents.

Task T2-5. Enhancing network performance utilizing the cognitive features of the agents: Due to recent advances in electronics and communications, the agents are currently equipped with cognitive radio devices that significantly improve the performance of the commu- nication systems. This cognitive capability allows the users to sense the environment and monitor the operation of other agents to take proper responses for the observed information. The intelligence capability of the agents provides the possibility of independent decision-making regarding their com- munications parameters including the transmission power, transmission frequency, bandwidth, and utilized coding and modulation technique. We will take advantage of the agents' cognition toward planning an efficient and reliable user-centric communication system, where most communication design parameters are determined in a decentralized fashion. This involves several design paradigms including power allocation, utilized frequency and also channel access with the goal of interference reduction in the network, taking into account the communication strategies of other agents. We will adopt a game-theoretic framework in which the rational users set their communication parameters in a way to improve their individual benefits while taking into account the social welfare. Game theory is a powerful mathematical tool to analyze the interactions among the intelligent users when they have conflicting interests or they coordinate in performing a common task.

Task T2-6. Delay-tolerant and loss-tolerant consensus in networks of agents: Communications among agents often experience substantial delays, packet dropouts, and intermittent connectivity and link breakage. The effects of these impairments on system performance are not trivial, but also not predictable. Development of an effective solution to this is still an open-ended problem. The dynamical behavior of each agent is dependent on received information. Thus, the control design taking into account communication disruptions and delays is important. Controllers in such networks should work robustly against changes in the network topology. In order to address these issues, we will perform stability analysis for networked multi-agent systems. The research aims at developing a theoretical framework to establish stability criteria for multi-agent systems considering network topology. Randomly changing network topology as well as fixed topology will be considered. For example, connection between the robustness margin to time delays and eigenvalues of network topology will be used to investigate convergence properties of consensus protocol. Also, stability in case of communication channel collapse will be analyzed, for example, through Lyapunov equations. In our recent work, we have designed controllers to improve system performance even in the presence of uncertainty by breaking the barriers of infinite-dimensionality using the Lambert W function. We will further study this problem, by considering the effect of both time delay and uncertainty on networked multi-agent systems to yield better understanding and advanced control strategies.

Task T2-7. Wireless Scale Free Topologies for Resiliency and Jamming Immunity: Network topologies with random link connectivity are often formulated in many small communication

networks. However, very large scale wireless networks are capable of exhibiting new "differentiated" properties that are observable only when the network is large and the network topology is complex. One particular property of complex networks, denoted as "scale-free," demonstrated immunity to additive noise power that was hundreds of times larger than signal power. This new wireless scale free topology is a potentially transformative model for a fundamentally new class of robust wireless sensor networks (WSNs). In scale free sensor network organizations, the fundamental entity of communication is a multi-node group or cluster of wireless nodes, rather than a wireless single node. Clusters can vary from as few as three nodes and up to millions of nodes. However important properties for resiliency occur for large network sizes. Scale-free clusters can serve as relays, recipients, or sources of network information. The clusters that we have investigated share an attribute related to network degree connectivity that is power-law distributed. Due to the fact that our protocol applies wireless node constraints, we encounter emergent properties distinct from the conventional wired networks. In this task, our hypothesis is that scale free network topologies enable high quality of service (QoS) cluster communications in high interference and jamming attack scenarios. Investigation of attack resilience, interference robustness, and control fusion of cluster to cluster sensor communications occurs in the face of poor data link quality, jammer interference, and low SNR.

2.2.3.4 Detailed Description of the Tasks in Thrust 3

Task T3-1. Developing an effective hierarchical abstraction technique: Given a (complex) high level specification, our aim is to realize whether the system can satisfy the desired specification. This problem in general may not be directly solvable since the model checking process could face the state explosion problem. Alternatively, we can use abstraction techniques to reduce the system dimension while preserving the properties of the original system. We have developed a polar and spherical Bisimulation-based abstraction technique for the motion dynamics of a team of UAV helicopters. The proposed approach is applicable to multi-affine dynamical systems. Bisimulation relation between the abstract model and original system ensures that they have equivalent behavior. Therefore, the model checking outcome for the abstract model is valid for the original system as well. Nevertheless, Bisimulation relation is a conservative and restrictive relation and is applicable to limited classes of systems. Therefore, this proposal aims at developing a hierarchical abstraction technique, which is less restrictive and can be applied to more general classes of systems including nonlinear and hybrid dynamical systems. In the first layer of the proposed hierarchy, we will model the original system by a transition system and will capture those transitions, which lead to important changes such as changes in the operation modes of the system. Then, we will use approximate bisimulation and simulation relation techniques to further abstract the model by grouping together those transitions that start and remain within a short distance in the presence of nondeterministic system evolution. This will result in a finite state model for the original system, which significantly facilitates the model checking process.

Task T3-2. Developing a compositional verification approach for LSASV: As discussed, abstraction technique can effectively facilitate the verification of a given system by converting the system to a reduced (finite) state machine. Nonetheless, the verification problem for a large scale system of vehicles, each modeled by a finite state machine, is still exponentially large and complex. For a large scale system of vehicles, which are modeled by finite state machines $M_1, M_2, \ldots, M_n$ and for a given desired specification, S, the verification problem is to check whether their collective behavior, $M = ||_{k=1}^{n} M_k$, satisfies the temporal logic specification S, where $M \vDash S$ means that

the model M satisfies S, and $\|$ denotes the parallel composition operator, which synchronizes the components. The centralized version of this verification problem is not practically implementable for large values of n. To develop a scalable verification technique for such a large scale system of vehicles, we propose a "divide-and-conquer" technique which breaks up the verification of large scale systems by decomposing the desired specification into several smaller components and then, each component is checked separately. This can be performed by decomposing the specification into $S_1, S_2, \ldots, S_n$. Then, the centralized verification problem $M \models S$ can be converted into n model checking problems $M_k \models S_k$, $k = 1, \ldots, n$, while $R(S_1, S_2, \ldots, S_n, S)$ is a logical condition relating $S_1, S_2, \ldots, S_n$ and S. For the decomposition of the desired specification S we will adopt the assume-reasoning techniques by making some assumptions about each component's environment to guarantee the satisfaction of the desired specification.

Task T3-3. Developing a modular simultaneous modeling and evaluation technique: Sufficient information about the components of a large scale system does not always exist. Furthermore, uncertainty in the model parameters and external disturbances may challenge model-based analysis and synthesis techniques. Therefore, to further improve our proposed verification approach, we will augment it with learning techniques to simultaneously and incrementally capture the non-modeled parameters of the system while evaluating the performance of the system as shown in Figure 2. The core of this algorithm is the updating mechanism for which we will use the L^* algorithm. The L^* algorithm generates a deterministic finite-state machine through minimum interactions with a supervisor whose role is to answer whether the model output matches with the actual system's observations. This information is used to incrementally update the model of the system. With this adaptation algorithm, the proposed method can be trained to capture the non-modeled parameters during the verification process.

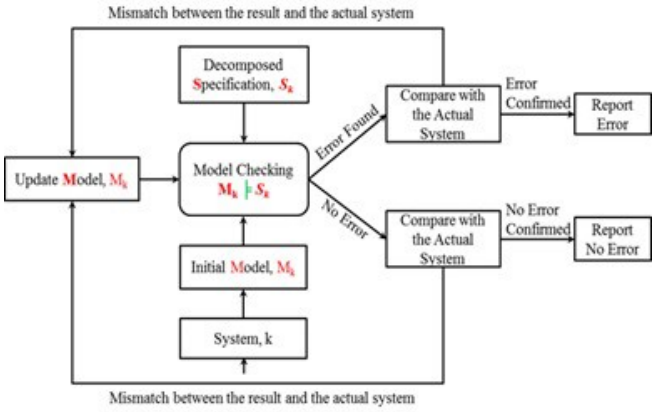


Figure 2: Modular simultaneous modeling and evaluation of LSASVs.

Task T3-4. Formulation of LCS to Learn Emergent Behaviors: Learning Classifier Systems use reward-mechanism to capture internal processes of a system based on system's environmental variables. Here, the environment for the LCS is the LSASV under test and all of the entities and features in the simulation environment. Unlike a single agent, one major challenge to address when dealing with multiple agents is methodology to standardize the different inputs of the different components of the LSASV for computer processing. This will be handled using advanced data normalization techniques. The inputs from the LSASV (like position, heading, speed etc.), the messages it transmits as well as the messages and rules it transmits, can be coded in form of string bits or classifiers which are the fundamental units of information in the LCS implementation. The outputs or "actions" of the LCS correspond to the output (emergent behavior) of the LSASV. The rules that the LCS learns are the ones that govern the transitions between emergent behavior states of the LSASV. The solution domain comprises a set of rules that collectively give a model of the LSASV's interaction with its environment as shown in Figure 3. LCS needs a trainer to provide a

reinforcement signal in the form of a reward or punishment using just a sufficient number of rules (called training data). The evaluator receives the response of the LSASV for a given input and compares it with the corresponding action of the LCS system and generates a feedback to the LCS block. The intention here is to train the LCS block using this known set of rules. Once sufficiently trained, the LCS can be utilized for predicting correct state transitions for previously unknown input scenarios.

Task T3-5. Formulation of FLS to Handle Uncertainty: To evalu- ate the LSASV's perceptual capabil- ity representative sensing tasks will be used such as the detection of en- emy targets in order to launch an at- tack; formation execution; task allo- cation and detection and localization of landmarks. The task here is to translate experts' heuristic knowledge into fuzzy IF-THEN statements. Earlier work will be extended to automate

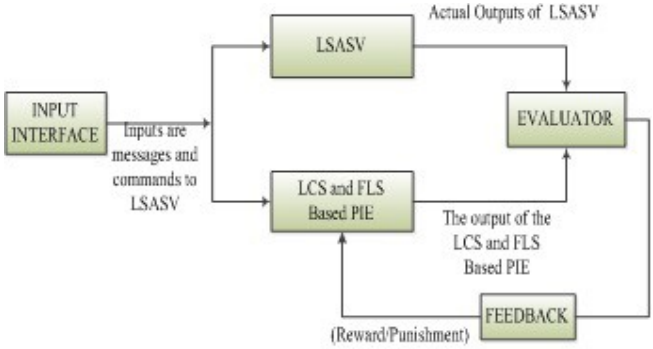


Figure 3: LCS and FLS Based PIE for System Modeling at Learning Mode.

the process of categorizing the input/output relations to associated decisions. The fuzzy logic approach can provide the tester with the reasons for a particular cause of action in human understandable terms. The most fundamental aspect of this connection is that the uncertainty involved in any problem-solving situation is as a result of some information deficiency, which may be incomplete, imprecise, fragmentary, not fully reliable, vague, contradictory, or deficient in some other way. The general framework of fuzzy reasoning allows handling much of this uncertainty. The effects of uncertainty in a system can be handled in a better way by using type-2 fuzzy logic because it offers better capabilities to cope with linguistic uncertainties by modeling vagueness and unreliability of information.

Task T3-6. Train and Test LSASVs using PIE: Embedded in this task is the task of inferring the LSASV's perception of its environment because this perception determines what the LSASV's active behavior state should be. As the complexity of LSASV increases, efficiency and cost are considered critical factors in T&E; the search for near-exact models of behavior and perception inference becomes challenging. The method for developing these models must use observations of the LSASV and its interactions with its environment and learn from these interactions. LCS that embraces a hierarchically cooperating and competing set of classifiers can meet these needs. The uncertainties associated with the system can also be modeled with a type-2 Fuzzy Logic System. Once sufficiently trained, PIE will be able to correctly predict the outputs of LSASV given input scenarios that were not exposed to the PIE during the training phase of the PIE. The PIE infers the internal processes/transitions of the LSASV's and predicts behavior of the LSASV next state, such as the ability to determine an enemy target. Because the rules governing a LSASV's behavior are not fully known, nor are the capabilities of its sensor systems, the behavior and perception models require ongoing adaptation in order to approximate unknown scenarios intuitively. Secondly, the internal parameters governing the behavior of the LSASV might change with time, which may cause the PIE's behavior to deviate from the behavior of the LSASV. The online adaptation of the PIE enables it to adjust its training parameters to align itself with the LSASV's behavior.

Task T3-7. Test uncertainty in human perception of system states: Within large-scale

semi-autonomous systems, operators must accurately perceive information being communicated prior to making a decision about how to intervene in the system's behavior. Thus, it is critical to reduce error in operators' perceptions of system states. This research will verify/validate displays for operators of semi-autonomous systems by testing humans' perceptual judgments of information transmitted by the system in real time. Testing will consist of behavioral experiments in which participants make judgments about parametrically varying cues from LSASVs. Because large-scale systems could involve multiple operators, experiments will include testing of how multiple participants perceive system states within the same environment. In some, participants will complete multiple training/testing sessions to observe how their perception of information changes with experience. Based on these findings, further experiments will use refined versions of the multimodal displays in simulated LSASV scenarios, and will focus on identifying and eliminating ambiguities in the signals and error rates in human users.

Task T3-8. Evaluate human trust and its calibration in human machine interaction: Studies in trust have demonstrated a compelling relationship between decision makers and the use of automated devices in making crucial judgments. The main objective of this task is to conduct simulated experiments and to develop empirical methods for calibrating trust metrics among human-automation interactions. The subordinate objectives include: a) Defining the necessary factors for calibrating trust in a two-dimensional domain of continuum and human perception when human and autonomous machine interact to perform a task, b) Developing a simulated environment to collect trust-related data in human-automation interaction systems, and c) Calibrating human trust in machine based on experimental data.

Task T3-9. Evaluation of visualization of uncertainty dynamically and intuitively: In addition to error reduction and improvement of trust in the human-machine interaction, a further means for coping with uncertainty in semi-autonomous systems would be to enable human awareness of uncertainty itself. This task proposes to develop and test a prototype that represents uncertainty dynamically in order to support operators' judgment and sense-making. An experimental framework to examine the efficacy of the developed prototype for tactical visualization of uncertainty through many dimensions and morphing mechanisms will be provided to collect objective mea- sures of performance. Studies into display factors influencing decision-making under uncertainty are planned. Specifically, indications from past research show that a more concrete representation may lead people to believe that information is more certain than it actually is, while a degraded or less concrete representation may result in an understanding as less certain. The proposed research will utilize a multi-dimensional framework to systematically vary display factors to simulate condi- tions of degraded information corresponding to uncertainty present. For example, information could be represented through the use of numeric ranges or graphical areas rather than point estimates. Experiments will evaluate and compare how these various visualizations influence human judgments in operating simulated LSASVs.

2.2.4 Deliverables Description

The deliverables will be quarterly reports and final report detailing the tasks' progress and the accomplishments, which will be submitted to the Air Force Research Laboratory.

2.2.5 Technology Transition and Technology Transfer Targets and Plans

Demonstration, Implementation and Integration (DII) uses scenario-driven cases to integrate the proposed tasks in the three Thrusts to achieve coordinated outcomes. For this purpose, a group of different multiple ground robots and UAVs will be used to perform cooperative tasks such as cooperative search and coverage and coordinated rendezvous in a decentralized fashion. These vehicles are equipped with high-quality cameras, high-capacity communication devices, various sensing devices, laser sensors, and onboard positioning devices. This wide variety of autonomous vehicles puts our test-beds in a unique position to investigate different applications in direct interest of DoD. A human supervisor defines the mission for the multi-agent system and these agents then coordinate with one another to execute the assigned mission. This coordination involves control, synchronization, decision-making, positioning, formation and data sharing. The capability of the system in terms of localization and positioning, distributed task coordination in navigation and target tracking will be investigated here to evaluate the outcomes of Thrust 1 and its proposed tasks. The developed algorithms for fault detection and fault accommodation through optimum re-formation and reconfiguration in the case of detected faults or agent loss also will be examined in accordance with Thrust 2. Besides, near-continuous connection among the vehicles in the network, near-constant accessibility of the vehicles by the human operator and coordination among the agents in task management will be accomplished through the communication protocols developed. The communication network will support different radio transmission techniques, distributed multi-hop communications, optimum routing planning, real-time communication and robustness to jamming attacks to address the desired goals in Thrust 2. The testing, evaluation, and verification methods proposed in Thrust 3 will be studied using these test-beds. Since the interaction between human operator and the vehicles is a key factor in verifying the proposed semi-autonomous system, we will investigate this matter from various aspects of testing the several human operators' behavior while having uncertain knowledge of the system, human trust during human-machine interaction and tactical visualization of uncertainty information as detailed in Thrust 3. The behavior of the entire system integrating the proposed algorithms will be tested under various scenarios including dynamic environment, agent failure and hypothetical attacks to verify the objectives of all three Thrusts.

3 Results and Discussion

3.1 Project Progress and Project Plans for Thrust 1: Modeling, Analysis and Control of Large-scale Autonomous Vehicles (MACLAV)

3.1.1 Project Progress for Task T1-1 (Mathematical Modeling of LSASVs)

3.1.1.1 Period of Performance under Task T1-1

Start Date: Q3 2015

Conclusion Date: Q3 2018

Faculty lead: Mo Jamshidi

3.1.1.2 General Description of Task T1-1

LSASVs cannot be thought of as a set of independent agents since they usually interact with each other, share information and resources to collaboratively achieve given tasks. Hence, traditional methods which ignore the coupling effects between the system's components cannot be applied to these tightly coupled systems. Developing an augmented model for the overall structure of the system is also too complicated, computationally expensive, and impractical. To address these problems, we will develop novel methods using abstraction and model reduction techniques, which help to manage the complexity of the resulting model. Along with the developed models, communication protocols will also be established to interconnect these systems (machines or assets) together via a wireless network encapsulating the entire system. We will use a cloud data center to house the models and facilitate access for the LSASV systems. This will also facilitate incorporating addi- tional services into the model such as an Object Recognition Engine and a centralized world map, which will be available to all entities in the connected systems.

3.1.1.3 Progress Against Planned Objectives in Task T1-1

The objectives of this task were accomplished. Advances were made in the following areas: modeling network connected unmanned vehicles time delay simulations and experiements and state estimation. A breakdown of the achievements follows with a highlight of each major task.

The first year included modeling of UAV and UGV dynamics for robotic platforms in the initial UTSA testbed. A differential drive platform was modeled for the UGV and a quadcopter was modeled for the UAV. The particular vehicles studied were the Kobuki TurtleBot2 (UGV) and the ArDrone Bebop (UAV). A Matlab-Simulink model was developed for research and tests. Considerations of time delays were taken into account in these models. A system comprised of one UAV and one UGV has been simulated with delays between the vehicles and delays between the UGV and its controller. Because the UAV acts as a reference trajectory for the UGV, it was able to tolerate higher delays, while the UGV reached its goal. Delays were applied to the UGV controller to imitate a system with a delayed communication and computing element (e.g. cloud based con- troller). The inputs and outputs are being sent between the mobile agent and a ground station to lessen the computational burden of the robotic agent. The system simulation model utilizes an update time for the UGV controller, where the UGV model updates its inputs at discrete steps, mirroring the

conditions of a real network controlled system. This minimizes the network load and reduces the delay issue of creating a large enough update time to encompass the time delay itself. An analysis of the maximum update time has been conducted through simulation, which indicates that 1.2 seconds is the maximum update time for a stable UGV/UAV system. With the successful simulation, a second UGV was added into the system to determine the perturbation it may cause the system.

Improvements were made to the simulation system by implementing time-bounds to validate different types of system delays. Improvements were made to the Matlab-Simulink system, by adding True-time, a package for simulating realistic network delay models. Network delay models can be configured and saved as a profile, which represents the specific communication methods and operating conditions. A control system can be compared against multiple profiles for repeatable and accurate tests under defined conditions. We have made progress in research of the time delay concepts and their impact on control systems. Since Matlab-Simulink lacks the necessary timing capabilities and because our hardware testing environment uses ROS, the simulation environment was converted to python. The python testing environment consists of a UAV that directs a UGV to a point on the ground to move to.

The initial test-bed was completed in Q3-2016. For the UGV, we built a robust UGV system using Raspberry PI3, running the Ubuntu operating system and ROS. This vehicle implements the control system that we used for this task. A Parrot Bebop Drone (UAV) and a customized Kobuki TurtleBot2 (UGV) were also used for the testbed. Routines to read sensors and feed data to control algorithms were implemented in Robot Operating System (ROS). Hardware implementation of a multi-agent system with time delays using state estimation was performed as part of combined efforts with Task 1-3. Outputs of the work for this task were used in Task T1-4 for vision-based mapping research. Work transitioned from the single vehicle modeling in this task, to the multi- vehicle modeling efforts underway in Task T1-3 at the time.

3.1.1.4 Modeling Testbed Systems

The initial dynamic equations found in the literature were too general and inconvenient to apply to a specific robot in the test bed (i.e. the ErleCopter, Parrot Bebop, Kobuki TurtleBot2). By review- ing the dynamic equations and mathematical models for land rovers and quad-copters, we were able to finalize our representation of the mathematical models for our robotic agents [1]. Mathematical models for the UGVs and UAVs follow.

3.1.1.4.1 UGV Model The Kobuki Turtlebot2 (UGV) uses a differential drive to steer. Cook provides a clear approach in order to obtain and simulate differential drive robot kinemat- ics [2]. Figure 4 represents the UGV kinematic model. In the model, R is the instantaneous radius of curvature of the robot's trajectory, W is the distance between the wheels, and θ is the heading angle with respect to the "ground" frame. Kinematic equations for the UGV follow.

$$v_l = \dot{\theta}(R - \frac{W}{2}) \tag{1}$$

$$v_r = \dot{\theta}(R + \frac{W}{2}) \tag{2}$$

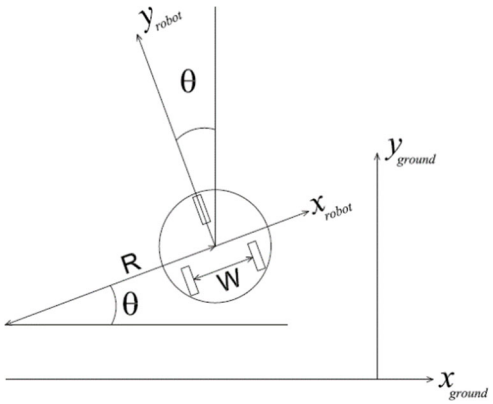


Figure 4: Representation of the TurtleBot2 Land Rover in the Coordinate System

where:

v_l: velocity of the left wheel

v_r: velocity of the right wheel

$\dot{\theta}$: angular rate

Next, the angular rate of the robot is calculated:

$$x_r - v_l = \dot{\theta}(R + \frac{W}{2} - R + \frac{W}{2})$$

$$v_r - v_l = \dot{\theta} W$$

$$\dot{\theta} = \frac{v_r - v_l}{W} \tag{3}$$

Next the instantaneous radius of curvature is calculated using (1):

$$\frac{v_l}{\dot{\theta}} = R - \frac{W}{2}$$

$$R = \frac{v_l}{\dot{\theta}} + \frac{W}{2}$$

Using (3) we have:

$$R = \frac{v_l}{\frac{v_r - v_l}{W}} + \frac{W}{2}$$

$$R = \frac{2Wv_l + W(v_r - v_l)}{2(v_r - v_l)}$$

$$R = \frac{W}{2} * \frac{v_r + v_l}{v_r - v_l} \tag{4}$$

Velocity along the robot's longitudinal axis is calculated using (3) and (4):

$$v_y = \dot{\theta} R = \frac{v_r - v_l}{W} * \frac{W}{2} * \frac{v_r + v_l}{v_r - v_l}$$

$$v_y = \frac{v_r + v_l}{2} \tag{5}$$

Representing the robot's velocity on earth coordinates we have:

$$\dot{x} = -v_y cos(90 - \theta) = -v_y(cos(90)cos(\theta) + sin(90)sin(\theta))$$

$$\dot{y} = v_y cos(\theta)$$

Using equation (5) we have:

$$\dot{x} = -\frac{v_r + v_l}{2} sin(\theta) \tag{6}$$

$$\dot{y} = \frac{v_r + v_l}{2} cos(\theta) \tag{7}$$

For this reason the control variables will be:

$$\dot{v}_r = u_1$$

$$\dot{v}_l = u_2$$

Figures 5 and 6 show results of the simulation where different speeds used to simulate the change on the steering.

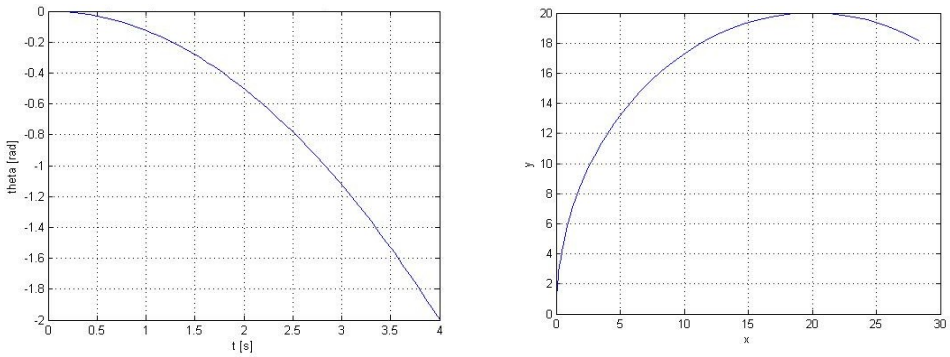


Figure 5: The effect of increasing the speed on the left wheel, (a) Angle-Longitudial axis of the robot with respect to y-axis, (b) Earth coordinate of the land rover

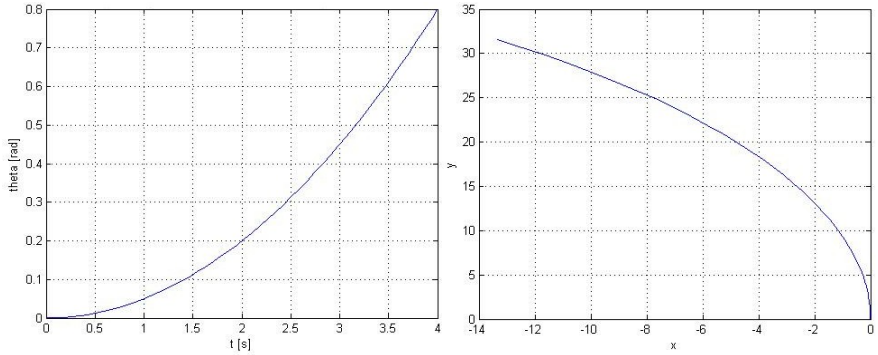


Figure 6: The effect of increasing the speed on the right wheel, (a) Angle-Longitudial axis of the robot with respect to y-axis, (b) Earth coordinate of the land rover

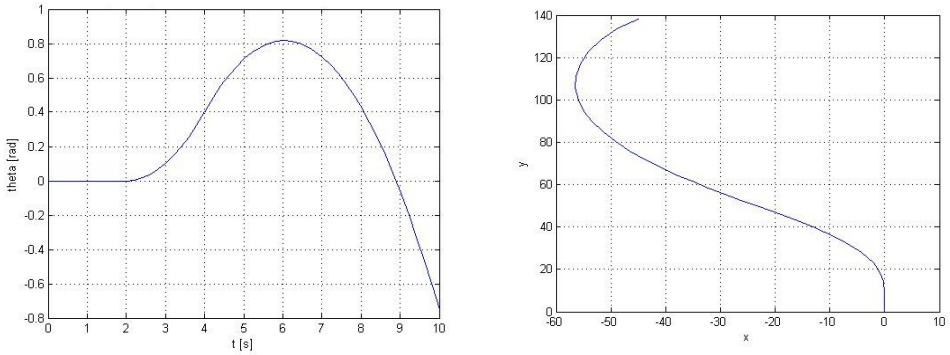


Figure 7: The effect of increasing and reducing the speed of the wheels

3.1.1.4.2 UAV Model The equations to simulate the quad-copters were obtained using Bouabdallah's work as following [3].

The following are the dynamic equations for the quad-copters:

$$\begin{bmatrix} mI_{3\times3} & 0 \\ 0 & I \end{bmatrix} \begin{bmatrix} \dot{V} \\ \dot{\omega} \end{bmatrix} + \begin{bmatrix} \omega \times mV \\ \omega \times I\omega \end{bmatrix} = \begin{bmatrix} F \\ \tau \end{bmatrix}$$

Where:

Symbol	Definition
I	inertia matrix
V	body linear speed vector
ω	body angular speed

The equations of motion for the quad-copter:

$$\begin{cases} \dot{\zeta} = v \\ m\dot{v} = RF_b \\ \dot{R} = R\hat{\omega} \\ J\dot{\omega} = -\omega \times J\omega + \tau_a \end{cases}$$

From these equations we can have the approximate model:

$$\begin{cases} \dot{\zeta} = v \\ \dot{v} = -ge_3 + R_{e3}\left(\frac{b}{m}\Sigma\Omega_i^2\right) \\ \dot{R} = R\hat{\omega} \\ I\dot{\omega} = -\omega \times I\omega - \Sigma J_r(\omega \times e_3)\Omega_i + \tau_a \end{cases}$$

Where:

Symbol	Definition
ζ	position vector
R	rotation matrix
$\hat{\omega}$	skew symmetric matrix
ϕ	roll angle
θ	pitch angle
ψ	yaw angle
Ω	rotor speed
$I_{x,y,z}$	body inertia
J_r	rotor inertia
J_m	motor inertia
J_p	propeller inertia
τ_a	torque on airframe body
b	trust factor
d	drag fator
	lever

The torque in the air-frame body along an axis is represented in the following way:

$$\tau_a = \begin{pmatrix} lb(\Omega_4^2 - \Omega_2^2) \\ lb(\Omega_3^2 - \Omega_1^2) \\ d(\Omega_2^2 + \Omega_4^2 - \Omega_1^2 - \Omega_3^2) \end{pmatrix}$$

Considering the motor inertia and a reversing gearbox with a negligible inertia, we will have:

$$J_r = J_p - J_m r$$

Then the quad-copter dynamic model is given by:

$$\begin{cases} \ddot{\phi} = \dot{\theta}\dot{\psi}\left(\frac{I_y - I_z}{I_x}\right) - \frac{J_r}{I_x}\dot{\theta}\Omega + \frac{l}{I_x}U_2 \\ \ddot{\theta} = \dot{\phi}\dot{\psi}\left(\frac{I_z - I_x}{I_y}\right) + \frac{J_r}{I_y}\dot{\phi}\Omega + \frac{l}{I_y}U_3 \\ \ddot{\psi} = \dot{\phi}\,\dot{\omega}\,\left(\frac{I_x - I_y}{I_z}\right) + \frac{1}{I_z}U_4 \\ \ddot{z} = -g + (cos\phi cos\theta)\frac{1}{m}U_1 \\ \ddot{x} = (cos\phi sin\theta cos\psi + sin\phi sin\psi)\ \frac{1}{m}U_1 \\ \ddot{y} = (cos\phi sin\theta cos\psi - sin\phi sin\psi)\frac{1}{m}U_1 \end{cases}$$

The system inputs and disturbance are given by:

$$\begin{cases} U_1 = b(\Omega_1^2 + \Omega_2^2 + \Omega_3^2 + \Omega_4^2) \\ U_2 = b(\Omega_4^2 - \Omega_2^2) \\ U_3 = b(\Omega_3^2 - \Omega_1^2) \\ U_4 = d(\Omega_2^2 + \Omega_4^2 - \Omega_1^2 - \Omega_3^2) \\ \Omega = \Omega_2 + \Omega_4 - \Omega_1 - \Omega_3 \end{cases}$$

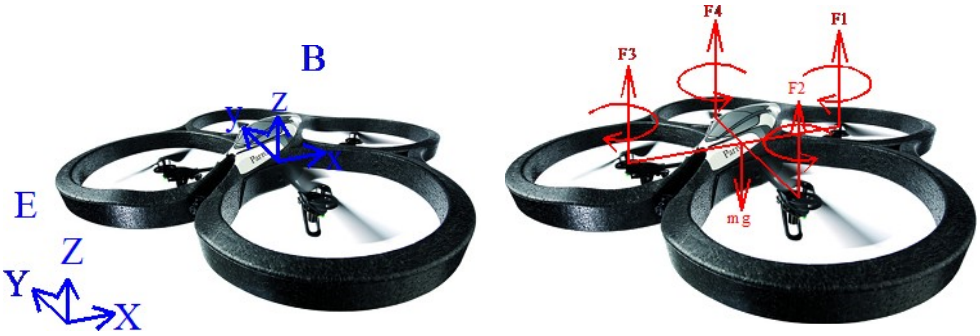


Figure 8: The system dynamics and coordinates for a quad-copter, (a) Coordinate system, (b) System dynamics and body forces.

3.1.1.4.3 Hardware implementation of a multi-agent system with time delays using state estimation In this effort a single UGV moves to a destination using the location provided by the UAV with a larger time delay using state estimation. The UGV uses state estimation whenever communication is not active between the UGV and the UAV. The state estimation is used to predict the position and orientation of the UGV based on the commanded velocities. The state estimation equations can be seen in Equations 8 - 10, where $\hat{\Theta}$ is the estimated heading angle of the UGV, $\dot{\Theta}$ is the angular velocity of the UGV, $\hat{x}$ is the estimated x position of the UGV, $\dot{x}$ is the linear velocity of the UGV, and $\hat{y}$ is the estimated y position of the UGV.

$$\hat{\Theta}(k+1) = \hat{\Theta}(k) + \dot{\Theta}(k)T, \tag{8}$$

$$\hat{x}(k+1) = \hat{x}(k) + (\dot{x}(k)T)sin(\Theta(k+1)), \tag{9}$$

$$\hat{y}(k+1) = \hat{y}(k) + (\dot{x}(k)T)cos(\Theta(k+1)), \tag{10}$$

The estimated position and oriententaion of the UGV is then used in the PID controllers update the UGV's velocities. The process of estimating the UGV's states and then updating the PID controller is repeated until communication with the UAV is restored.

The results of the state estimation can be seen in Figure 9, where the UGV was commanded to drive to two different destinations, represented by the black stars. It can be seen that the UGV struggles to reach the destination when the time delay is four seconds or larger. However, the UGV is successfully able to reach the destination with delays of four seconds and eight seconds when using state estimation.

3.1.1.4.4 Mathematical Modeling of Time Delays This section includes the mathe- matical model for Unmanned Ground Vehicles (UGV) and Unmanned Aerial Vehicles (UAV). The particular vehicles studied were the Kobuki TurtleBot2 (UGV) and the ArDrone Bebop (UAV). A Matlab-Simulink model is under development for further research and tests. Once the simulation model is ready, the model will be implemented on hardware in the testbed.

Through studying the existing time delay systems researched by Johan Nilsson in [4], we developed simulation models for the UGV and UAV. The single UAV is able to communicate location coordinate details, the X and Y coordinates as well as the angle from the original starting orientation. The UGV is capable of navigating to the target location that is provided by the UAV.

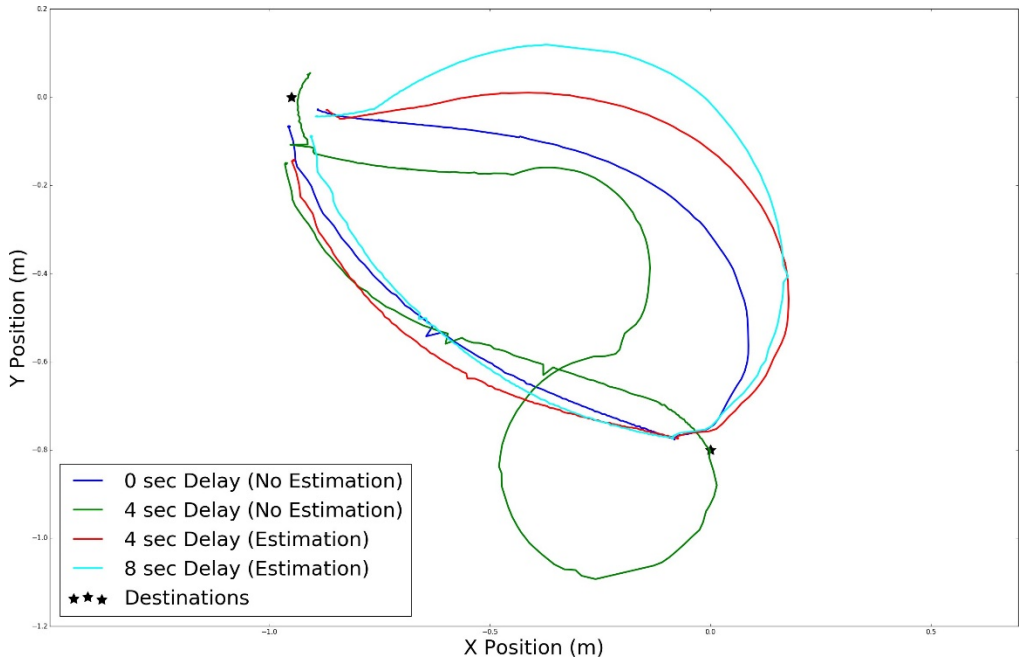


Figure 9: Hardware results of the PID controller with and without state estimation

$$
\begin{aligned}
x_1 &= x_2 \\
x_2 &= x_4 x_6 a_1 + x_4 a_2 \omega + b_1 U_2 \\
x_3 &= x_4 \\
x_4 &= x_2 x_6 a_3 + x_2 a_4 \omega + b_2 u_3 \\
x_5 &= x_6 \\
x_6 &= x_6 x_6 a_5 + b_3 u_4 \\
x_7 &= x_8 \\
x_8 &= \frac{-g + (\cos(x_1)\cos(x_3))u_1)}{m} \\
x_9 &= x_{10} \\
x_{10} &= \frac{(\cos(x_1)\sin(x_3))\sin(x_5) + \sin(x_1)\cos(x_5)u_1}{m}
\end{aligned}
\tag{11}
$$

The simulation provided a model to move the UGV to a given point. This did not require the UAV model as shown in equation 11. The UGV receives a coordinate and moves until it reaches the position within 0.3 meters and 0.05 radians. Once the margins are satisfied, the UGV sends a request signal for another position. In addition, time delays have been added simulating the wait time for wireless communication from UGV to UAV and UAV to UGV. Figures 10 - 13 represent the system with a time delay in different parts of the system. For each figure the reference trajectory is

the top left window while the trajectory taken by the rover is the top right. The bottom left area shows where the delay takes place with respect to the UAV and UGV, and the bottom right figure shows the X, Y, and θ of the vehicle with respect to the starting position of the UGV. In this first testing, we see the delays affect the system trajectory when the delays are in both communication lines. The situation must be further tested for different delay values and possible delays in actuators or controllers of the subsystems (UGV or UAV).

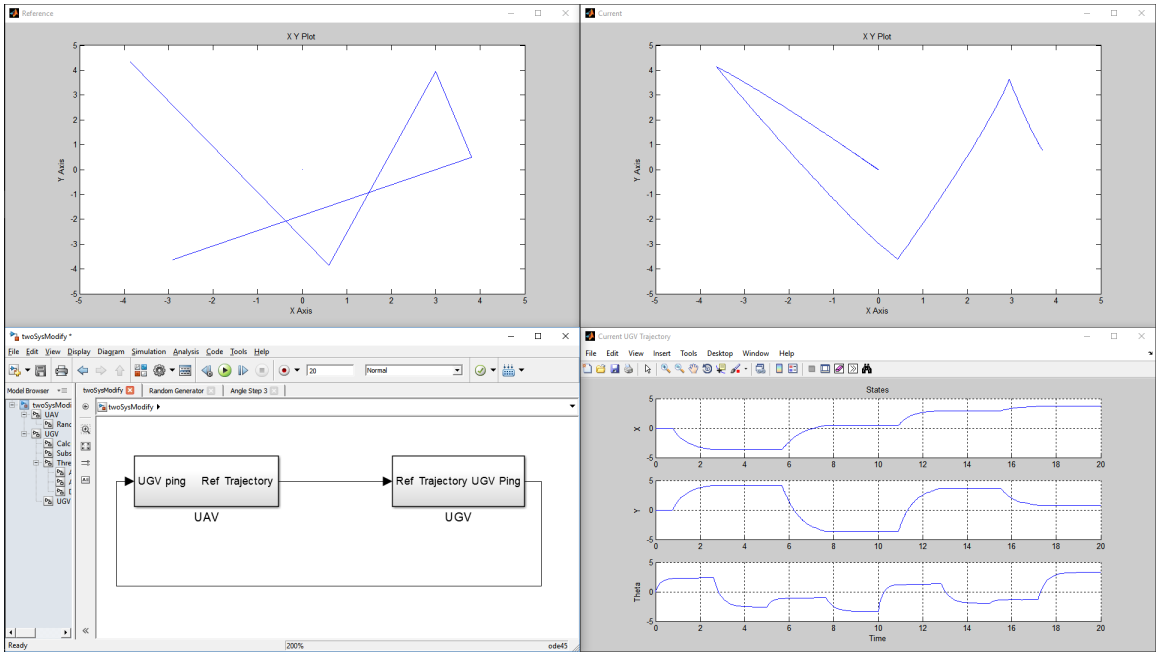


Figure 10: Simulation of UGV & UAV with No Delay.

3.1.1.5 Summary

In this task we performed studies and modeling for various aspects of single and multiple agents of a LSASV. Kinematic and dynamic models for UAV and UGV systems were modeled in Matlab. Physical characteristics and performance capabilities of the Parrot Bebop quadcopter (UAV) and Kobuki Turtlebot 2 (UGV) were taken into consideration for development of these models. With those baseline models, performance of the models were compared to physical UGVs and UAVs from the testbed. Model parameters were estimated and adjusted until the error between the two were reduced to a minimal acceptable range.

As vehicles in a LSASV can rely on network connected computation for decision making, research was performed in the various types of communication problems present within a networked system. Improvements were made into the control of the physical models of UGVs and UAVs. Simulations and physical experiments were performed to ensure that the developed models were appropriately representing the vehicles. Truetime simulations in MATLAB were used to simulate realistic network scheduling and channel effects. Python code was developed for the physical experiments. We developed Matlab True-Time simulations to model those realistic network scheduling and channel effects. In those studies we determined bounds for direct network control between systems of UAVs

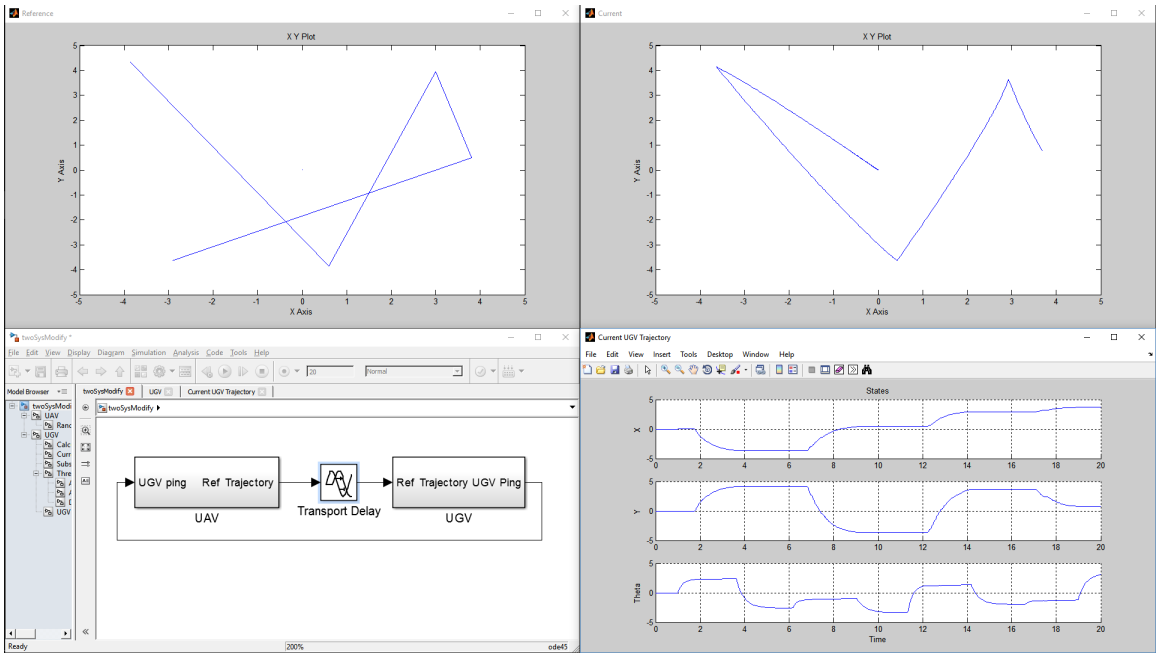


Figure 11: Simulation of UGV & UAV with a 1 second delay from UAV to UGV.

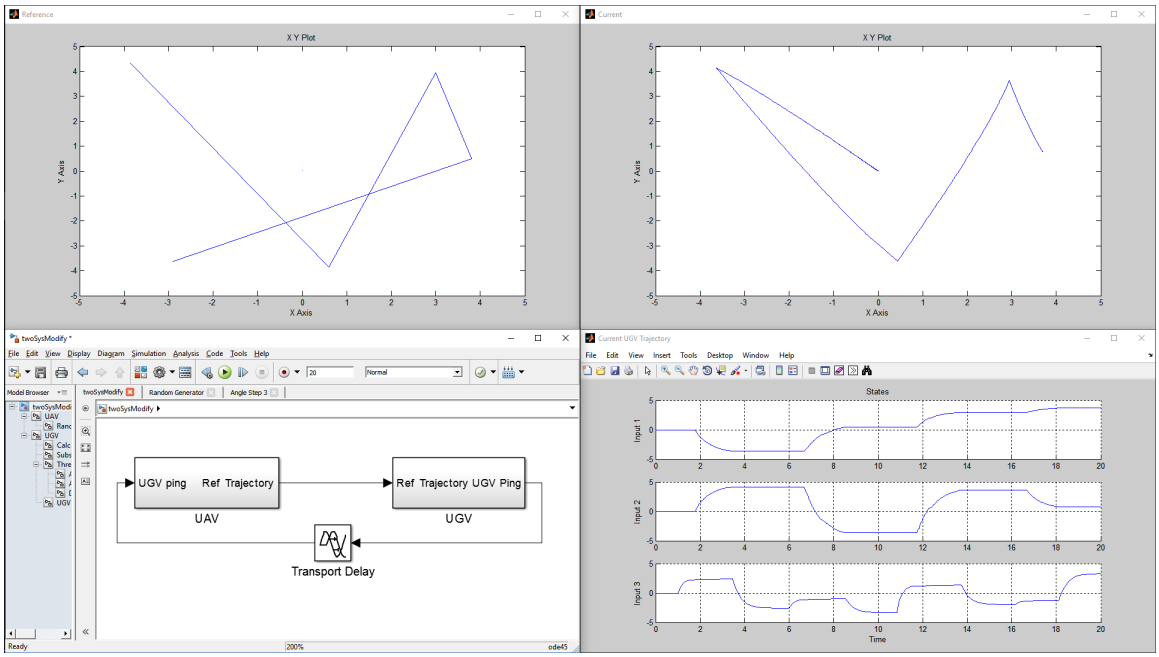


Figure 12: Simulation of UGV & UAV with a 1 second delay from UGV to UAV.

and UGVs that will help determine what can and cannot be controlled directly. Python code was developed for physical experiments that were performed to validate the simulations. State-estimation techniques were developed for mitigating time delays in a system. Simulations and

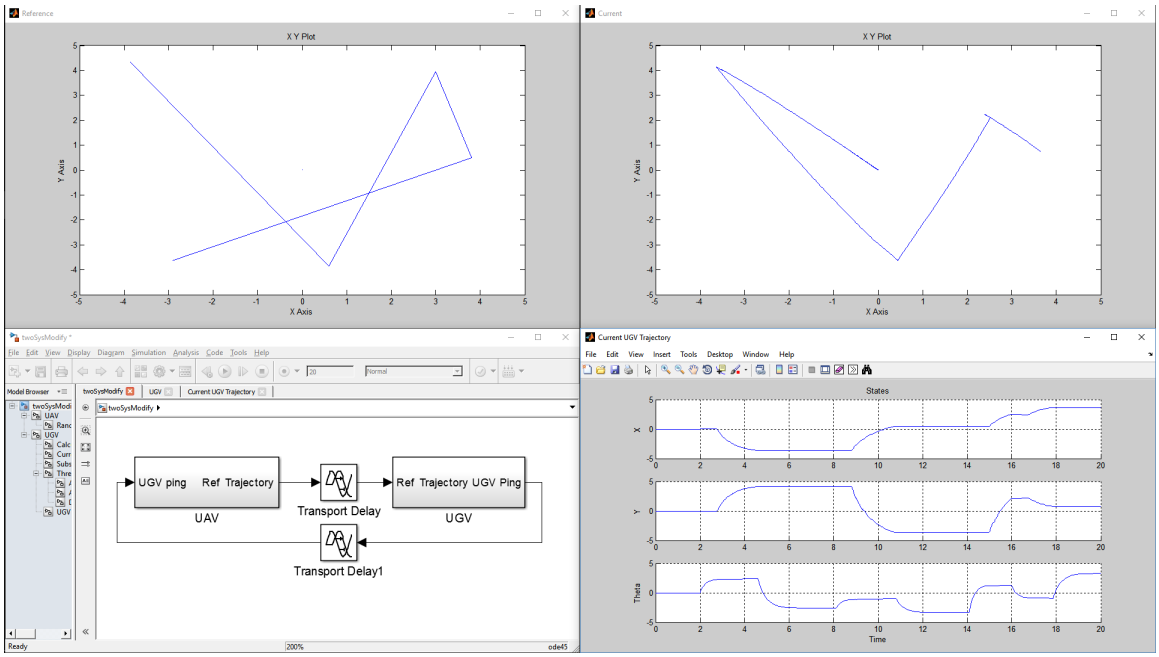


Figure 13: Simulation of UGV & UAV with a 1 second delay from UGV to UAV and from UAV to UGV.

experiments were performed to determine the model validity using multiple UGVs and a single UAV. Outcomes of this task were taken into consideration for Task T1-3 which built upon the models and developments in this task.

3.1.2 Project Progress for Task T1-2 (Data-and Knowledge-Based Modeling of LSASVs)

3.1.2.1 Period of Performance under Task T1-2

Start Date: Provide the start date

Conclusion Date: May 2021

Faculty lead: Dr. Mo Jamshidi

3.1.2.2 General Description of Task T1-2

One of the key aspects of LSASVs is the exchange and management of large datasets (numerical, textural and image) among the constituents of the system. In some applications, data computation, Bayesian networks accumulation, and security are the key pieces of information. There is a need for employment of a machine intelligence method to enable any vehicle in the system to perform in a team of vehicles. In this task, we propose that a promising approach, called Data Analytics be used. Data Analytics utilizes statistical and computational intelligence (CI) tools such as principal component analysis (PCA), clustering, fuzzy logic, neuro-computing, evolutionary algorithms, and data mining to reduce the accumulative large size of the LSASV datasets to a manageable size. We will then apply these tools to a) extract information, b) build a knowledge base using the derived

data, c) develop a cooperative teamwork framework through the consensus among members and d) eventually develop a non-parametric cooperative model for LSASVs. This task will construct a bridge between LSASVs and Data Analytics to develop reliable models for resilient cooperative control of such systems, described in Task T2-4. One of the more recent promising data analytic tools is Deep Learning. Deep learning is the broad term for the recent development and extension of neural networks in the Machine Learning community, which has allowed for state-of-the-art computational results in speech, image, and natural language processing tasks. We will also develop a hierarchical learning approach to extract high order representations from low-level data. Data analytic tools and approaches will be further used to filter data in circulation among vehicles by eliminating communication noise, anomalies, and outliers in communicated data in a system of vehicles. This task will feed directly into other Thrusts such as test and evaluation (Thrust 3) and fault diagnostics (Thrust 2). Data Analytics also plays a central role in allowing the vast amount of data created to be converted into actionable information to be used by human operators.

3.1.2.3 Progress Against Planned Objectives in Task T1-2

The objective of this task is accomplished. Some methods and models are developed for:

- Implement D* algorithm on a Kobuki Turtlebot II robot in order to create our dataset
- Multi sensor data will be fused through an Extended Kalman Filter (EKF)
- 3D object detection based on lidar data
- combining lidar, radar, and image data
- training 3D object detection system using transfer learning method and learning without forgetting method
- 3D object detection based on combination of lidar, radar, and image data
- Path planning problem for UAVs
- Abnormalities on LSASV

3.1.2.4 Technical Accomplishments in Task T1-2

In the implementation period, non-camera sensors, namely, LIDAR, IMU, GPS and wheel encoder were used to develop a data model for a mobile robot. The idea was to create a data model which takes range, position and acceleration data as input and predicts steering angle and linear velocity as output.All sensor data were combined in order to feed machine learning algorithms that were artificial neural network (ANN), support vector machine (SVM), K-nearest neighbor (KNN), AdaBoost,and decision tree. sensor fusion technique, a computer based program, which intelligently combines and gathers data from several sensors to improve the system performance. After sensor fusion, ANN, KNN, SVM, a decision tree was used in the MATLAB's statistical and machine learning toolbox.

The 3D object detector, based on PointPillars network [5], was trained to predict 3D boxes for three different classes of objects (car, pedestrian, bus) using weighting method to combine image, radar, and lidar data. Investigation of training the 3D object detector using different training

methods is an ongoing process. The improved weighting method was used to combine image, radar, and lidar data of some urban scenes provided in nuScenes dataset [6] and used the combined data as input for PointPillars network [5] to predict 3D boxes for three different classes of objects (car, pedestrian, bus). The 3D object detector was trained for 50 epochs. We used a learning rate of 2×10^{-3} and random initialization for the weights of the network (no ImageNet pretraining). For optimizing the loss function, we used the Adam optimizer. The performance of the 3D object detector based on PointPillars [5] was compared to MonoDIS [7] which is a top 3D image detector. Results show that Pointpillars [5] has a better performance. Transfer learning was exploited to train our 3D object detector in order to improve its performance. Our 3D object detector is based on PointPillars network [5] to predict 3D boxes for three different classes of objects (car, pedestrian, bus) using weighting method to combine image, radar, and LIDAR data. A 3D object detector was trained using random initialization for the weights of the network. So, we didn't use ImageNet pretraining, or any other pretraining in the training process of the network. Then, we compared the performance of our 3D object detector based on PointPillars to MonoDIS [7] which is a top 3D image detector. Although results show that our 3D object detector has a better performance, it still should be improved to have a better accuracy and more stable results.

To use transfer learning, first we trained the network on Waymo open dataset [8] which includes 1950 sequences, each 20 seconds long with different camera and LIDAR sensors producing data sampled at 10Hz. Then, we used the weights of the trained model as initial weights for the network to be trained on nuScenes dataset [6]. Results show that utilizing transfer learning in the second experiment slightly improves the performance.

We utilized transfer learning as a feature extraction method to train our 3D object detector in order to improve its performance. Our 3D object detector is based on PointPillars network [5] to predict 3D boxes for three different classes of objects (car, pedestrian, bus) using weighting method to combine image, radar, and LIDAR data.

To train our 3D object detector utilizing transfer learning as a feature extraction method, we froze first few hidden layers of the network after training it on Waymo open dataset [8]. Then we started training it with a learning rate of 2×10^{-4} that linearly was increased to 2×10^{-3} in the first 5 epochs. The rest of total 50 epochs was then trained with a decreased learning rate of 2×10^{-5}. We used the Adam optimizer to optimize the loss function. The performance of the 3D object detector using feature extraction method was then compared to its performance we achieved in previous experiment explained in the previous report. Results show that using transfer learning as a feature extraction method slightly improves the performance of the network specifically in car and pedestrian classes that have more annotated objects in the dataset.

We utilized learning without forgetting method to train and test our 3D object detector and compared the results with the results of the model when using transfer learning as a fine-tuning method, and as a feature extraction method. We expanded our 3D object detector which is based on PointPillars network [5] to predict 3D boxes for five different classes of objects (car, pedestrian, bus, truck, bicycle). The model uses weighting method to combine image, radar, and LIDAR data. The network has some shared parameters (θ_s) and some task-specific parameters (θ_o). To utilize learning without forgetting method to train the model that is going to classify 2 more classes of objects (truck, bicycle), we first used our previously trained model on new input data containing objects of the 2 new classes and recorded the results. Then we added 2 nodes to the output layer for the 2 new classes of objects, and fully connected them to the layer before the output layer. The weights for these nodes were randomly initialized. These weights are new task-specific parameters

(θ_n) added for each new class of objects. Then we trained the network with new data to minimize the loss function for all classes of objects using stochastic gradient decent. In the first stage of training, we froze shared parameters (θ_s) and the task-specific parameters for the old classes of objects (θ_o), and then trained the task-specific parameters for the new classes of objects (θ_n) to convergence. This step is called warm-up step [9]. Then we trained all parameters ((θ_s), (θ_o), (θ_n)) together until convergence. This step is called joint-optimized step [9]. The performance of the 3D object detector using learning without forgetting method trained with new data to classify 5 classes of objects (car, pedestrian, bus, truck, bicycle) was then compared to its performance when using feature extraction method and fine-tuning method (explained in previous reports). Results show that the performance of the network to detect old classes of objects (car, pedestrian, bus) is almost the same for learning without forgetting and feature extraction methods. But it is slightly lower for fine-tuning method. However, the performance of the network to detect new classes of objects (truck, bicycle) is slightly better for learning without forgetting method compared to when using fine-tuning method and is significantly enhanced compared to when using feature extraction method. The reason for lower performance of the network for the three classes of bus, truck, and bicycle, is that they have less annotated objects in the dataset than the car and pedestrian classes.

For the path planning of UAVs and the abnormalities on LSASV; first, ground truth data that was created from images which were obtained from UAVs. Second, the data-set that is vital for relative pose estimation is pre-processed by PCA. Then, we implemented a comprehensive deployment method targeted at allowing implementation of complex deep learning algorithms by using LSTM and the data-set to get acceptable pose estimation for UAVs. LSTM network had been implemented to get an accurate tracking prediction to navigate target under delay or GPS issues. In addition to path prediction network, PCA (Principal Component Analysis) which is a pre-processing tool was used to get better results by eliminating the faulty and missing data in large data-sets. Network delay between UAVs and UGVs is caused missing data that increase the error for path estimation. With pre-processing estimation accuracy is increased however long computing time. As an additional model improvement, we have worked on the pre-processing method instead of PCA. LSTM network with the feature removing layers is created. Improving the LSTM model with these layers decrease the training time and increase the training performance. Before the training process, removing the features which does not affect the training performance but affects the taring time will increasingly decrease the processing time. Unnecessary features are detected by discovering the constant features. The algorithm which compares the max and min value of the features is created. When a row has same maximum and minimum value, this undesirably impacts the training performance.

3.1.3 Project Progress for Task T1-3 (Qualitative and Quantitative Analysis of LSASVs)

3.1.3.1 Period of Performance under Task T1-3

Start Date: Q3 2015

Conclusion Date: Q1 2022

Faculty lead: Mo Jamshidi

3.1.3.2 General Description of Task T1-3

Followed by modeling LSASVs, the next step is to analyze the resulting models to investigate the properties and emerging behaviors of LSASVs. Traditional concepts developed for single agents are no longer valid for LSASVs and new notions of stability, controllability, observability, and reachability are required. For the stability analysis of LSASVs, we will focus on the input-to- state stability (ISS) of the system and will employ small gain theory, whose proven compositional properties ease the development of a modular scalable stability analysis method for interconnected large scale systems. We will also formulate the controllability of the LSASVs to realize how to drive the vehicles toward desired positions in the presence of tight interactions with other vehicles. This will be carried out by exploring the reachable space of the abstracted space and investigating simulation-based structural properties of the system. Communication protocols will be a part of this investigation and maximum communication delays will be determined to guarantee system stability in coordination with Thrust 2.

3.1.3.3 Progress Against Planned Objectives in Task T1-3

The primary objectives of this task were accomplished, resulting in a simulation environment for training a vehicle to autonomously navigate using deep reinforcement learning algorithms. The single agent ground rover was also modeled in the software. The major achievements of each objective are highlighted below.

3.1.3.3.1 Development of the Multi-Agent Testbed This task built upon the models generated in Task T1-1. As such some models are not repeated in this report.

The single ground robot was studied in Task T1-1 and traditional linearization techniques have not produced a system which is controllable. Nonlinear approaches were analyzed to find a viable method for controllability. In order to compare the model and real data, the statistical programming language R was used to create a graph of spatial coordinates in real-time as the robot moved around. This allowed us to verify the model against actual data from the robot and find better ways to improve upon it.

The single agent ground rover was modeled in the software with the corresponding data shown in Figure 14. The real data in Figure 15 shows a similar circular path, but the sliding of the system produces a drag effect which skews the path over time. The model does correctly emulate the system but without accounting for real world perturbations, such as the wheel slippage which we see in a real-time data set. A comparison between modeled data and the actual data from a Kobuki ground rover is illustrated in the following figures, with an angular velocity of 0.33 m/s and a linear velocity of 0.5 m/s.

For evaluation of the UGV, the vehicle needed to be controlled successfully to move from an initial point to a goal location. A simple state feedback control has been devised to advance the system further. Since the Kobuki Turtlebots UGVs are differential drive robots, a simple 3-stage state feedback controller was derived to move them, removing the problem with control of a non-holonomic system. Normally problems arise because differential drives cannot slide left or right (non-holonomic constraint). They can only move forwards and back, or rotate with angular motion. The control of the UGVs has three separate stages which run sequentially as the previous control stage is finished. These stages are listed below:

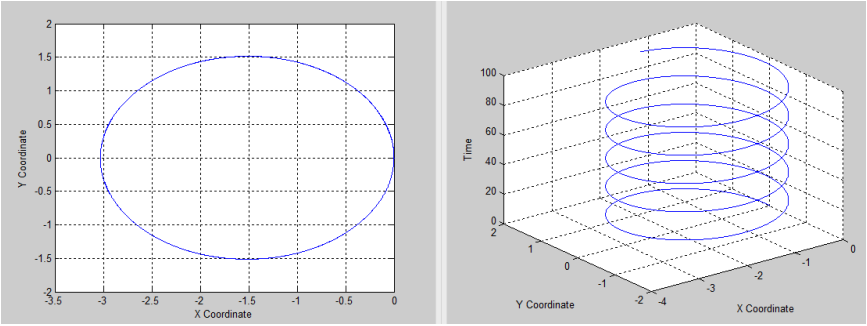


Figure 14: Modeled Kobuki Ground Rover

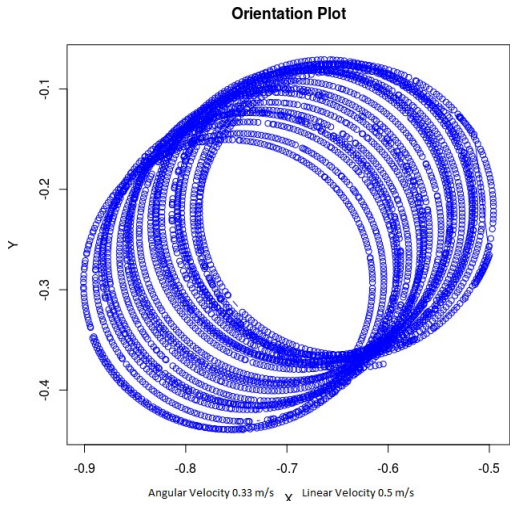


Figure 15: Equilibrium position of the Formation

1. The UGV moves in an angular motion to face the desired point.
2. The UGV then traverses the linear distance until it reaches the desired coordinate.
3. Finally, the UGV once again moves in an angular motion to face the desired object of interest.

The PID controller was under development for the multi-agent testbed. The goal for the PID controller was to have the agents better approach their target locations while also reducing the pose error. The previous implementation utilized a PI controller which resulted in movements of the agents with a noticeable jerk, which can be expected from a controller which does not take into account the rate change of the error. The values for proportional (K_p), integral (K_i), and derivative (K_d) were calculated through trial and error. For calculating the position of agent, error in position is multiplied separately with K_p, K_i, K_d and added together. This process is carried out for all the agents. Similarly the orientation of all agents are calculated the same way by multiplying and adding the error in θ with K_p, K_i, K_d.

The vehicle selected for the research platform is based on a single child vehicle Figure 16. The overall dimensions of the vehicle: 46 (L) × 31 (W) × 31 (H) and weight Capacity: 65 lbs. This vehicle was selected for its payload capacity, built-in gearbox and provided steering motor. The payload capacity of the vehicle allows for customization of the space with mounting brackets, and custom-built supports and panels.

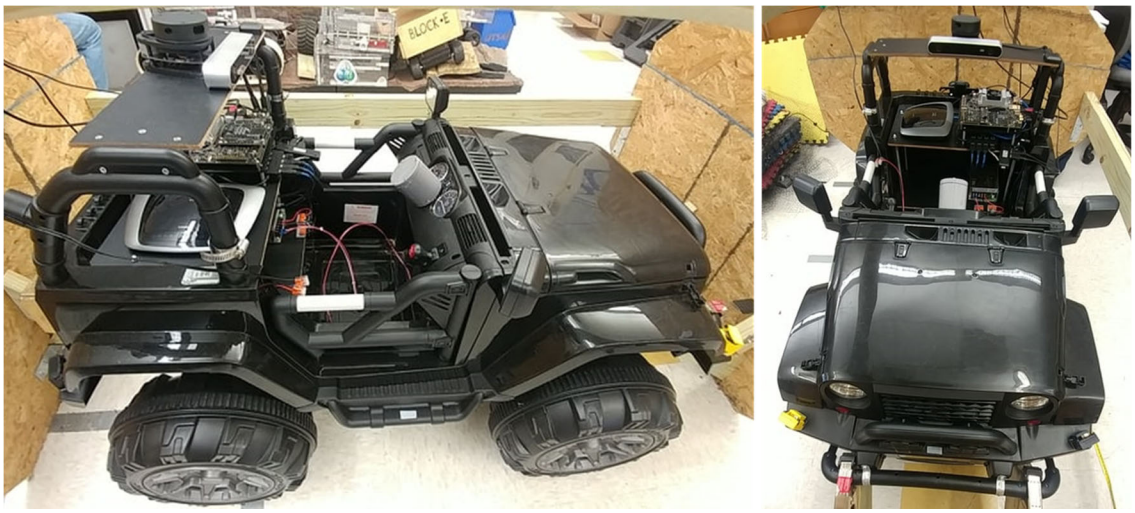


Figure 16: Views of the robot URDF model in RVIZ

Sensor Configuration of UGV The sensors initially selected for the project are a Global Positioning System (GPS), Inertial Measurement Unit (IMU), ultrasonic sensor, time of flight distance sensor, Light Detection and Ranging (LIDAR), stereo vision camera, and a standard web camera. GPS helps provide the location of the vehicle using up to 22 satellites and 66 channels. These pro- vides enough information about the rough location of the vehicle when testing in outdoor scenarios. The BNO055 IMU is 9-DOF (Degree of Freedom) sensor that provides absolute orientation, angular velocity, acceleration, magnetic field strength, linear acceleration, gravity vectors and temperature for compensation. The ultrasonic sensor is a high noise tolerance and narrow beam sonar used as obstacle detector. The time of flight distance sensor uses an invisible laser source and a matching

sensor to detect the distance of a surface (obstacle) located in front of it. The RPLIDAR-A1 LIDAR is a 360 degree Laser Scanner capable of scanning 360 degree environments to built maps or 3D models. The ZED Stereo vision camera provides capability for depth perception, positional tracking and 3D mapping. The web camera is a low cost solution for image or video acquisition to perform basic image processing.

Computing Platform of UGV The main processing unit in the vehicle is a NVIDIA Jetson TX2 development board. The main objective is to take advantage of the NVIDIA GPU capabilities either for image processing or running machine learning algorithms. Arduino boards are initially selected to interface with some of the sensors that required either an I2C bus or an UART port for communication. Most of the low level sensors (e.g. GPS, IMU, Ultrasound, or Time of Flight Distance sensor) have libraries written for Arduino. A WiFi Cisco router is installed in the vehicle to provide a mobile WiFi connection to the vehicle. This enables a simple and more secure method of providing a communication channel to the user during research and development of the vehicle. The low-level communication between sensors and microcontrollers will be performed either through a UART port or an I2C bus. The initial approach for communication between the microcontrollers and the Jetson board is through a CAN bus. For these purpose the MCP2515 CAN Bus Module is considered. The Jetson TX2 board has an installation of Linux for Tegra (L4T) 28.1 using a program called Jetpack 3.1. This version of L4T is based on Ubuntu Linux 16.04. The Kinetic Kame version of the Robot Operating System (ROS) is installed on the computer. ROS provides all the software libraries required to interface the Jetson TX2 with the Arduino boards and sensors such as the RPLIDAR-A1 and the ZED Stereo camera.

Kinematic and Dynamic Models of UGV The dynamic model of the the vehicle can be represented using the Ackerman steering model as shown in Figure 17. For this purpose, it can be considered that the two front wheels turn in a slightly differential manner. In this way, the center of rotation of the vehicle can be found doing only a kinematic analysis. This type of approach makes the analysis of the steering mechanism look the same as that of a bicycle model.

The kinematic bicycle model can be represented by the following nonlinear continuous time equations.

$$\dot{x} = v\cos(\phi + \beta) \tag{12}$$

$$\dot{y} = v\sin(\phi + \beta) \tag{13}$$

$$\dot{\phi} = \frac{v}{l_r}\sin(\beta) \tag{14}$$

$$\dot{v} = a \tag{15}$$

$$\beta = \tan^{-1}\left(\frac{l_r}{l_f + l_r}\tan(\delta_f)\right) \tag{16}$$

where:

- x and y : coordinates of the center of mass in the inertial frame (X, Y)
- ϕ : inertial heading
- v : speed of the vehicle
- l_f : distance from the center of mass of the vehicle to the front axle

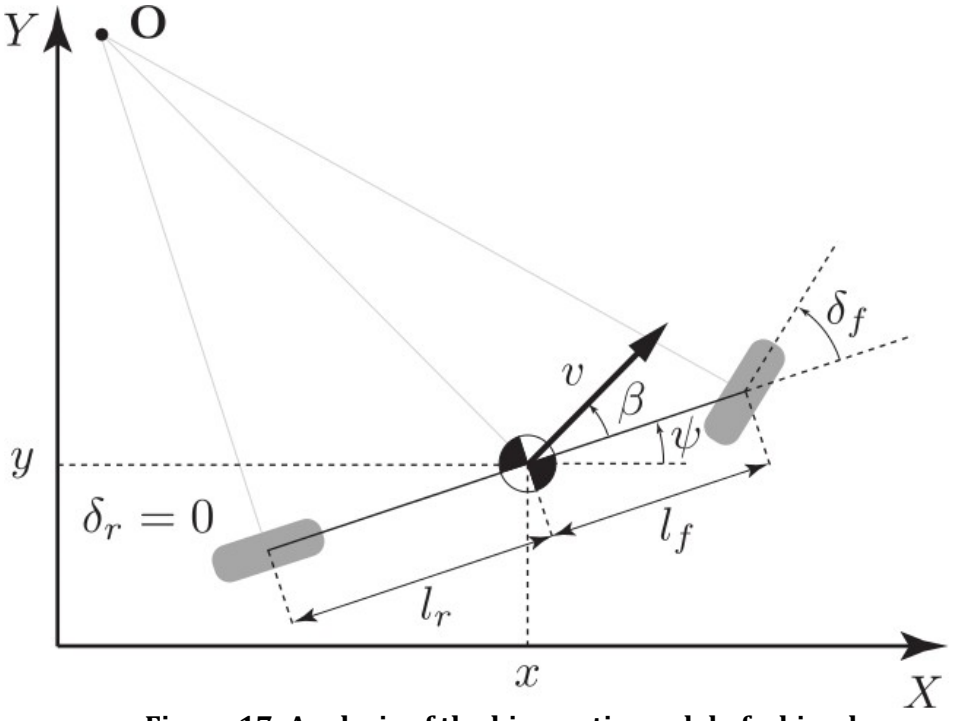


Figure 17: Analysis of the kinematic model of a bicycle

- l_r : distance from the center of mass of the vehicle to the rear axle
- β : angle of the velocity of the center of mass with respect to the longitudinal axis of the car
- a : acceleration of the center of mass
- δ_f: front steering angle
- $\delta_r = 0$: rear steering angle. Rear wheels in the platform cannot be steered.

The system model described above has δ_f and a as control inputs. The following equations describe the dynamic bicycle model.

$$\ddot{x} = \dot{\phi}\dot{y} + a_x \tag{17}$$

$$\ddot{y} = -\dot{\phi}\dot{x} + \frac{2}{m}(F_{c,f}\cos\delta_f + F_{c,r}) \tag{18}$$

$$\ddot{\phi} = \frac{2}{I_z}(l_f F_{c,f} - l_r F_{c,r}) \tag{19}$$

$$\dot{X} = \dot{x}\cos\phi - \dot{y}\sin\phi \tag{20}$$

$$\dot{Y} = \dot{x}\sin\phi + \dot{y}\cos\phi \tag{21}$$

where:

- $\dot{x}$: longitudinal speed
- $y^{\cdot}$: lateral speed
- $\dot{\phi}$: yaw rate
- I_z: yaw inertia
- m: mass
- $F_{c,f}$: lateral tire forces at the front wheel
- $F_{c,r}$: lateral tire forces at the rear wheel

The definition of the linear tire model is given by:

$$F_{c,i} = -C_{\alpha_i}\alpha_i \tag{22}$$

where:

- $i \in \{f, r\}$
- α_i : tire slip angle
- C_{α_i} : tire cornering stiffness

ROS tf2 URDF Model of Vehicle An important aspect of an autonomous vehicle is con- figuring the motor controls and sensors to have the vehicle identify its environment. However, to prevent errors in the physical environment, it would be efficient to create a simulation with a robot model using RVIZ. This method of simulating robot models in a virtual environment is known as URDF modeling. URDF modeling uses XML code to transfer robot models into a simulated environment. For the RVIZ software to recognize a package containing URDF files, a directory is placed in the ROS workspace (catkin_ws) that will allow the RVIZ software to detect the URDF package. Inside the package is a URDF file, which can be configured in XML to create the geom- etry, visuals, and physics of the robot model. To keep the Jeep model simple, a rectangular base named base link connects the wheels to establish a skeleton of links and joints. With regards to the wheel, four cylinders were created and named with regards to its position in the vehicle such as Front_wheel_left, Back_Wheel_Right, etc. Each wheel has a joint that determines the x (roll), y (pitch) and z (yaw) axes. The Z-axis allows the wheel to pivot left and right for turning while the Y-axis rolls the wheel forward or backwards. The Y-axis also allows for the vehicle to travel along Y coordinates. The x-axis has no significance with how the wheels move but it allows the vehicle to travel along X coordinates. Once these joints are combined together through links, this creates the skeleton of the vehicle and allow for movement. Each joint in the wheel can be configured to move the base link as one whole group. Figure 18 shows the resulting robot model in RVIZ.

Evaluation of Methods for UAVs: The quadcopter has been studied for controllability and observability as a linearized system. The model of the quadcopter is originally highly nonlinear due to the trigonometric relationship between the pitch, roll, and yaw angles. This presents a difficult

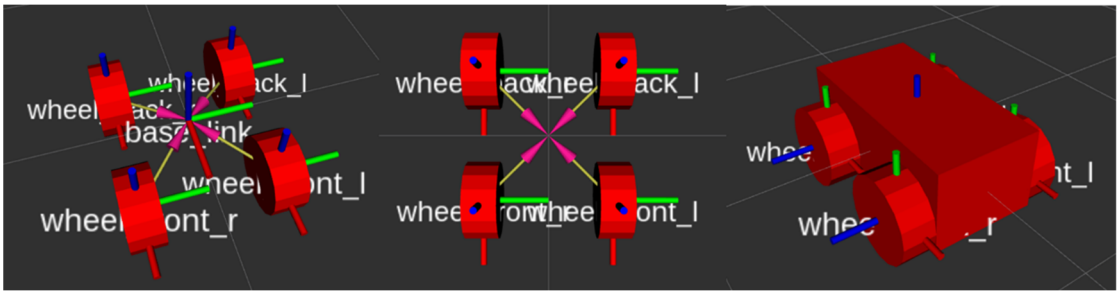


Figure 18: Views of the robot URDF model in RVIZ: left two images without the main chassis, right image with the main chassis.

problem in controls, but we first started observing its behaviors by linearizing through the small angle approximation. If the perturbations to the angles of the quadcopter are sufficiently small, the mathematical model can be reduced to a linear system based around an operating point. The operating point chosen in this case was a hovering position where the thrust of the quadcopter is essentially canceling out the force of gravity to keep the quadcopter upright. The hovering UAV was proven to both be controllable and observable. Detail on the analysis is provided below.

Consider the quadcopter with 12 different states:

- ϕ, θ, ψ: The pitch, roll, and yaw angles respectively
- x, y, z: The coordinates with respect to the earth's reference frame
- u, v, w: The linear velocities in the UAV's inertial frame
- p, q, r: The angular velocities in the UAV's frame

Also, there are 4 inputs to the quadcopter that control the value of each of the 12 states:

- u_1 = Vertical Thrust
- u_2 = Angular motion along the X direction
- u_3 = Angular motion along the Y direction
- u_4 = Angular motion along the Z direction

Using these 12 states of the quadcopter and the small angle approximation, a linear state-space system can be created to map the inputs to the outputs of the system, which are the same as the states in this case. This state space model is in the form

$$\dot{x} = Ax + Bu$$

$$y = Cx$$

where x consists of the 12 states and y is the output.

Since this linear system can be controlled, the first control algorithm attempted was a linear quadratic regulator (LQR). This control paradigm finds the optimal control value, K, which can be applied to the desired states producing the correct inputs to drive the quadcopter. The system is shown in a Simulink block diagram in figure 19.

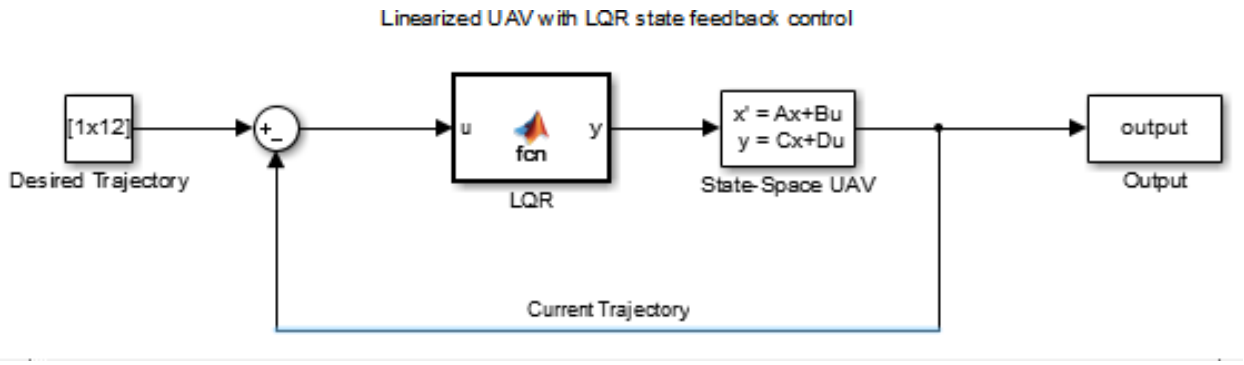


Figure 19: Block Diagram of LQR state feedback control of the UAV

Stabilization of the UAV centered around a hover point is only really sufficient for the system model, where the UAV acts as a supervisor to the UGVs by searching for points of interest to send the UGVs towards. With LQR, the UAV can also be given a target destination to move to in a 3-dimensional space. This will enhance the capabilities of the system beyond its current state in allowing for motion in the UAV.

3.1.3.3.2 State Estimation Analytical results were deduced for the experiments performed using state estimation to predict states with time delays. The modified state estimation showed the performance of the model to perform satisfactorily upto 15 seconds of time delay, meaning each update of the state of the system was received only every 15 seconds. The system fails to converge at larger delays.

To test the setup, an experiment with a *Kobuki* robot was made to maneuver back and forth between two pre-defined points $P1 = (0.0, 2.0)$ and $P2 = (-2.0, -0.5)$ respectively, simulating a UGV. The choice of these points are arbitrary, but were constrained by the field of view of the camera in our experimentation. An overhead camera was mounted on a ceiling, simulating a vision based UAV. Different set of delays were incorporated into the state information available to the UGV, which performed navigation using a PID controller and state estimation.

Figures 20 shows the trajectories taken by the UGV during these tests with communication of state information delayed with 0 (*ideal*), 1, 4, 10, 15, 20 seconds respectively. It should be noticed that with 15 seconds of delay, the UGV is still able to traverse, but with sub-par performance. A delay of 20 seconds shows complete failure of the system to recover from losing a signal, since it goes out of detection range of the UAV camera.

The metric of the comparison of the performance of the state estimation used on UGV across each delay was calculated using a mean absolute error for each position point of the trajectory. The error in this case was the measure of the outlying position point with respect to the vector joining

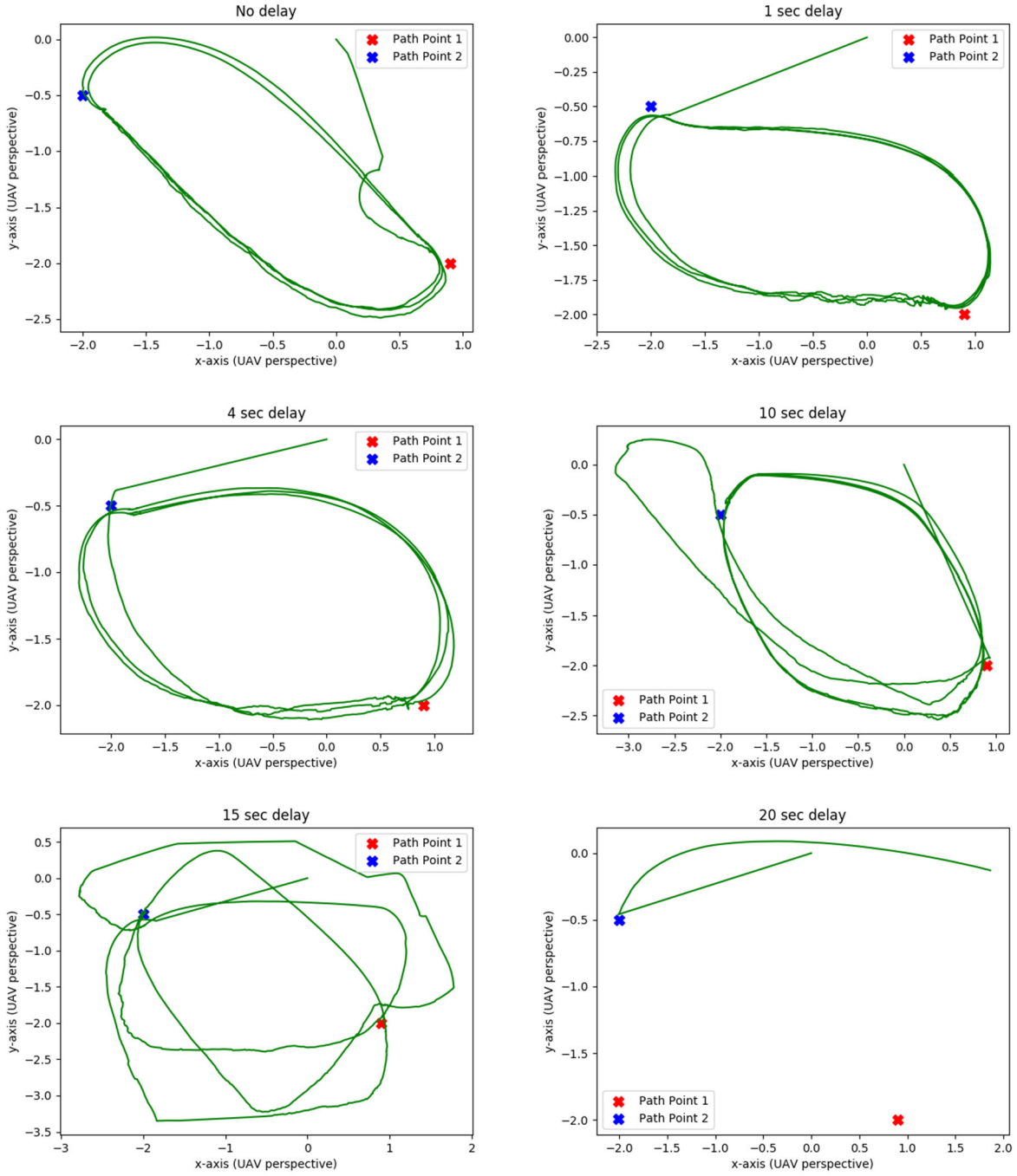


Figure 20: Trajectories taken by UGV to maneuver between P 1 and P 2 with different communica- tion delays.

$P1 = (x_1, y_1)$ and $P2 = (x_2, y_2)$. If $P = (x, y)$ is a point on the trajectory, the error e_P is computed

as

$$e_P = \| y - y_1 - \frac{y_2 \quad y_1}{x_2 - x_1}(x - x_1) \| \tag{23}$$

The figure shows the distribution of such errors for each delay, and it can be observed that the general error increases with the increase of delay.

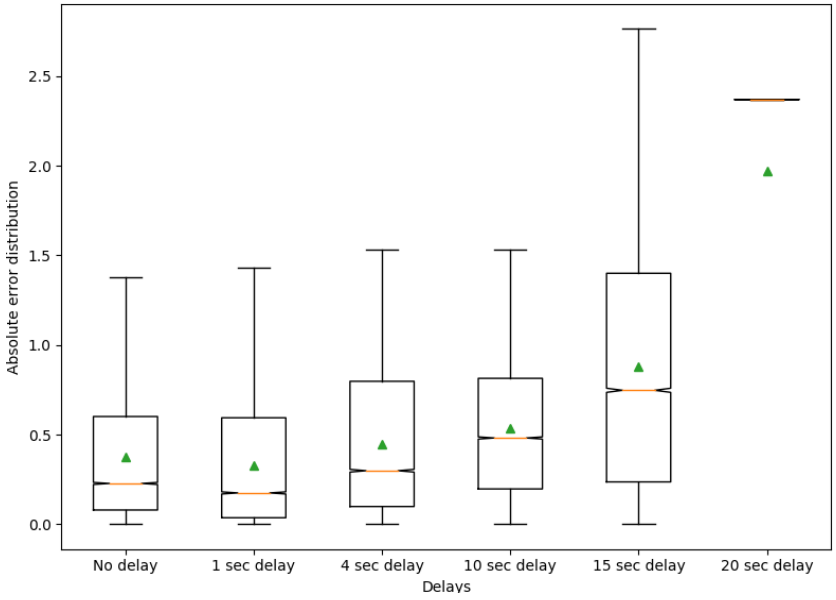


Figure 21: Error bar plot showing error distribution for different communication delays.

Software Implementation of Position Estimation of UGV

In this section, The Long Short-Term Memory (LSTM) network is implemented to estimate the position of the UGV. The inertial coordinate (x, y, z) is used for navigation and state estimate. The UGV had been controlled manually starting from a random coordinate. Data had been taken from the UGV to provide estimated reliable three-dimensional positioning with 8478 data points for each dimension. These data were used as input data into the Long Short-Term Memory (LSTM) network thath consist of a deep feature extraction module. LSTM network can solve the long-term and short-term dependency problems. The hidden layer of LSTM is called memory cells, a fundamental of the LSTM. There are three gates such as input gates, output gates and forget gates which allow to update state and add/remove information to the memory cell. The updating the state of cell feature improves calculation of the output of LSTM network and accuracy of the network.

Figure 22 illustrates the pose of manually controlled UGV pattern of driving. Data-set was created by using three-dimensional position data that is recorded from the UGV trajectory. With the dataset, 70% of the data was used to train the network and the remaining 30 % of the data was estimated.

The main idea is to make the trajectory of a UGV stable under longer delay. Data analytic methods are used to get estimation with better performance. Figure 26 illustrates the used metric

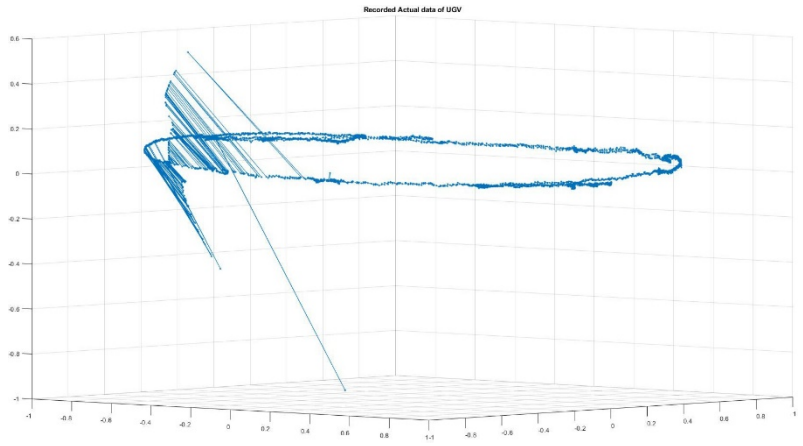


Figure 22: Recorded pose of manually controlled UGV pattern of driving

rules to evaluate estimation performance for position estimation in terms of Root Mean Square Error (RMSE) and Mean Absolute Percentage Error (MAPE). We had implemented all three axis separately and then combined them. In Figure 23 and 24, the results depict that the proposed model estimates inertial coordinates accurately.

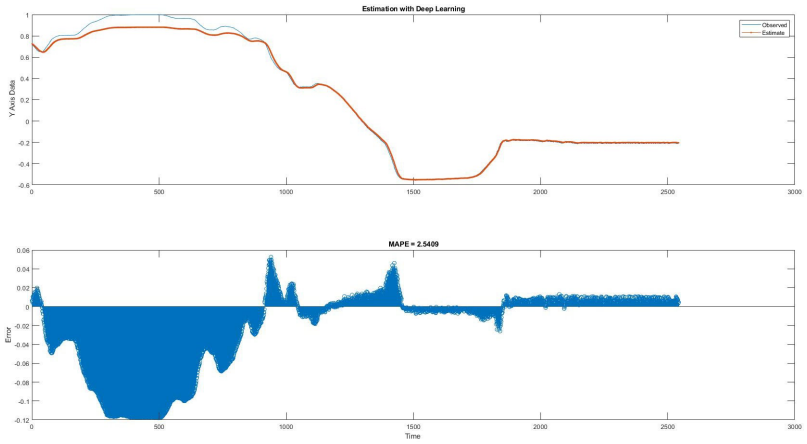


Figure 23: Y-Axis Estimation

As shown in Figure 25, the red color represents the observed/actual position and the blue color presents the estimated position. Table 1 shows the examination of the state estimation error in terms of MAPEs for different axis, illustrating that they range between 0.32% and 2.54%. These estimation performances show that the UGV is able to navigate to a target under delay or communicate problems between the UAV and the UGV.

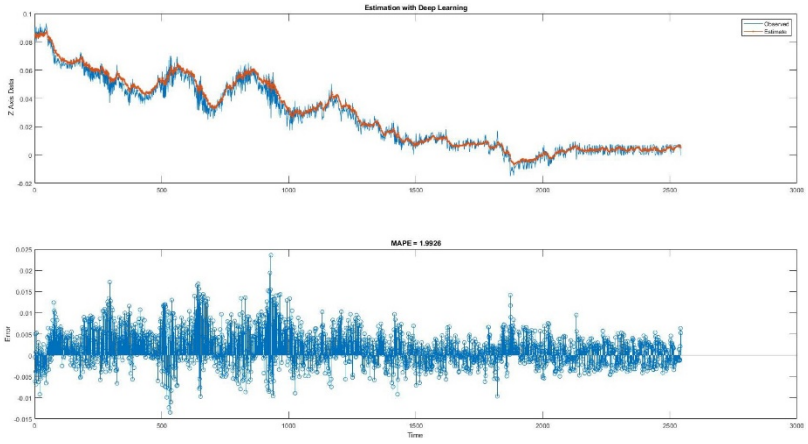


Figure 24: Z-Axis Estimation

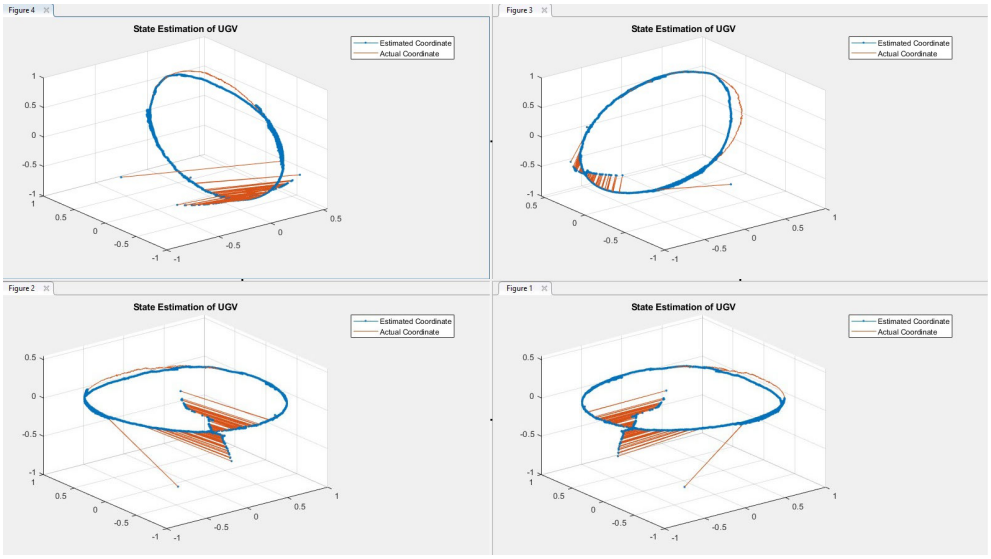


Figure 25: Observed and Estimated

Table 1: Estimation Performances

Error Metric	X Axis	Y Axis	Z Axis	Average
RMSE	0.0096	0.0191	0.0042	0.0109
MAPE	0.325	2.54	1.99	1.61

3.1.3.3.3 Hardware Description and Dynamic Model of Self Learning UGV ***UGV Specifications and Dimensions*** The vehicle selected for the research platform is based on a

Metric	Definition	Equation
MAPE	Mean absolute percentage error	$MAPE = (\sum_{t=1}^{N} \lvert(X(t)-\hat{X}(t))/X(t)\rvert)/N$
RMSE	Root mean square error	$RMSE = \sqrt{(\sum_{t=1}^{N} [X(t)-\hat{X}(t)]^2)/(N-1)}$

Figure 26: Metric Rules

traditional RC car, i.e., Ackermann style. The dimensions of the UGV are: Length = 10.4 inches, width = 6.2 inches, and height = 4.7 inches. It comes with two 4.8 V, 700 mAh rechargeable batteries. Batteries can last up to 30 min of continuous usage, and up to 60 min to fully recharge. It can reach a maximum speed of 15 km/h. and ground, grass, and sand are applicable fields where this UGV can navigate.

Sensor Configuration of UGV The sensors initially selected for this project are, Inertial Measurement Unit (IMU), time of flight distance sensor, infrared laser scan localization. The BNO055 IMU is a 9-DOF (Degree of Freedom) sensor that provides absolute orientation, angular velocity, acceleration, magnetic field strength, linear acceleration, gravity vectors and temperature for compensation. The time of flight distance sensor uses an invisible laser source and a matching sensor to detect the distance of a surface (obstacle) located in front of it.

Computing Platform of UGV The main processing unit in the vehicle is a raspberry Pi 3 board. The main idea is to incorporate all modules into the Raspberry Pi 3. Arduino boards are initially selected to interface with some of the sensors that required either an I2C bus or an UART port for communication. The low-level communication between sensors and microcontrollers will be performed either through an UART port or an I2C bus. The Kinetic Kame version of the ROS is installed on the computer. ROS provides all the software libraries required to interface with all devices mentioned above.

ROS Network System ROS is a powerful tool when it comes to communication between systems. ROS needs a master in order to operate and manage communication between nodes. Since the physical system's microcmoputer is not powerful enough to process and train the collected data with the deep neural networks, a transaction of the collected data by the physical model to the Graphics Processing Unit (GPU) must be made in order to process and generate a new model efficiently. To achieve this idea, the GPU is going to be set as the master on the network and the physical system as the slave.

Figure 27, demonstrates a high level communication network system between physical system and GPU virtual system.

Data from both localization and distance laser sensors are collected on the physical system. These data are important to drive the UGV, given that initial model for the vehicle is already imported from the virtual simulated version of the robot. With real-time readings, a new model can be generated that can adapt to the new physical system. Discrepancies between world map geometries of the simulated and real-world environments, a localization distance normalization, is performed that uses a ratio relative to the size of the map. The distance magnitude of the UGV to

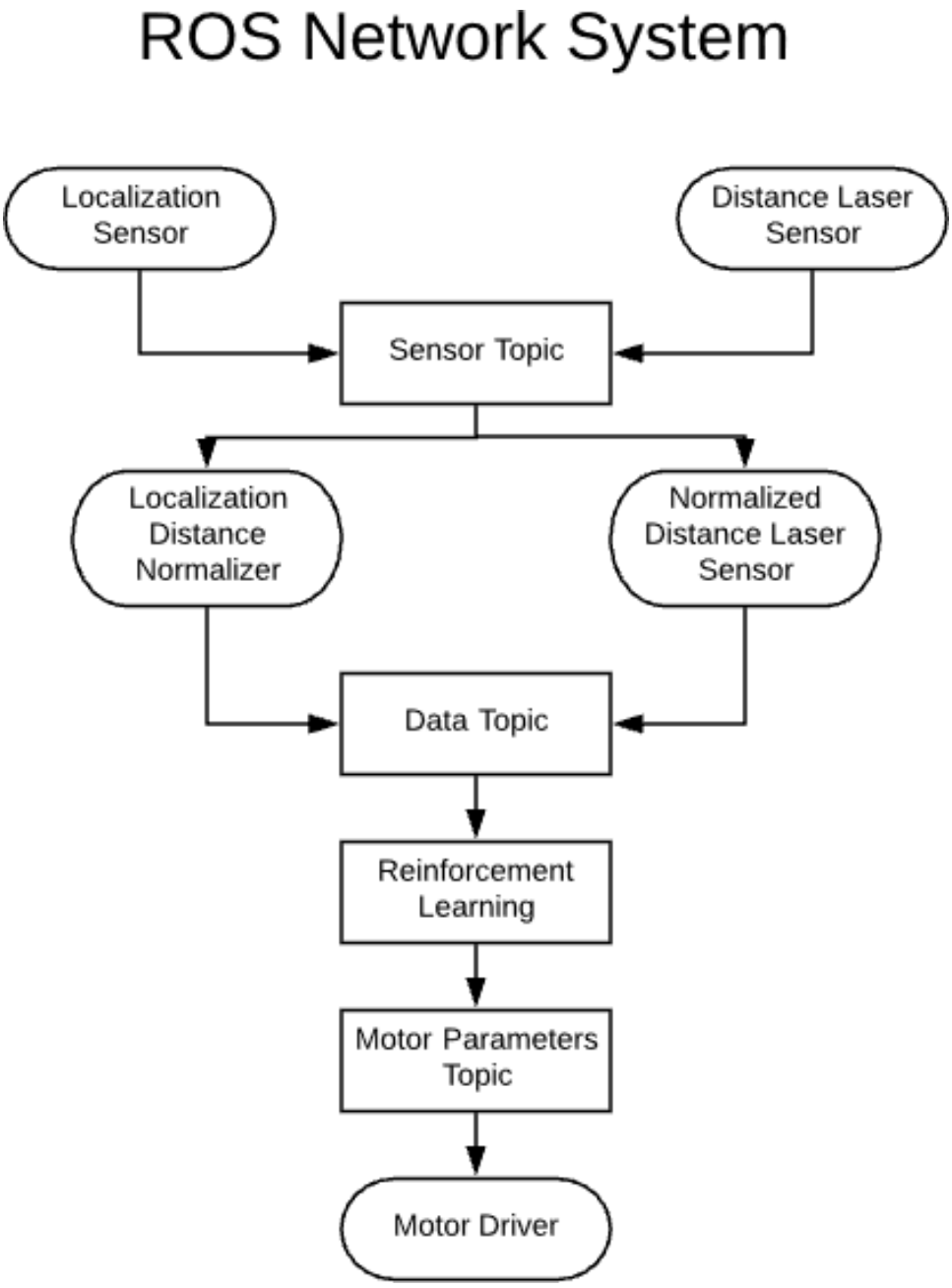


Figure 27: ROS Network System

the goal will never exceed the size of the map. A ROS node is used to map normalized laser distance sensor readings to the ranges compatible to those of the model inputs. Normalized localization and distance sensor information are time-aligned and concatenated into a single input vector, which

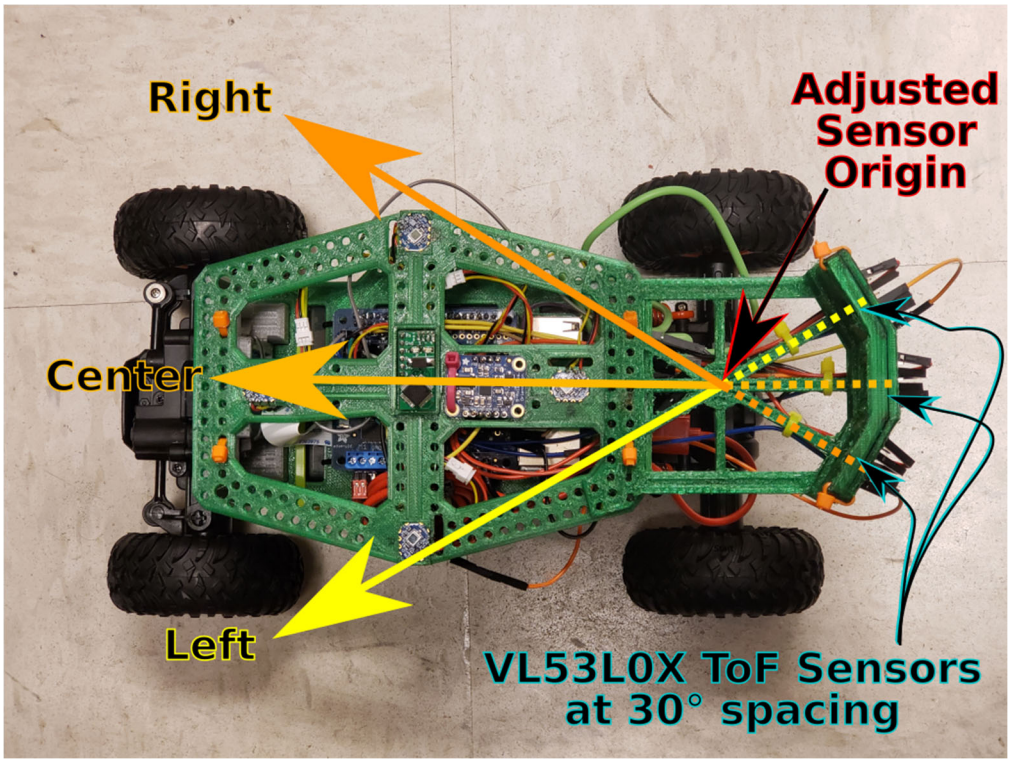


Figure 28: VL53L0X ToF sensor layout on small Ackermann steered vehicle. The arrow marked Center points towards the front of the vehicle.

Table 2: 100 Time of Flight Sensor Samples at Close and Long Range

Time of Flight Sensors	Physical distance from sensor origin	Average distance recorded from sensor	Standard deviation recorded from sensor	Distance error
Left Sensor	61cm	57.3cm	1.47cm	3.7cm
Middle Sensor	54cm	52.7cm	1.73cm	1.3cm
Right Sensor	61cm	56.8cm	1.41cm	4.2cm
Left Sensor	27cm	26.3cm	1.35cm	0.7cm
Middle Sensor	23cm	23.3cm	1.56cm	0.3cm
Right Sensor	27cm	26.1cm	1.23cm	0.9cm

then is processed by reinforcement learning DQN. The resulting output of the DQN is a decision on one out of six fixed possible outputs, (i.g., go slow and turn right), which is based on the maximum probabilistic output value of the last layer of the hidden layer.

Hardware for this experiment is modeled off a simulated vehicle developed by former TECHLAV student Abhijit Majumdar from his MS Thesis. The self-learning capabilities are illustrated in the simulation. The agent(car) in Figure 29 that is learning throughout the simulation; will be represented by the new model UGV shown in Figure 30. Figure 31 shows a recorded trajectory of the UGV under manual operations.

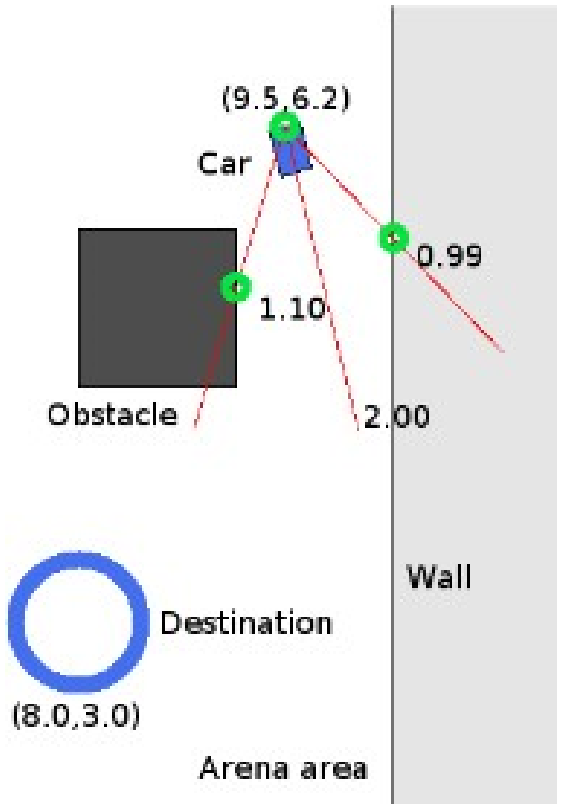


Figure 29:Photo of Simulated Agent(car) in an Environment.



Figure 30: Photo of old model on left and new Ackerman steered platform on right with AM-L03 sensor package.

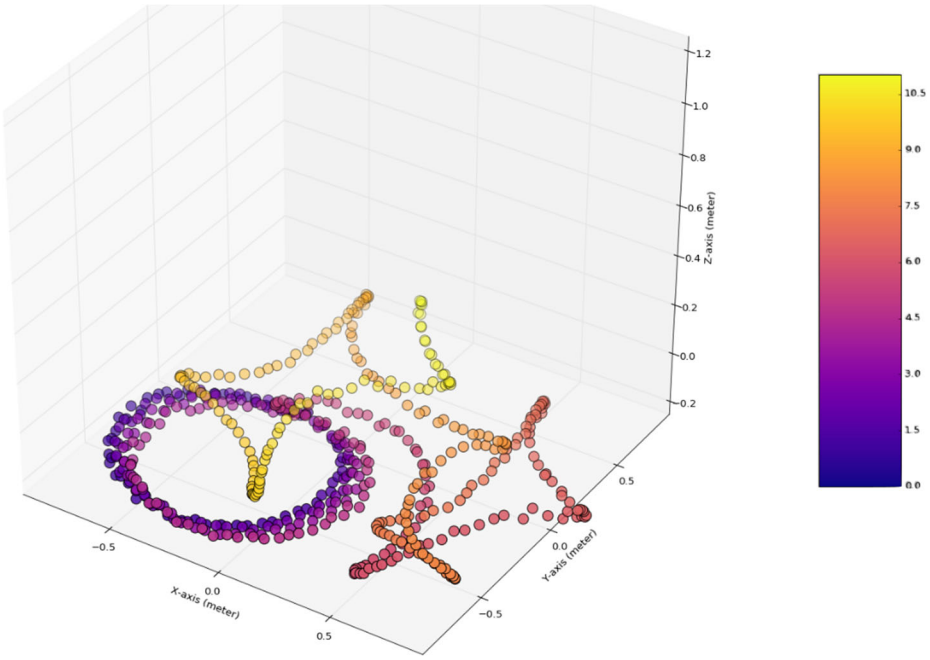


Figure 31: Recorded pose of manually controlled UGV pattern of driving in circles for approximately 4s followed by a pattern of three point turns for another 6.5s

Physical Environment: In this section, the importance of matching the physical environment on the virtual environment is discussed. The virtual agent explores and exploits a given environment to maximize the possibility of having the best Q-function at the end of training. As a result, the physical environment has to be similar to the virtual one that is constructed. See Figure 32 to observe an example of a similar pair of constructed physical and virtual environments. Additionally, the physical UGV has to be as similar as possible to the virtual one, as demonstrated by Figure 33. The dimensions of the physical environment are 1.63 x 2.43 m; this gives the UGV enough space to maneuver and test the learned model.

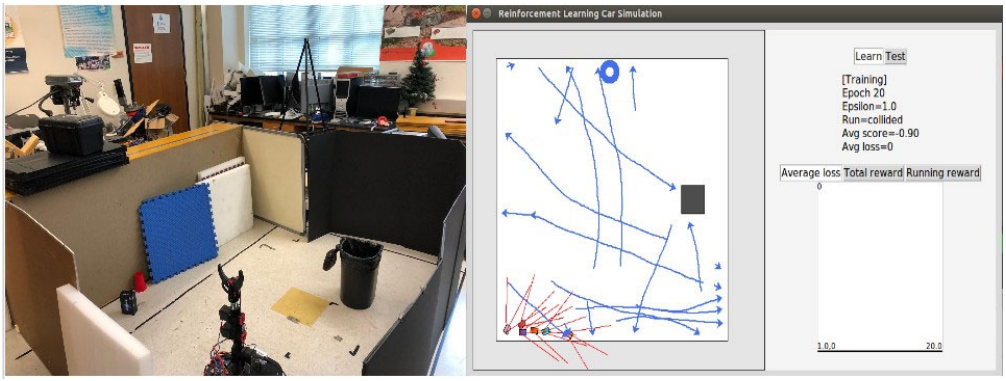


Figure 32: Physical Environment and Virtual Environment.

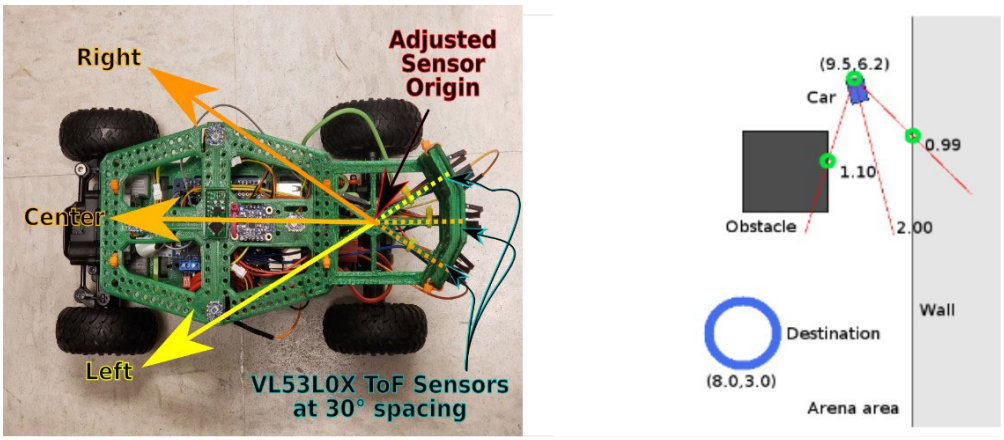


Figure 33: Physical UGV and Virtual UGV

Testing Generated Model on UGV: In this section, The GUI that is used to interface the virtual simulator was modified in order to test the generated model by the virtual agent. The GUI comes with two buttons, learn and test. In the learning mode, the reinforcement learning model is learning through the virtual UGV sensor data. Before, the test button only tested the learned model on the virtual environment. When the simulation is on test mode, the model starts testing the learned model based on the real physical UGV's sensor data. In Figure 34, it is observed that by using RVIZ ROS tool, the UGV moves from a pose A to a pose B.

Some compatibility issues arose in connecting the simulation agent's behavior model to the real robot. The physical UGV was not responding appropriately with the trained model to the distances read by the sensors. The root cause of this was determined to be that the virtual agent works off of integer type distances for ease of model convergence, while the physical UGV obtains floating point values from the sensors. The physical UGV's data will be scaled up and cast into an integer format to fit the pattern used by the virtual agent. This will utilize more range of the sensor's capabilities as the sensors utilized have higher accuracy in the 0-1.9m range and lesser accuracy in the 2-8m range. The proposed solution to this problem will lead to better decision making for the physical UGV in evaluating and learning a model.

3.1.3.3.4 DQN vs DQN With GA Box Map Figures 36 and 37 were simulated with the same amount of run time to compare the following characteristics. The agent running in Figure 36, is a second generation genetic algorithm neural network. Figure 35 and tables 3, 4 illustrate the neural network structure, and all features that were chosen by the layer randomization function used in each agent. Figure 36 illustrates the training running reward gathered by the DQN with GA agent model. As observed in 36, the running reward exponentially increases, thus, the neural network structure has successfully developed enough so that the model trains for a more reliable future test model.

On the other hand, Figure 37 has a different running reward behavior. As illustrated in the image, the running reward had temporarily spike then fell off, and struggled to increase the reward back again. This is an example of a neural network structure that isn't complex enough to optimize

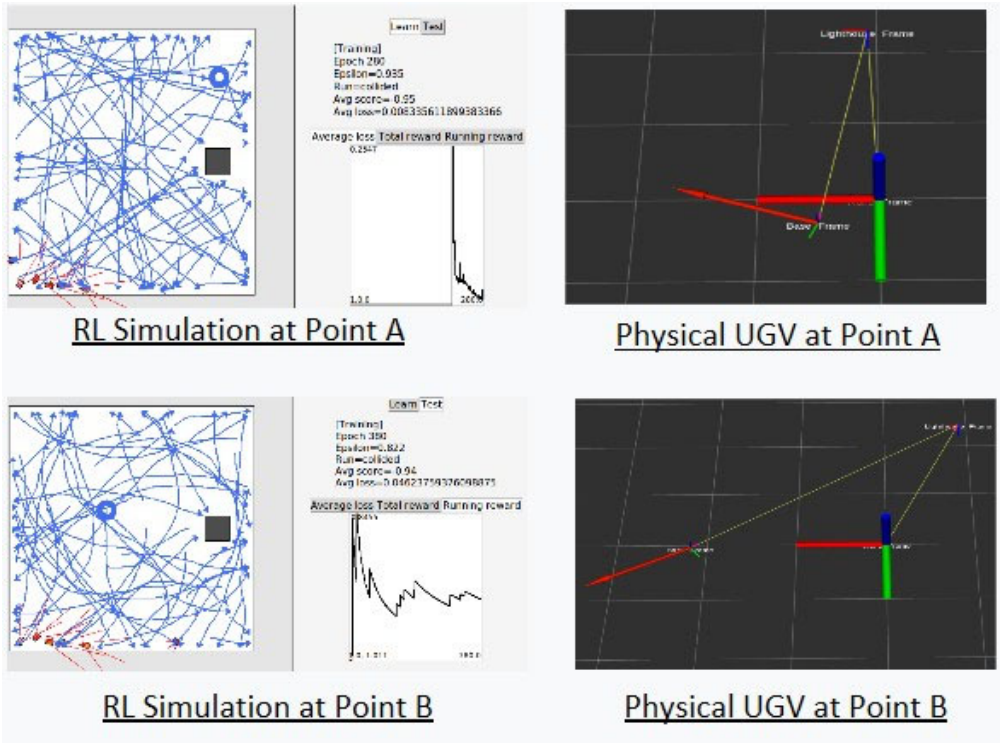


Figure 34: Testing Model on Physical UGV

running reward. Table 5 describes the neural network feature structure chosen by the user. As a result, the model suffers from a poor selection of hidden layer parameters, optimizer method, loss function, and activation function.

Table 3: Agent 1 Model Characteristics Used in DQN With GA Simulation.

Layers	Concatenate Layer	dTransitional Layer	Concatenate Layer	dTransitional Layer
Number of Parameters	86	48	81	96
Activation Function	Hard Sig-moid	relu	Exponential	Tanh
Optimizer	Adagrad	Adagrad	Nadam	Adadelta
Loss Function	Mean Absolute Percentage	Categorical Hinge	Mean Abso-lute Error	Categorical Hinge

DQN vs DQN With GA Track Map: A different response is observed from this map. In this map the reward is very small until the goal is finally achieved. As a result, once the goal is reached by the agent, it's very easy for the agent to achieve it gain, which explains the ramp up of running rewards. Despite that both models in Figure 38 and 39 ramp up in running rewards, DQN with GA has greater area under the curve while DQN approach had a minimum struggle in

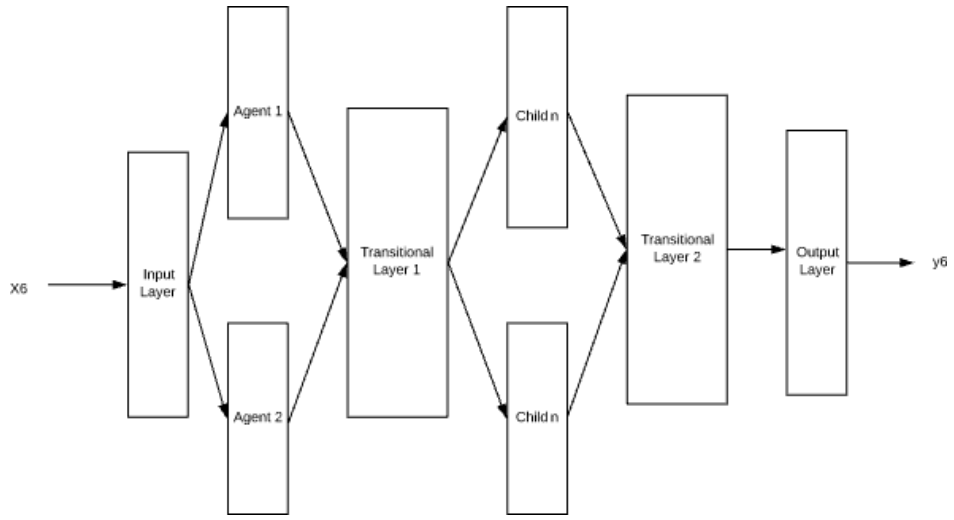


Figure 35: Neural Network Structure of Second Generation Genetic Algorithm Agent.

Table 4: Agent 2 Model Characteristics Used in DQN With GA Simulation.

Layers	Concatenate Layer	dTransitional Layer	Concatenate Layer	dTransitional Layer
Number of Parameters	26	48	36	96
Activation Function	Sigmoid	relu	tanh	Tanh
Optimizer	Adagrad	Adagrad	rmsprop	Adadelta
Loss Function	Hinge	Categorical Hinge	Categorical	Categorical Hinge

Table 5: Agent Model Characteristics Used in DQN

Layers	Hidden Layer 1	Hidden Layer 2	Hidden Layer 3	Hidden Layer 4
Number of Parameters	64	20	20	12
Activation Function	None	None	None	Linear
Optimizer	Adam	Adam	Adam	Adam
Loss Function	mse	mse	mse	mse

Simulation. The beginning which lead to a delayed ramp up.

Physical Results: As observed in Figure 41, the model had an interesting way of learning a policy for the environment. Under epoch 100, there is no sign of intelligence. At 500 epochs, the agent learned to reach the destination, and not to time out. At 2000 epochs, the performance increased dramatically, with zero time outs. At 4000 epochs, the agent began to time out again, in order to avoid collisions. At 20000 epochs, the model converged at some value, and the results are as illustrated in the figure. Again, the agent increased to 6% to avoid collisions; this happens when the goal is too close to the wall, and the agent knows that it needs to get the goal but it cant

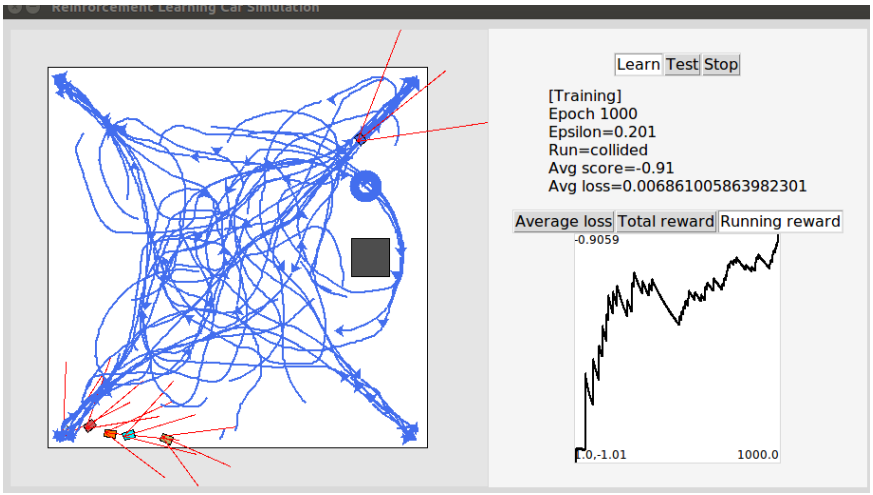


Figure 36:Image of DQN with GA Simulated Agent and Running Reward Graph Learning at 1000 Epochs

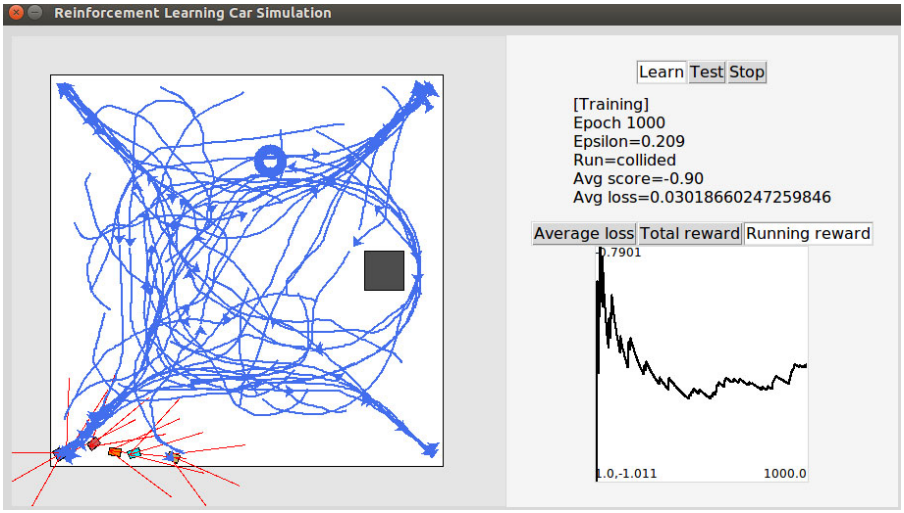


Figure 37: Image of DQN Simulated Agent and Running Reward Graph Learning at 1000 Epochs

collide, resulting in a time out.

This model was applied to a robot shown in Figure 40, and it was making the right decisions only 49% of the time. This test was made by placing objects on the UGV's sensors were the model was making decision based on the data collected. The same procedure was done with the DQN with GA model, and the results were better over all. The UGV was making the right decision 60.8% of the time; that is an increase of approximately 12% from the DQN mode to the DQN with GA model. Therefore, the method proposed did improve the performance of reinforcement learning

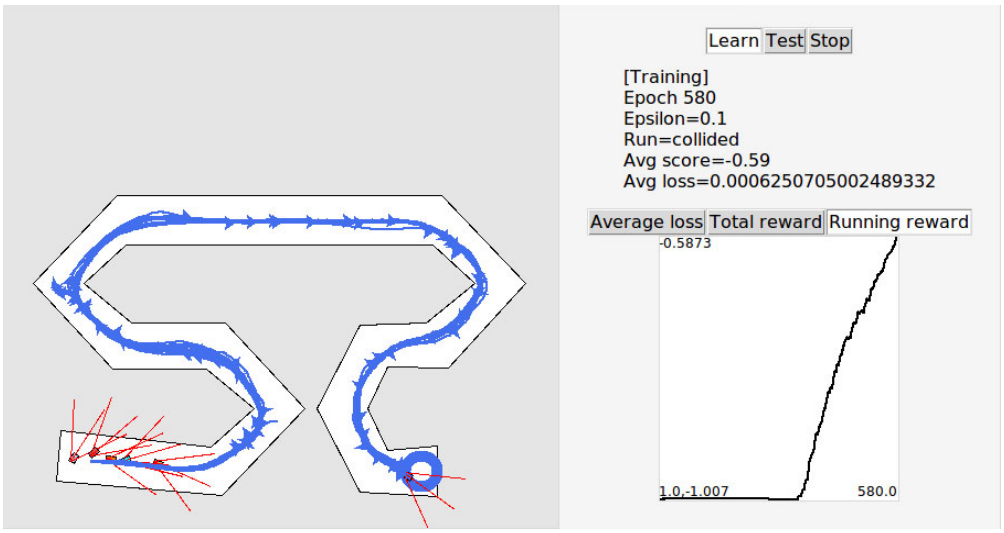


Figure 38: Image of DQN with GA Simulated Agent and Running Reward Graph Learning at 580 Epochs on Track Map.

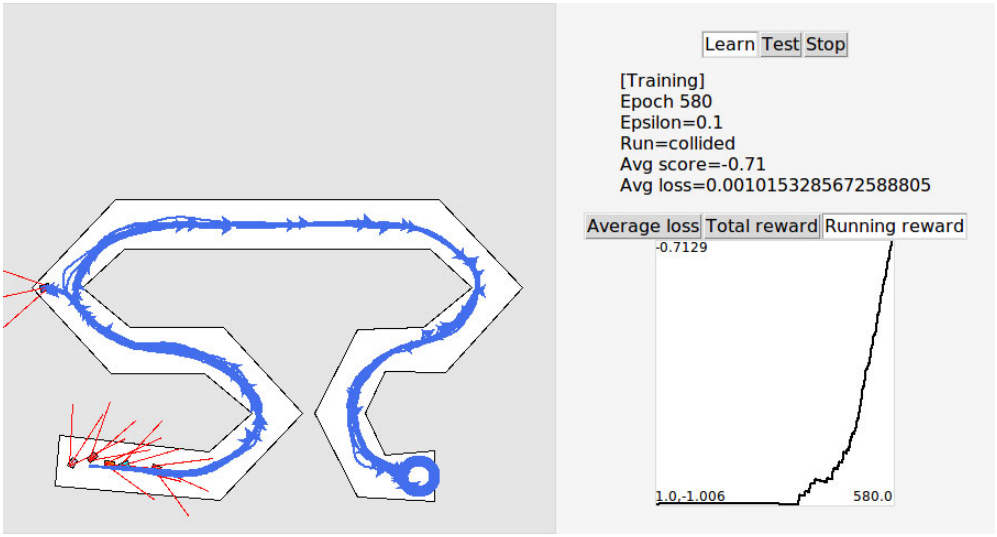


Figure 39: Image of DQN Simulated Agent and Running Reward Graph Learning at 580 Epochs on Track Map.

using DQN.

3.1.3.3.5 Improvements Upon Reinforcement Learning For Autonomous Agent Navigation After developing a simple training environment in simulation for the agent vehicle model, genetic algorithm was under consideration for improving the network of the deep reinforcement learning model. Ultimately it was not incorporated into the simulator, but instead attention was given to creating a more sophisticated simulation environment to be adapted for training the vehicle using reinforcement learning.

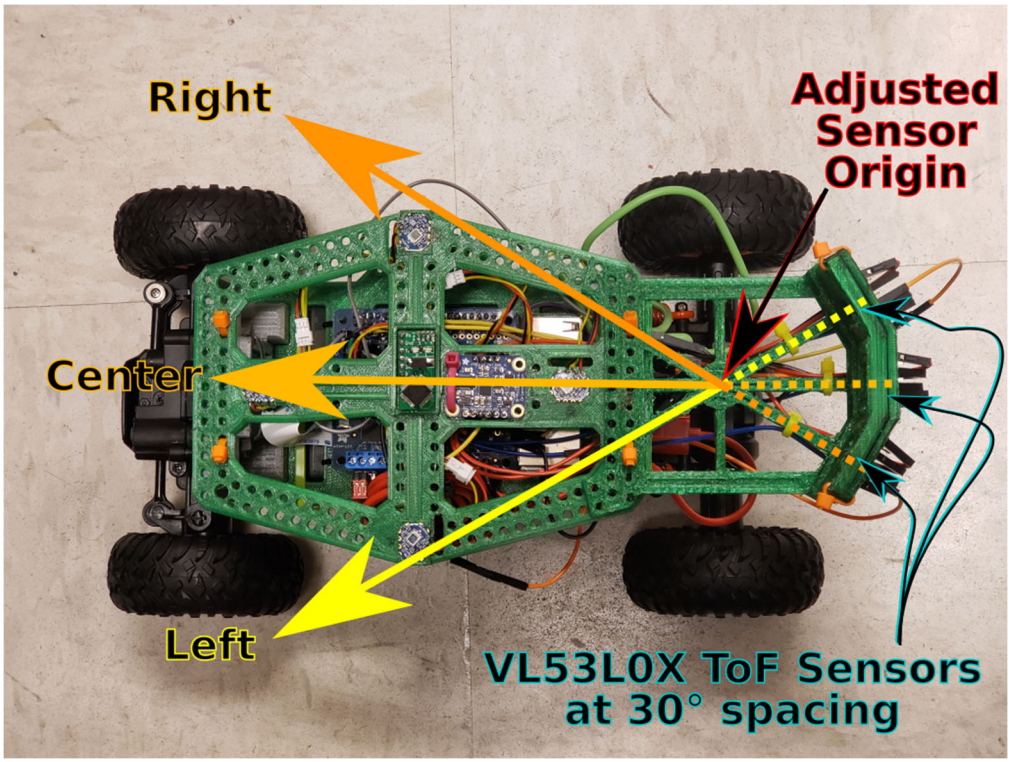


Figure 40: VL53L0X ToF sensor layout on small Ackermann-steered vehicle. The arrow marked "Center" points towards the front of the vehicle.

Testing Results at Specific Epochs					
	Epoch 100	Epoch 500	Epoch 2000	Epoch 4000	Epoch 20000
# Samples	50	50	50	50	50
% Destination	% 0	% 2	% 48	% 54	% 64
% Collided	% 84	% 94	% 52	% 44	% 30
% Time Out	% 16	% 4	% 0	% 2	% 6

Figure 41: Previous Results on DQN Simulated Model.

The components of the original application included functionality for training the navigation system using reinforcement learning and were combined with a new GUI to create a complete application for this project which can enable efficient deployment and testing of the mobile agents. Offering the ability to calibrate and read all of the individual sensors on the vehicle for testing purposes, manually control the vehicle via keyboard commands, and providing an extension of the reinforced learning navigation system into a single application allowed for the project to be maintained and scaled appropriately with ongoing development efforts. This enabled efficient deployment of the physical vehicle for evaluating the performance of pre-trained reinforcement learning models.

The next stage of the application would entail deploying the trained policies and evaluating the vehicles performance as it attempts to navigate towards a given destination and avoids collisions with obstacles. The simulator that was initially created in Python was enhanced to include a physics engine and more accurate vehicle model within Unity. Virtual reality was incorporated into the advanced simulation environment to allow tracking of the physical vehicle within a virtual space. If the position of the physical vehicle (tracked using the HTC Vive virtual reality system) were to coincide with the position of a virtual obstacle constructed in Unity, the trial could result in safe termination without an actual vehicle collision.

The ability to train a vehicle model to navigate towards a goal position autonomously while avoiding collisions with surrounding obstacles was made possible using reinforcement learning algorithms within a Python simulation environment as shown in Figure 42. Multiple vehicles were incorporated into the environment for enhanced learning in terms of speed and performance.

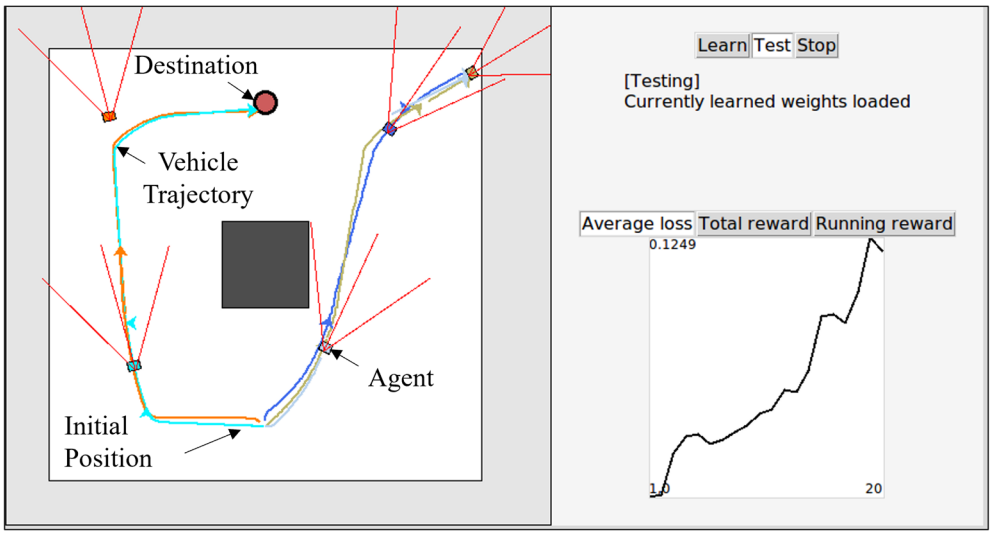


Figure 42: Python simulation environment for training a vehicle using reinforcement learning

Deployment, testing and control of the mobile robot (client) will be facilitated by a custom GUI application running on a separate PC (host) by providing several utilities accessible via a user-friendly interface with labeled button inputs initiating the following utilities (spawning threads for the individual processes). By running the GUI from a separate network host to control the processes on the remote vehicle, the overall memory requirements of the robot's microcomputer are reduced drastically. The application is suitable for testing and controlling the robot at the human-operator level, with several utilities made easily accessible, as shown in Figure 43.

A virtual reality testing environment shown in Figure 44 has been constructed in Unity using the SteamVR SDK which replicates the simple arena used for training the robots using reinforcement learning. In this environment, it is possible for the user to issue waypoint destinations for the vehicle, or configure static destination for consistent testing. With the ability to fully visualize testing as it is occurring, this method allows for greater insight into the behavior of the trained models.

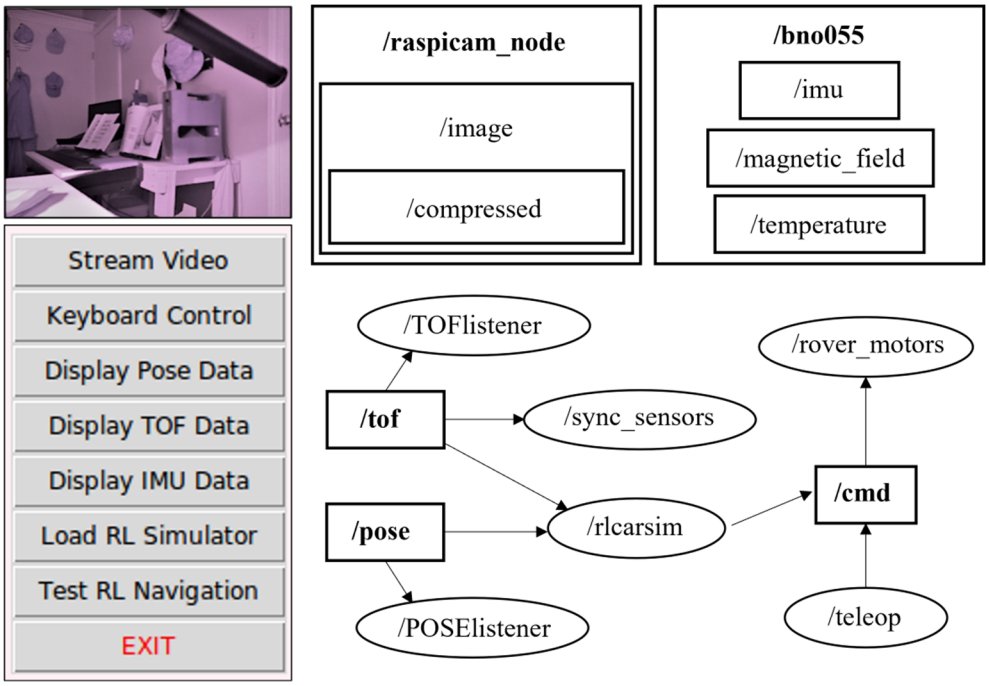


Figure 43: GUI Application for Testing Pre-trained Reinforcement Learning Navigation Models

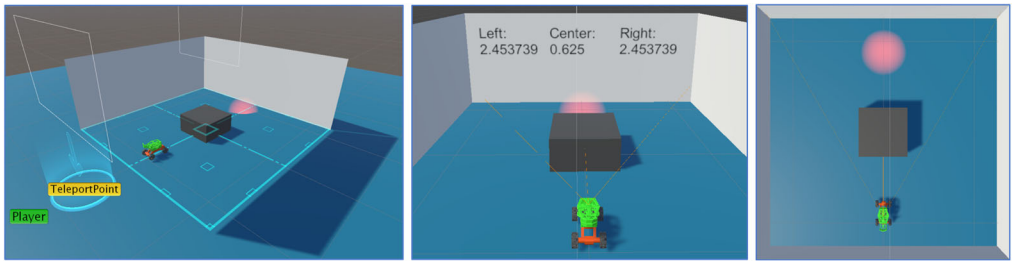


Figure 44: Virtual Reality Testing Environment

2.1.3.4 Summary

In this task the implementation of autonomous navigation of an unmanned vehicle with collision avoidance was examined. A custom simulation environment was created using Python for training a system of vehicles using deep reinforcement learning methods. The vehicle model and simulation environment were developed as a result of T1-1. The result of training was a navigation policy, which can be described as a set of rules for the vehicle to follow so that it avoids collisions with nearby obstacles as it traverses its environment searching for a provided goal position.

Deploying this trained policy on the physical vehicle was assisted by the development of a simple user-friendly GUI for visualizing the sensor data, manually controlling the vehicle using keyboard teleoperation, and selecting a specific pre-trained policy for evaluation. To mitigate the concern over deploying a poorly trained policy that could potentially result in collisions within

the testing environment, a new approach was proposed to track the physical vehicle using virtual reality equipment, and create a virtual environment around it composed of simulated obstacles. This strategy could be used for safely and cost-effectively evaluating many different environments with the actual vehicle.

3.1.4 Project Progress for Task T1-4 (Cooperative Localization and Navigation for LSASV)

3.1.4.1 Period of Performance under

Task T1-4 Start Date: Fall 2015

Conclusion Date: Spring 2021

Faculty lead: Mo Jamshidi

3.1.4.2 General Description of Task T1-4

A class of localization and navigation algorithms called Simultaneous Localization and Mapping (SLAM) provides a robust and accurate approach for both tasks. SLAM algorithms that utilize visual information (vSLAM) require large amounts of computational power and storage, which makes it difficult to implement in real-time for LSASV. Of the main bottlenecks in vSLAM, feature identification and matching across a large database is the most time-consuming. And so, in this task, we will implement a novel method to reduce computational time, storage requirements, and network load to perform feature identification and matching components of vSLAM in real-time. The proposed process selects out only the most feature-rich components of visual data to be used in matching and database creation. We will further improve feature detection algorithms using proven feature detectors such as ORB (Oriented FAST and Rotated BRIEF) and SURF (Speeded-up Ro- bust Features) on readily available datasets. This will significantly reduce storage requirements and calculation time. The proposed more efficient vSLAM algorithm will be implemented remotely from the LSASV entities themselves (in a cloud data center), taking advantage of high-performance computing systems and parallel processing of algorithms. The network and cloud access delays will be compensated by predictive approaches to minimize their effects on the overall system perfor- mance. Visual information will be fused with known, learned, or sensed physical location references to create a world map. World maps will be kept in a big-data database and made equally available to all required parties. While each vehicle (agent) uses the developed SLAM algorithms to create world maps, on an LSASV level, a fusion of agent world maps is utilized to correct and provide global references to the system. Complex alignment operations to fuse world maps are to be inves- tigated in this task. Complexity in the operations comes from many variables in LSASV, including the pose and perspectives of various sensors on the agents, errors in position data, interference between agents, and dynamic environments. The fusion of this data must be performed with high precision in order to acquire the best estimate of the position of all agents in the system as well as important spatial features within the world map. Registration is one form of fusion in which maps are compared for matching features and transformed to fit together. The registration process can be time-consuming, if not performed with high levels of parallel computing. This task, therefore, develops a real-time registration method for multiple LSASV entities.

3.1.4.3 Technical Accomplishments in Task T1-4

Visual SLAM using image-based feature mapping fused with odometry data was explored throughout the length of the project as a means to mimic natural processes. Early surveys into the state of the art in SLAM techniques exposed many routes to explore in the generation of a cooperative localization and navigation system. The format of maps generated by algorithms varied in complexity between a 2D flat map and a 3D point cloud. Various methods can arrive at the same types of maps. Computation heavy mapping was determined to be not done in real-time using any of the initial methods investigated. All works in the literature had conditions on the "real-time" nature of their developed algorithms were expecting advances in hardware that were not yet realized, as they relied on post-procesing using either GPU-enhanced desktop workstations and/or servers.

3.1.4.3.1 SLAM using Image-based Feature Mapping Techniques Early investiga- tions explored methods using fundamental building blocks of image-based feature matching. Fea- ture detectors such as BRIEF, ORB, and SURF were explored as an image-to-image comparison technique. Early maps generated using the method were successful, but exposed other problems and methods for the group to pursue, such as the loop closure problem, optimal landmark selection, particle filtering, and handling false-positive matches (due to visually complex objects that matched almost anything). More complex feature matching techniques and supplemental navigation data sensor fusion were identified as means of improvements.

The ODROID XU4 has an octacore CPU (quad Cortex A15 + quad Cortex A7) and six cluster GPU (Mali-T628 MP6) that can be used heterogeneously. The CPU works best for management and processing of data, while the GPU shines at matrix calculations, such as those typically performed on image data. An experiment was run on some data sets collected from the TUM RGB-D SLAM Benchmark Datasets. Selected from the database are the 'freiburg2_pionee_slam*' datasets. Each dataset comes from data captured on a Pioneer II mobile robot with a Microsoft Kinect. The reported elapsed times includes the time it takes 1) to read all images in a dataset from the disk and 2) to extract the features from the images. In the second test, with active re-sizing of images from 640x480 to 320x240 pixels, the elapsed time also includes the time to re-size an image. From the results in the tables, re-sizing the images down to a more manageable set of dimensions will aid in approximately a 2.37x improvement (2.333x to 2.399x) in the processed FPS.

Investigation Into Real-Time Appearance-Based Mapping

Real-Time Appearance-Based Mapping (RTAB-Map) is an algorithm that is able to create a map though the use of RGB-D features it was created by two researchers in the paper "Online Global Loop Closure Detection for Large-Scale Multi-Session Graph-Based SLAM". This package was used to begin to create maps using the hand held Microsoft Xbox Kinect, and used to create a map of one room. While the results are not perfect the package does have other algorithms to help create more prefect maps.

Fig. 45 shows the program in the process of creating a map of the hallway near the ACE Lab. The map can be viewed in the main frame of the window. While the three smaller windows show the feature matching process. RTAB also contains a database viewer, shown in Fig. 46 and 47 where a user is able to view a map that has been previously created and saved.

The database viewer can be used in order to view all images that were collected in the creation of the map. This can be useful in order to ensure that the feature detection and comparison was correct during the mapping process.

Table 6: Trials for Processing Image Features using ORB (Image Size 640x480)

Database	# of Frames	Elapsed Time (5 Trials)	Processed FPS
'freiburg2_pioneer_slam360'	1225	116.936369s 116.996979s 117.128929s 117.501053s 118.658438s	10.324 (min) 10.430 (avg) 10.476 (max)
'freiburg2_pioneer_slam1'	2921	278.156957s 279.611749s 279.832552s 279.876063s 280.397665s	10.417 (min) 10.448 (avg) 10.501 (max)
'freiburg2_pioneer_slam2'	2113	199.630108s 201.791167s 202.082897s 202.237943s 202.482589s	10.435 (min) 10.479 (avg) 10.585 (max)
'freiburg2_pioneer_slam3'	2544	247.055978s 247.076492s 247.126593s 248.831792s 250.983332s	10.136 (min) 10.249 (avg) 10.297 (max)

Table 7: Trials for Processing Image Features using ORB (Image Size 320x240)

Database	# of Frames	Elapsed Time (5 Trials)	Processed FPS
'freiburg2_pioneer_slam360'	1225	48.941726s 48.947537s 48.948939s 48.957369s 48.974762s	25.013 (min) 25.023 (avg) 25.030 (max)
'freiburg2_pioneer_slam1'	2921	117.372682s 118.062992s 119.221245s 119.388023s 119.415593s	24.461 (min) 24.610 (avg) 24.887 (max)
'freiburg2_pioneer_slam2'	2113	84.490087s 84.710071s 85.832518s 86.150631s 86.589737s	24.402 (min) 24.698 (avg) 25.009 (max)
'freiburg2_pioneer_slam3'	2544	104.847575s 105.941169s 106.885078s 107.094273s 107.323132s	23.704 (min) 23.906 (avg) 24.264 (max)

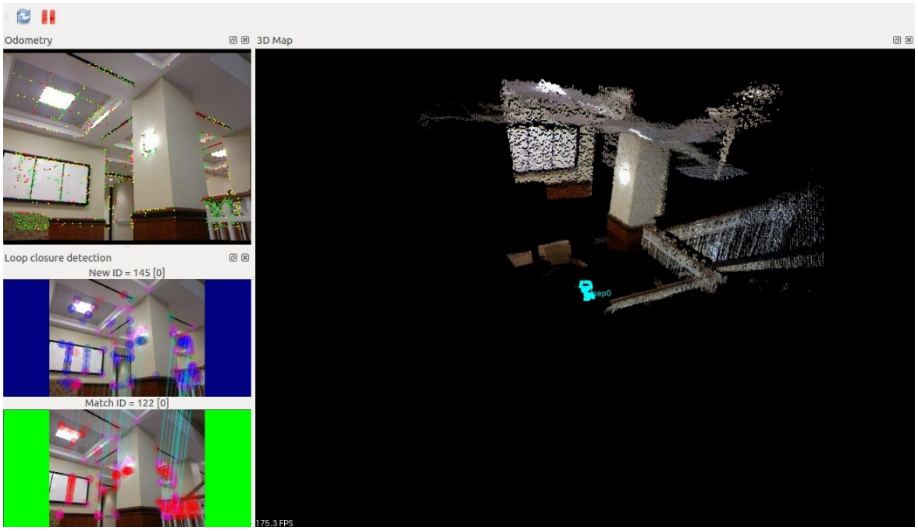


Figure 45: A map captured through RTAB ROS

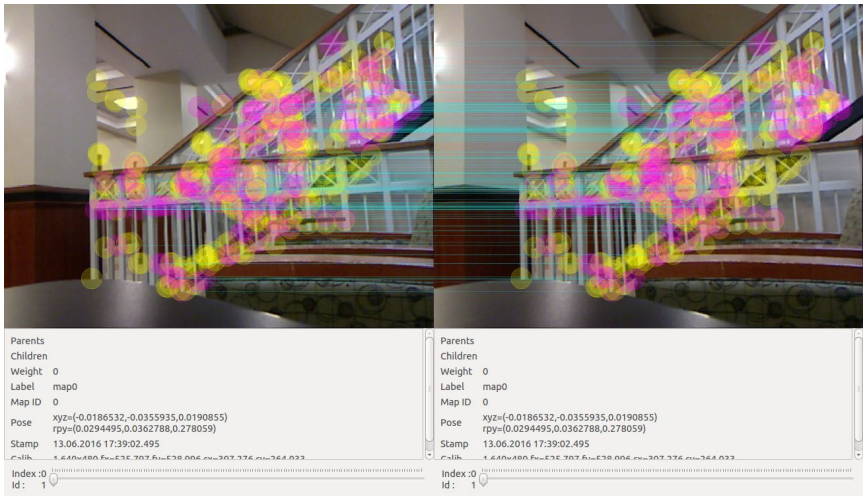


Figure 46: The database viewer showing the same image



Figure 47: The database viewer showing different images

3.1.4.3.2 Multi-agent Coordination for Underwater and Ground Vehicles A visit- ing researcher studying underwater rovers worked with the team on a ground vehicle that interfaces with it. Special QR codes called AR Tags were used for waypoints to test algorithms for naviga- tion. A visual system recognized large and small tags as well as combination of large and small tags. The ROS package AR track Alvar was used where the position of each tag that is detected by the visual system. Simulations involved generating meeting points for the underwater and the land based swarm using an auctioning method. Each agent was able to determine which meeting point was best suited to transfer an object to a land rover by calculating a score based off which agent was closest to the land rover. A depth map was generated from the location of a submerged object, determinend relative to the position of the underwater rover, and registered into the global space. From this depth map, the depth of any unknown underwater objects was determined based of the closest depth locator position, and thus find the relative size of the unknown objects using a single camera. Retrieval was accomplished through calculating position of the unknown objects, navigating towards them in a position where the robot was ready to pick them up. Hardware issues arose that prevented the autonomous collection of the objects themselves to be fully completed. A slower approach profile was identified as a need for improvement while traveling towards an object. The developed setup with a fixed manipulator sometimes caused the robot to bump or nudge the object as it approached, potentially causing it to fall over or translate position, making it more difficult to collect.

3.1.4.3.3 Exploration of Point Cloud-based Methods One of the goals in the research with point clouds was to implement and perform hardware experiments on ROS compatible SLAM GMapping package. The Kobuki Turtlebot 2 UGV platform, a differential mobile robot, was used for the experiments to share gained knowledge on both hardware and software for producing a cooperative research effort. The experiments were performed on the campus of North Carolina A&T and State University (NCAT) campus as a part of the TECHLAV Summer Student Mobility Program. Another planned objective for this reporting period was to integrate Darknet YOLO deep neural network object detector into ROS for performing selective landmark identification.

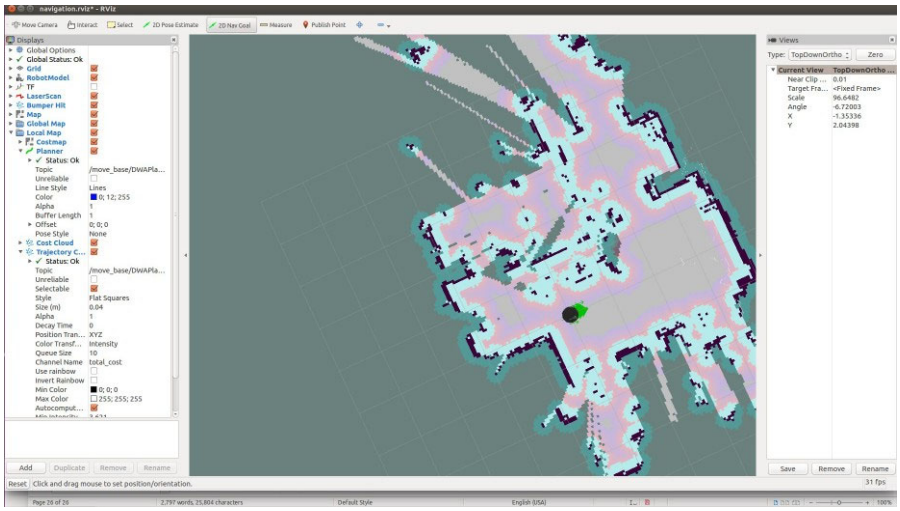


Figure 48: Preliminary simulation results for single agent mapping in RViz environment by using GMapping package with default settings.

SLAM Gmapping Implementation The SLAM Gmapping package had to be modified in order to make it compatible with the NCAT UGV setup for testing purposes. It was quite challenging to configure two UGVs for cooperative navigation tasks, and also to create a global map using inputs from different sources. The following Linux bash shell scripts summarize the first stage of the experiment. The experimental results for one UGV with default setup in the RViz environment can be seen in Figure 48.

- *roslaunch turtlebot_rviz_launchers view_navigation.launch*
- *roslaunch turtlebot_navigation amcl_demo.launch map_file:=/tmp/my_map.yaml*

Multi-agent mapping adds extra contraints to the operation. In addition to expected constraints such as network connectivity, latency and control errors, particular problems from the default parameters of the package and ROS distribution compatibility increased the frustrations. The scripts to perform the experiments are written below, followed by the experimental results before the modifications for two UGV with default parameters as can be seen in following Figures 49 and 50.

- *roslaunch turtlebot_bringup minimal.launch*
- *roslaunch turtlebot_teleop keyboard_teleop.launch*
- *roslaunch turtlebot_navigation gmapping_demo.launch*

For having better results, we had to modify the default parameters and setup a better connectivity through the network. Another problem was to introduce the *other agent* on the system or in the network to our local agent. We solved this issue by integrating the master-slave approach into the system by naming the UGVs (by changing the argument on the *ROS_NAMESPACE*=) and running the application through the master while slave was performing the SLAM.

Since the compatibility and hardware issues were giving us trouble on the UGV and the SLAM package, we added *"export TURTLEBOT_3D_SENSOR=kinect"* into the *.bashrc* file on ROS,

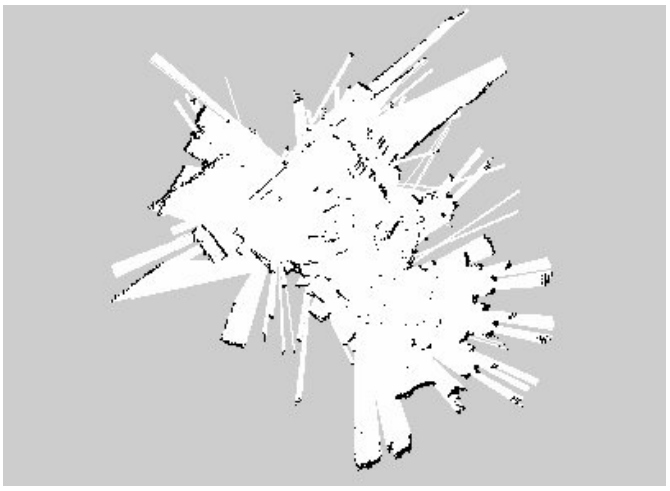


Figure 49: Preliminary experimental trials for cooperative mapping from Gmapping package with RGB-D camera feedback. It is easy to determine the lack of merging capability before the modifi- cations.

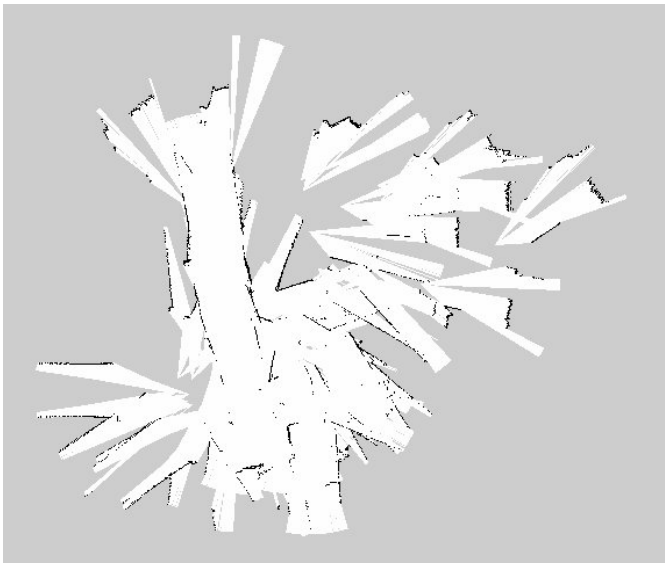


Figure 50: Experimental trials for cooperative mapping from Gmapping package with RGB-D camera feedback after some modifications on default settings. Improvements can be seen in the solid lines which indicate the shapes of the walls and corridors.

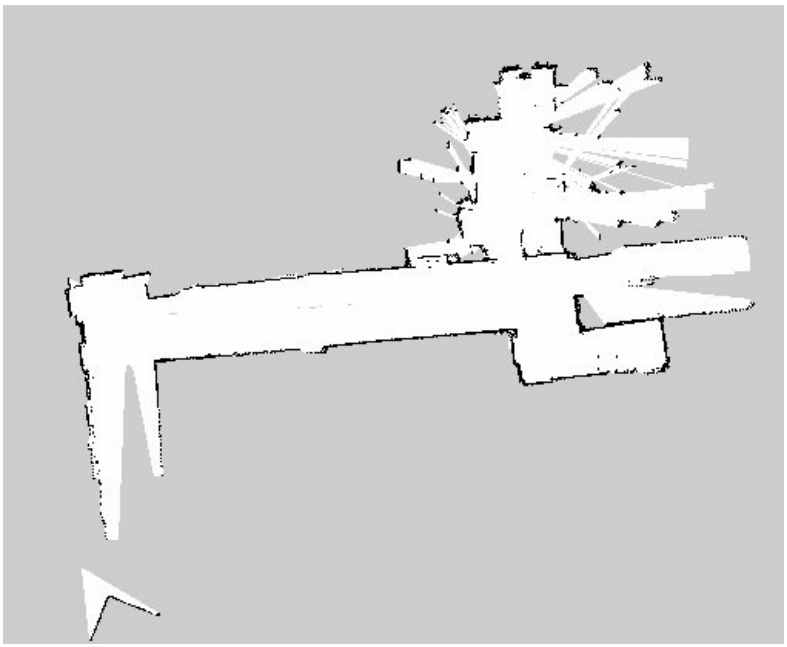


Figure 51: Cooperative mapping from Gmapping package with RGB-D camera feedback after final- izing the modifications on default settings and solving the networking issues. It is easy to observe higher accuracy obtained through cooperative mapping. A better global map was obtained through increased amounts of overlapping areas and better feature matching by sensor reading improve- ments. The particular noises can be seen in the figure that are related to the office environment, which is mostly a result of the tables, chairs and the location of the camera on the UGV.

NCAT has different version of the Kobuki Turtlebot platforms. The following scripts summarize the steps of our approach that helped us to improve the multi-agent system and to build cooperative maps resulting in Figure 51.

- *roslaunch turtlebot_bringup minimal.launch*
- *roslaunch turtlebot_navigation gmapping_demo.launch*
- *roslaunch turtlebot_rviz_launchers view_navigation.launch*
- *roslaunch master_discovery_fkie master_discovery _mcast_group:=224.0.0.1*
- *rosrun master_sync_fkie master_sync*
- *roslaunch multirobot_map_merge map_merge.launch (after sourcing the catkin workspace)*
- *rosrun map_server map_saver -f /tmp/my_map()*

SLAM Gmapping with Slice of RGB-D Camera Depth Data Inputs to the ROS pack- age *slam_gmapping* are a laser scan slice of the environment about the z −*axis* and the estimated odometry data. The software takes these two inputs and maps them back into a history of scans and odometry taken by the robot. Particle filtering is used to determine the best estimate of where the robot is within the learned map.

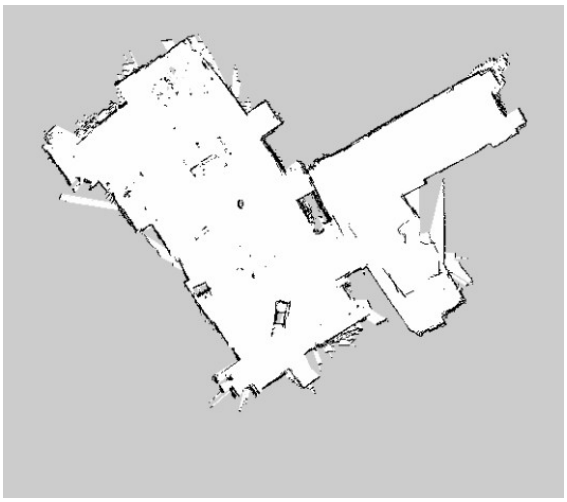


Figure 52: Map generated from slam_gmapping.

The level of information obtained from a single *z axis* slice of data at a certain height does not tell much about the environment in terms of features. There is a function available in ROS which allows you to convert a depth image from an RGB-D camera into a an emulated laser scan. The function provides an option to effectively condense a range of *z axis* depth image slices into one laser scan style measurement. This operation is done in a ROS node called *depth_image_to_laserscan*, which is pre-compiled program for performing the operation of converting depth images to laser scans. The height of the camera, with respect to the pre-defined "*base_link* " in the Kobuki's Uni- versal Robot Description File (URDF), is set and published by a *static_transform_publisher* node from the ROS *tf* package. The result is shown in Figure 52. The map contains several noisy elements that appear as trailing triangles or rotated rectangles. These elements are due to noise and odom- etry errors from the limited amount of information being obtained and stored. In Figure 53, the map from Figure 52 has been annotated to indicate some landmark features within the environment.

Integrating Robot Operating System (ROS) and MATLAB to control Unmanned Air Vehicles (UAV) A second planned part of the agenda for the student mobility program was using ROS and MATLAB to control robotic agents and analyze the system components. A controller package for MATLAB Simulink was introduced to us; however the package was not able to fully control the UAV. In the demonstration, the NCAT student used a Simulink control box to fly the ARdrone 2.0. Use of this controller only resulted in taking off, hovering, and landing the UAV. As other controls were not implemented, the ARdrone flew to an altitude of 1m above the floor and stabilized it using the on-board controller. We looked at possible ways to connect MATLAB and ROS, and the best solution was the Robotics Toolbox in MATLAB. The first idea we came up with was to use a virtual machine with in order to Linux to study the interaction between ROS and a windows computer running MATLAB. Unfortunately, we could not achieve bidirectional communication between the Virtual Machine and the host machine in the first trials. Once we configured a Linux machine and installed ROS, it was very simple to connect with the MATLAB toolbox. Then, we worked on connectivity between MATLAB and ROS operating on two different computers using the MATLAB Robotic Toolbox. Later, we installed the ARdrone

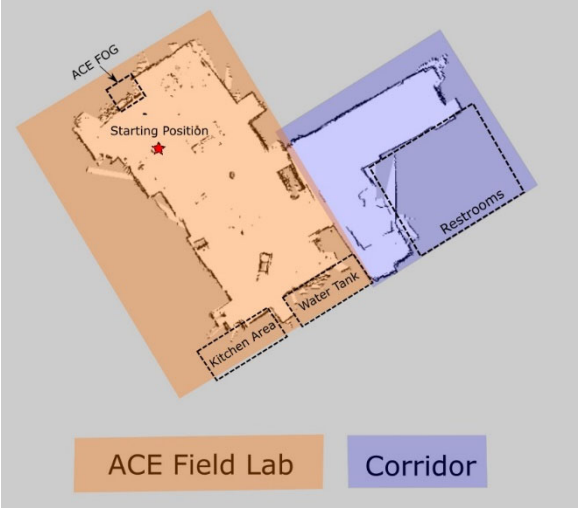


Figure 53: Landmark annotated version of Figure 52

drivers onto the Linux machine; however, interfacing with the ARdrone was not possible due to incompatibility of the provided USB Wi-Fi dongle with Linux platforms, as there were no drivers available for it. Even with a compatible Wi-Fi dongle, it was determined for some reason that the Parrot's ARdrone would not connect to ROS Indigo distribution. Therefore, we decided to switch to the Parrot's Bebop Drone, and performed a successful takeoff. Since Bebop drone is relatively newer, the communication was more reliable. Being that the goal was to connect between ROS and MATLAB, we demonstrated the ability to publish the drones odometry data on a Simulink block diagram. This will allow anyone who can create a controller in MATLAB or Simulink to control the drone without knowledge of the intricacies of ROS programming in python or C++. The MATLAB Robotic Toolbox and ROS connection based on network nodes of ROS messages for/from publisher/subscriber model can be seen in Figure 54. The steps we followed in the toolbox to build the connections that allow two platforms to communicate with each other are following:

- Run the Matlab toolbox on a Windows computer to create a master node, publishers, subscribers, and topics.
- Create both MATLAB and Simulink interfaces using the provided examples as a reference.
- Run a node based on the *hello world* publisher example on a Linux computer, which also subscribes to the output on a MATLAB Simulink block diagram.

The next goal was to fly the Bebop drone, instead of just hovering, by using a *teleop* method through either a keyboard or joystick. Using a configuration file, we were able to modify the axis and buttons to allow it to be used with the joystick that was available. The drone flew in all directions and the odometry data was published to MATLAB.

The final objective was to be able to record the odometry data and also control the drone's camera during flight. We demonstrated two ways to record any ROS topic. The options are to record the ROS topics using a bag file or to redirect the *rostopic echo /topic_name* to a text file through " ", t pipe redirection >> and concatenation operator. The interaction between ROS and MATLAB can be seen in Figure 55. One issue arose in configuring the ArDrone Bebop camera feed.

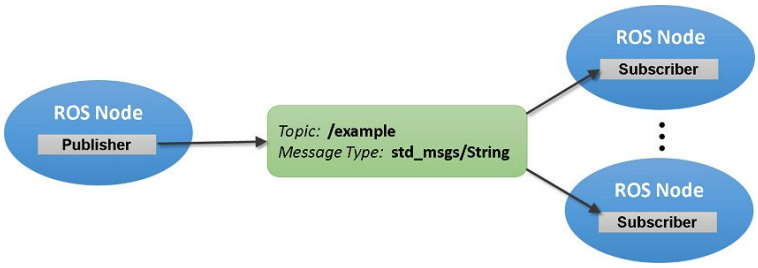


Figure 54: General representation of communication behavior on ROS, built on Publisher-Subscriber relationship. Sensory readings are illustrated by the Topics, messages are the type of data and/or information that available through the entire system.

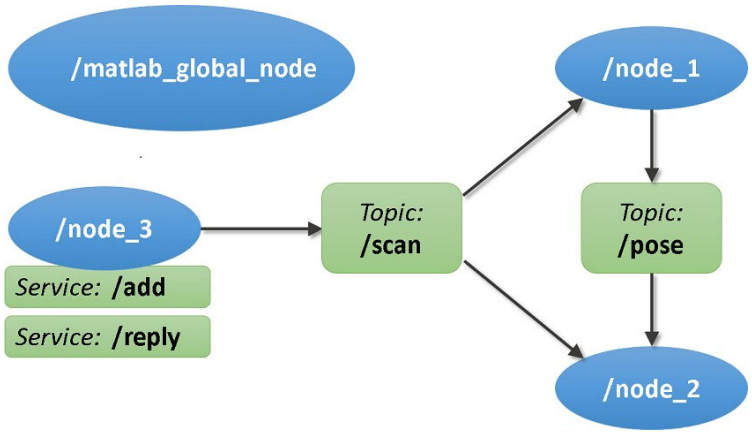


Figure 55: MATLAB Robotic Toolbox provides a global node for MATLAB to communicate to gather any ROS topics and parse the messages throughly.

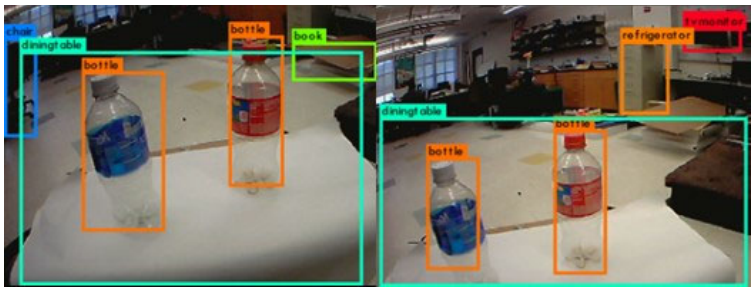


Figure 56: Selected outputs of YOLO demonstrating correct and incorrect classifications of objects in an image frame.

The ArDrone Bebop has a fisheye camera with a software pan tilt mechanism, which is steerable by a joystick or keyboard control. The issue on the drone's camera, enabling the steering mechanism, was a simple fix in the configuration file. Originally, we assumed that the joystick was initialized with 1, however, it was programmed to start if the array of joystick buttons is at 0 by the default, which was not the case for the particular joystick used. Changing the configuration file allowed us to successfully move the camera view to any position. Also, the image was output through the *image_view* ROS node to show vision output.

3.1.4.3.4 Multi-object Tracker for VSLAM In the latter half of the project a deep dive into a multi-object tracker version of VSLAM was developed as an integrated system. The goal for the system was to record locations of objects for mobile systems to navigate with, rather than obtaining massive amounts of data that must be processed offboard on a server or dedicated workstation. Many efforts were pursued to optimize the system and are detailed below.

Integrate Darknet YOLO object detector into ROS The algorithm YOLO (You Only Look Once) is one of the fastest to perform multi-object detection from a single RGB color image frame, Fig. 56. The benefit of using YOLO is that it passes the image only once through the network to identify all, including duplicated, occurrences of the identified objects. It also outputs the locations of those objects in the context of the frame, which can be used in object tracking. Performance of YOLO is highly dependent on the computing platform. For example, on a system with an 8GB NVIDIA GTX-1080 video card with 32GB of DDR4 RAM, a detection can be made in less than 10ms, which is ideal in a real-time applications such as autonomous navigation and driver- less car. Not only by a network access to a powerful computing system, but also integrating these GPUs on-board can make this possible. Fortunately, access to this level of computing is becoming easier nowadays with gaming computers and virtual reality systems.

The source code for YOLO was modified to allow connections to the camera feed through Wi-Fi. The Robot Operating System (ROS) was used to provide Wi-Fi interface to the camera feed over a protocol similar to TCP called TCPROS. To be compatible with ROS, use of a branched version of darknet written in C++ was necessary. This software branch contains the modifications necessary to generate a shared library file libdarknet-cpp-shared.so and arapaho, a C++ API to the library. ROS packages, *ace_arapaho* and *ace_arapaho_msgs*, were developed by the authors to use the arapaho API. These packages provide the capability to publish the identified objects with labels, timestamps, and the relevant region of interest bounds of the image. An additional input to the package is the object filter list. Objects in the list are filtered out from the reported identifications.

Outputs of the *ace_arapaho* ROS node are passed to a feature tracking package developed by the Task T1-4 team, called *ace_object_tracker*. Inputs to this package are parameters from the motion of the robot, parameters of the camera, and the image ROIs from YOLO. This package develops initial models of the detected objects from the inputs provided to it. These models are used to uniquely identify the incoming data as belonging to a unique object.

Initial performance tests were executed with a direct USB 2.0 connection from the desktop computer to a camera onboard a robot. In this configuration, images are processed at about 27 FPS, which is equivalent to the camera frame rate obtained from the USB 2.0 connection. Further tests utilized images transmitted over TCPROS on a Wi-Fi IEEE 802.11N connection from the robot to the desktop computer. In this configuration, YOLO processed image frames around 5 FPS. The drastic reduction in processing rate is solely due to the transmission of raw image data over Wi-Fi. Compressed image streams increase the performance over Wi-Fi back to the 27fps rate for the camera.

Prepare Testbed for Data Collection The Realsense D435 is an extremely compact, lightweight, combination of a stereoscopic infrared camera pair (848x640 at 90fps), an infrared dot projector (directed light), and a RGB camera (848x640 at 60fps). This new type of device has the advantage over other RGB-D cameras as it can be used day or night, both indoors and outdoors, utilizing a fusion of directed light (indoor/outdoor) and stereoscopic infrared modes (outdoor). The nearest competitor for outdoor stereoscopic operation with global shutter is the larger and heavier ZED Stereo Camera, which can only work in the presence of light, essentially making the device only function just during the day. Both are becoming very popular choices for small autonomous systems. For performance comparisons, we have selected both the ZED Stereo Camera and the Intel Realsense D435 camera for two different UGV systems.

The first UGV system to be described is a scale-sized jeep-like vehicle with capabilities to carry payload such as additional computing hardware, UAVs, UGVs and extra battery banks. This relatively slower UGV will act as a ***Mapper Master*** which provides computing and localization support to its smaller, and more agile, ***Mapper Agents***, which can be either UGVs or UAVs. The localization source signals will be placed on-board the ***Mapper Master*** , as a set of HTC Vive Lighthouses directed towards the desired work area for the ***Mapper Agents***. This robotic agent will also be equipped with the ZED Camera, a set of RPLIDAR-A1 units, a 9DOF BNO055 IMU, and a computer equipped with NVIDIA GPU for computing support.

Mapper Agents will receive local coordinates from the ***Mapper Master*** , which are used to navigate within a given area. In this area, ***Mapper Agents*** detect and tag objects of interest with timestamped local coordinates. Local coordinates are related back to the global frame through the ***Mapper Master*** via transforms up to HTC Vive lighthouse as shown in Figure 57. The ROS coordinate transform library **tf2** is used for performing coordinate transformations. Map data obtained by the agents will be filtered for selected objects of interest at each step.

The localization sensor package installed on the ***Mapper Agents*** is formed of two components, the ACE Labs' custom ***AM-L03*** interface board and a sensor frame for housing TS3633-CM1 light sensor breakout modules. The ***AM-L03*** combines a 32-bit STM32F103C8 microcontroller with necessary interface components to support a total of four TS3633-CM1 light sensor breakout modules. Figure 58 shows the sensing package and developed interface board rendering.

Configuration of a Single Vehicle The first goal for testing a single system's performance

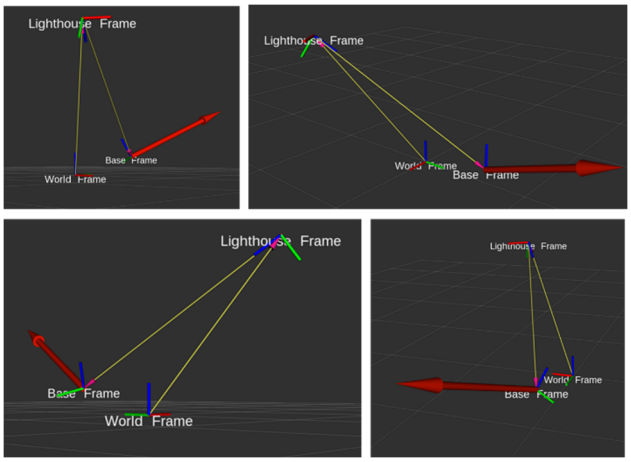
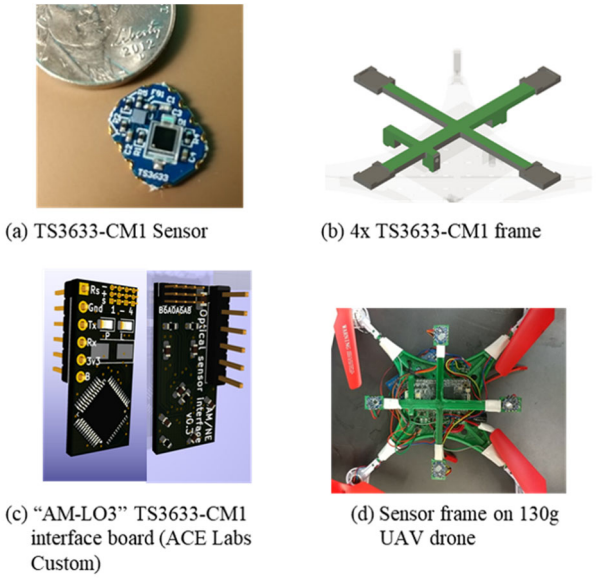


Figure 57: Sample sets of tf2 coordinate transforms generated from testing the AM-L03 board in various poses. Coordinate transforms are visualized in RVIZ (ROS visualization program).



(a) TS3633-CM1 Sensor

(b) 4x TS3633-CM1 frame

(c) "AM-LO3" TS3633-CM1 interface board (ACE Labs Custom)

(d) Sensor frame on 130g UAV drone

Figure 58: Localization sensing package installed on *Mapper Agents*. The sensing package uses a custom designed AM-L03 interface to TS3633-CM1 light sensors.

was to control it through manual stick controls over Wi-Fi using only the image stream as a means to gain visual feedback. This is not a trivial test of system performance, but a crucial first step towards a responsive system of vehicles.

When ROS is used as a remote control input, it can have a typical delay that to the observer, makes it appear as though the control through Wi-Fi is qualitatively "laggy" or "inefficient", with latencies typically over one second. Remote control inputs are generally small amounts of data being transmitted at a high rate. This problem is well-documented as a "small packet" problem that was "fixed" by John Nagle's algorithm to control congestion in TCP/IP networks. Nagle's Algorithm forces a delay until multiple packets of data are combined into a larger TCP frame before being sent out of the network. In a remote control application, Nagle's algorithm works against the goal of reducing a system's latency to increase its reactiveness. There are methods in the ROS that selectively disable Nagle's algorithm on a per-topic basis. Therefore Nagle's algorithm was disabled on the velocity control command topic for each agent, thus providing an optimal reactive response rate to inputs with negligible delay in manual control.

Vision feedback was relatively simple to configure to a usable rate over Wi-Fi streaming. Limits were imposed on the frame size from the Raspberry Pi Camera V2 and the image stream was compressed. Compression was performed using tools from the ROS *image_transport* package that were made available in the Raspberry Pi Camera interface package *raspicam_node* developed by "UbiquityRobotics".

In testing of visual feedback in combination with manual control, an unacceptable random hang-up or freezing behavior was observed on the Raspberry Pi 3b. It was determined that the network interface was being overloaded by the constant flow of image data, plus a potentially erratic stream of event-driven manual control inputs. Limits were added to the manual control rate as part of the solution to alleviate stress on the network interface. The other half of the problem was a network setting in the operating system, the Maximum Transmission Unit (MTU), which at a default setting of "auto" should have automatically accounted for amounts of network throughput. It turns out that the algorithm backing the automatic setting simply can not adjust fast enough before the system becomes overloaded and hangs. Through multiple tests of various values, a MTU size of 500 (bytes) was selected for the Raspberry Pi 3b, and a size of 750 (bytes) on the Jetson TX2. The higher MTU value on the Jetson TX2 is allowable due to the additional compute power that the NVIDIA board can provide.

Configuration of Multiple Vehicles After the tests of single vehicles, unit testing on multiple vehicle configurations was performed. For a background on the logistical process to develop and deploy multiple robots, a single robot interface was developed first in Linux on a Raspberry Pi 3b. Once the configuration was ready, the memory card for the Raspberry Pi 3b containing the operating system (OS) was cloned and imaged to other memory cards. An issue was encountered through the cloning process where the agent's computers using the same OS image would not always connect to the network, or they would connect and then disappear. The issue was resolved by setting the hostname to a unique value (previously all the same) for each of the agents in the system using the same OS image. A script was developed that takes the last three of six octets from the MAC address (i.e. XXYYZZ of AA:BB:CC:XX:YY:ZZ) of the wireless network card into account as the last six characters of the hostname after the hyphen (e.g. nonuniquehostname-XXYYZZ). For reference, the MAC address has two parts: the first three octets (i.e. the AA:BB:CC portion) represent the organization registering a network device, and the last three octets (i.e. the XX:YY:ZZ portion) represent the unique code identifying the specific device's network ID.

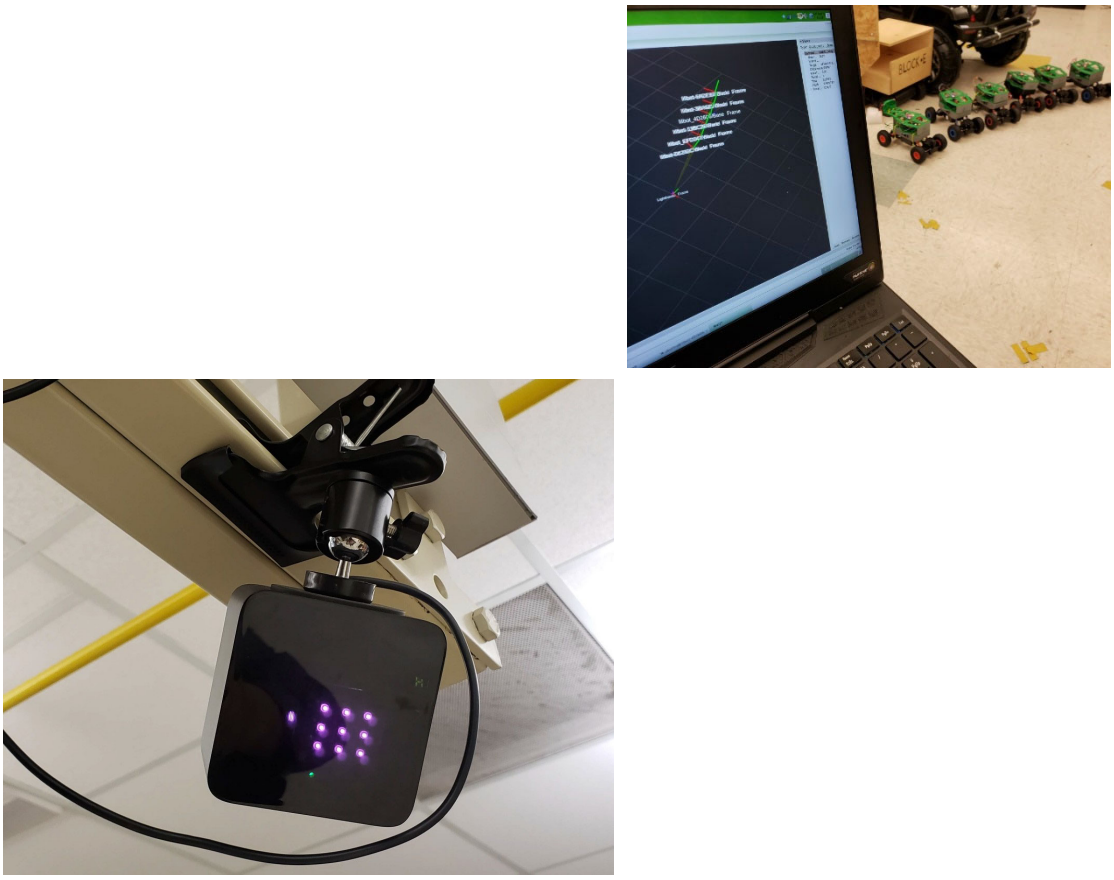


Figure 59: Top: Positioning of HTC Lighthouse in Room, Bottom: RVIZ Showing Multiple Vehicle Agents' Transforms Pointing Back to the Lighthouse Frame and Vehicles on Ground

The solution to the hostname problem actually helped solve a logistical issue in ROS for operating with multiple agents. That issue was in the need for proper namespacing of topics between multiple agents. ROS however does not accept hyphens in its naming convention, therefore underscores are used instead for namespacing topics (e.g. nonuniquehostname_XXYYZZ).

With the proper ROS namespacing complete, the HTC Lighthouse / Custom Tracker localization process had to be modified to allow for multiple agents. This task required modification to the transforms created using the ROS *tf*2 package. Once namespaced, the transform *frame_id* variables all contained the prefix in the style of "/nonuniquehostname_XXYYZZ/" (e.g. "/lil-bot_4D2606/Lighthouse _Frame"). For the "Lighthouse_Frame" to be the same across all agents, the frame name was set to be a global frame "/Lighthouse_Frame". The result of this setting is shown below in Figure 59. From the figure, the robots appear inverted compared to the positioning of the HTC Lighthouse. At this point the next step will be to identify the positioning of the "Lighthouse Frame" relative to the ground, or to the mobile agent it is mounted on.

Use of Multi-object Tracker to Perform SLAM In a multi-agent operation, it is paramount to have an accurate three dimensional map of the objective field so that each agent can cooperatively navigate amongst its peers. This means that each agent needs to be able to communicate its



Figure 60: Identification of a cell phone using darknet_ros

perspective of its field of view to its constituents. Thanks to the recent advances with our lighthouse tracking approach, a grid system can be implemented in order to obtain an omniscient view of all agents in play. With this technology we can accurately process what each agent is seeing and create a cohesive map of the environment by stitching together each individuals readings. In order to illustrate this proof of concept, point cloud data was used to represent the view of each individual agent. The objective was to identify various objects in view and analyze the dimensions of this object so that it could be rendered in the aggregate obstacle map.

Object identification has been widely implemented in various tasks using the open source ROS library ***darknet_ros*** which utilizes a convolutional neural network architecture to make accurate object identification predictions. It is remarkably efficient due to the clever filtering algorithm where pixel patterns are used to approximately identify where in the image frame objects are located and then only analyzing these areas. Using this method, each identified object is surrounded by a bounding box with its respective identification label based on the object's general rectangular pixel dimensions. This is of great value because these dimensions can be used to precisely measure the object's centroid pixel coordinate in the frame. This newly identified coordinate measurement can then be translated to the total obstacle map.

This proof of concept was implemented using the Intel D435 Realsense depth camera which allowed for accurate point cloud representation of three dimensional space. Using the centroid coordinate measured above, the (x, y, z) coordinate could be directly retrieved. Assuming that the camera's fixed position on the agent can be represented by a coordinate of $(0, 0, 0)$, the magnitude of the position vector to the centroid of the object can be calculated to get an accurate measurement of the distance with respect to the respective agent. This measurement can be used to produce a more detailed map of the environment by tracking the distance of these objects with respect to each constituent agent.

Positioning of Lighthouse on Mapper Master The next task was to connect the multiple

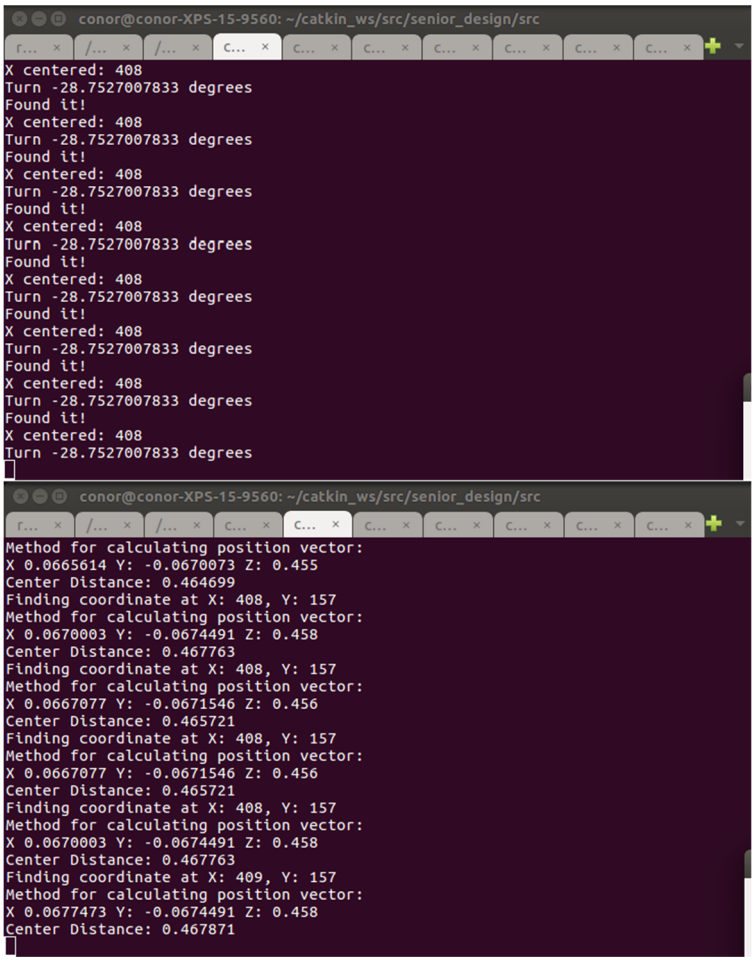


Figure 61:Top: Computing the Centroid of the Cell Phone, Bottom: Mapping the Centroid Coor- dinate to its Respective 3D Point Cloud Coordinate and Computing the Magnitude of the Position Vector

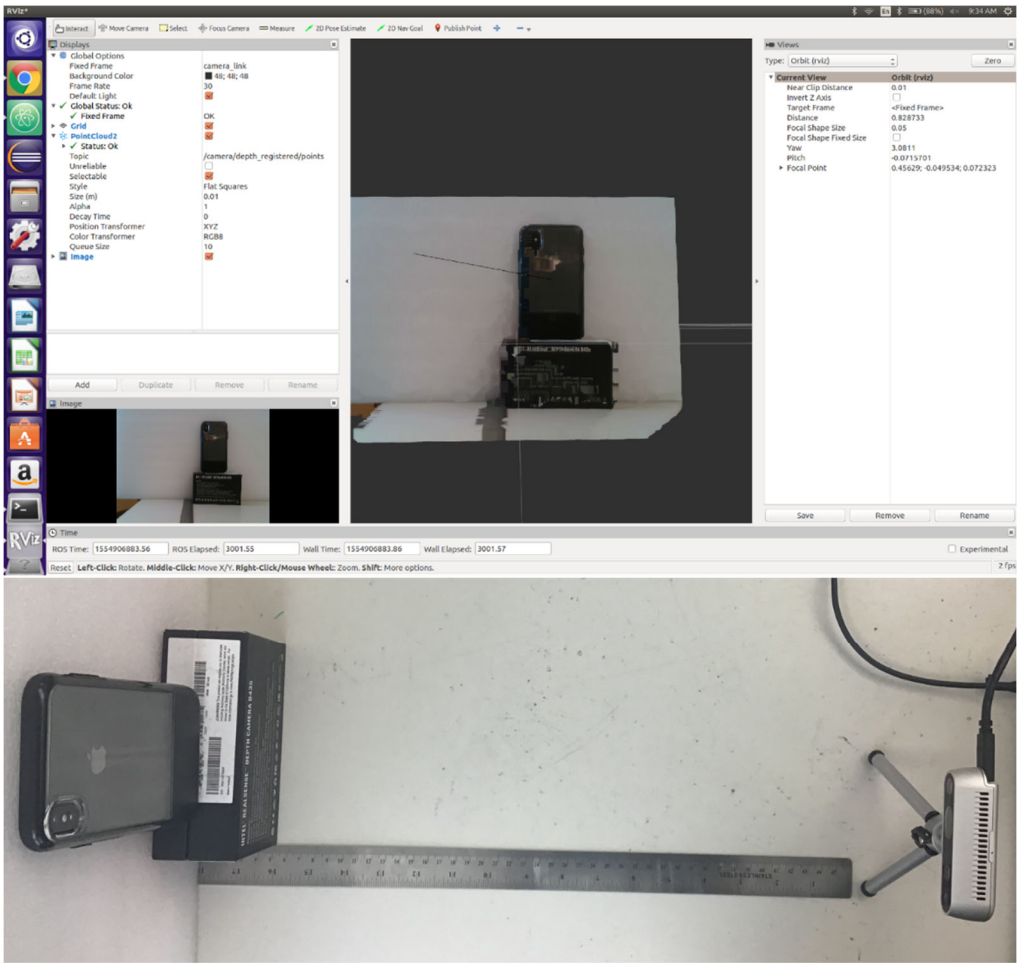


Figure 62: Top: RVIZ Point Cloud Visualization of a Cell Phone Including Accurate Point Mea- surement, Bottom: Testbed to Verify Calibration of The Camera

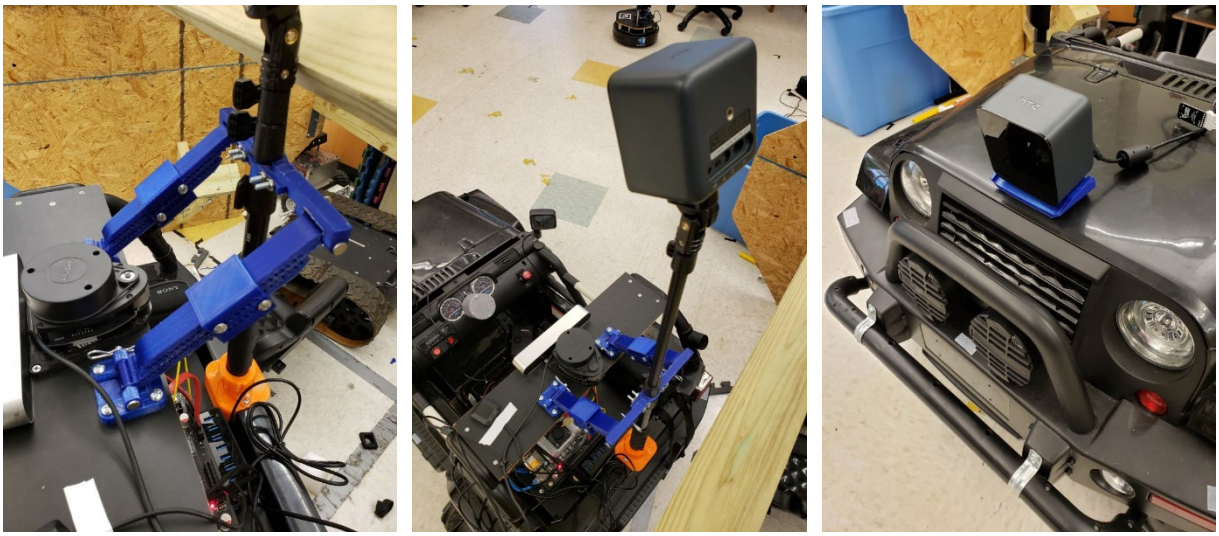


Figure 63: Positioning of the HTC Vive Lighthouse; Left: Brace for lighthouse post, Middle: Lighthouse on post, Right: Lighthouse on front instead of the post.

frames to the lighthouse source frame. For the "Lighthouse Frame" to be the same across all agents, the frame name was set to "/Lighthouse Frame" using the forward slash convention as a marker of a global frame. The purpose of using the lighthouse system was to make it mobile, unlike the common Vicon or Optitrack systems; accordingly, it was mounted on the scale-sized jeep-like vehicle which has the role of ***Mapper Master***.

Initially, to mount the device, the lighthouse was attached to a $1m$-long, extendable post on the back of the jeep-like vehicle to obtain a greater field of projection, and thereby covering a larger area. Multiple 3D-printed braces were made to stabilize the extendable post to prevent swaying while the vehicle moves. This is because, when the pole moves enough, the lighthouse turns off momentarily so as to not damage or overwork the motors providing the sweeping motion of the light pattern. Due to its positioning, the lighthouse was therefore especially sensitive to the vibration of the rear wheels and the swaying of the extendable post; the braces were not enough to counter these problems. In response, the lighthouse was moved to a more stable position on the front of the vehicle. Its new position also improved the reliability of the closest range supported by the lighthouse. Extremities of the supported range stayed relatively the same, as nearly the same angle of incidence of the light towards the ground was retained. Figure 63 shows the various positions tested for the lighthouse.

Positioning of Lighthouse in ROS Software The position of the lighthouse is accounted for in the Universal Robot Descriptor File (URDF). This file also contains the relative positions of all of the on-board sensors of the jeep-like vehicle, which is useful in 3D sensor alignment for data fusion. A series of rotations and translations was needed to correctly position the lighthouse relative to the vehicle and relative to the data obtained from the AM-L03 trackers, such that the tracked vehicles could be accurately placed on the ground. Some pose information in the transformations was measured, while the rest was empirically derived based on the multiple measurements of the multiple ***Mapper Agents'*** poses recorded on a flat surface. Pitch and roll Euler angles defined in the ***Mapper Master*** 's URDF were modified to calibrate to the various sensed positions of multiple ***Mapper Agents***. With multiple ***Mapper Agents*** occupying a known flat plane, the calibration

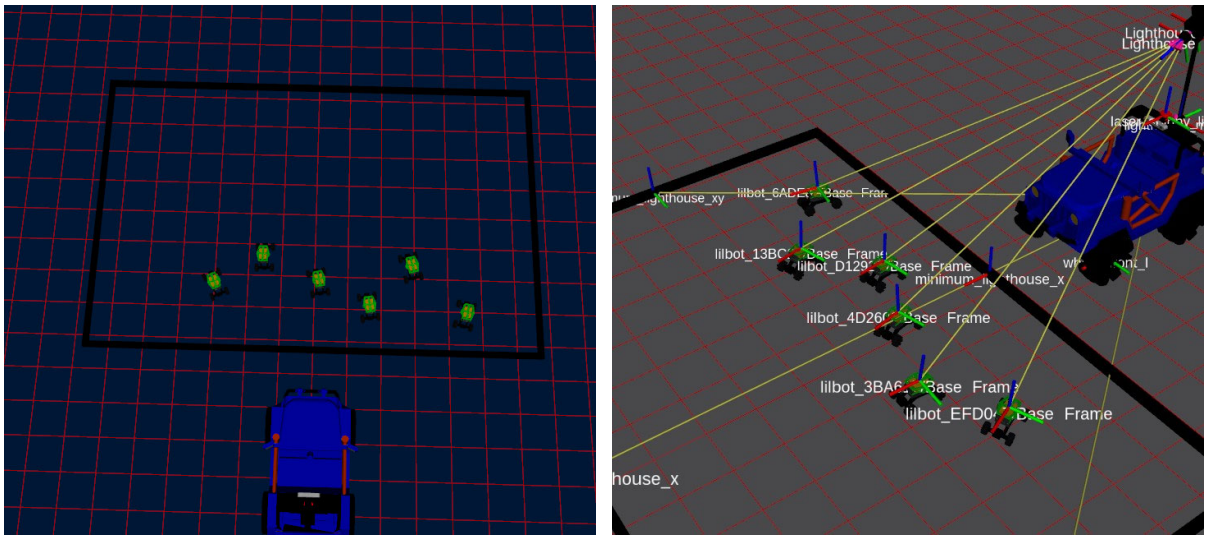


Figure 64: RVIZ showing vehicle positions identified relative to the scale Jeep vehicle. Left: vehicle positions without tf data displayed; Right: with tf data displayed

process was relatively straightforward and could be automated in the future. RVIZ was used to visually validate the pose measurements with respect to ground using the 3D mesh files for the ***Mapper Agents***. Figures 64 and 65 show parts of the process in RVIZ.

Control of Multi-vehicle System in ROS – Several methods exist in ROS to coordinate a set of vehicles. None of these methods are optimized for the purpose, so they are essentially workarounds to achieve the goal. The least optimal method would be an individual topic subscription based algorithm which would wait for data to arrive for all agents in a system. This method suffers from the need of the programmer to have to manage synchronization of all topics and decisions. A more advanced method would be to use the approximate time synchronizer class from ROS "**message_filters**" to grab a buffer of the N agents' pose data at one time. While synchronization is a solid approach, there are many messages that would need to be synchronized and subscribed to at one time if using a desired and actual position as two separate topics for each agent. Alternatively, it is possible through the ROS ***tf2*** package to "look up" transforms between two links that are described in a URDF. These transformations are updated at the same rate as the pose data is sent to the individualized pose topics. These transforms that are sent to the "*/tf*" topic are namespaced using the methods discussed above. This approach was selected over the other two options, for it actively collects data and controls the agents at a known rate, rather than responding to potentially bursty data.

Data Processing of Multi-vehicle System in ROS – A software package named ***zed_darknet*** was selected for investigation due to its compatibility with the ZED Camera (onboard the ***Mapper Master***) and the Darknet software which runs the YOLOv3 deep neural network. Initial tests were run on the ZED Camera, which determined that it does run faster than previous implementations of YOLO on the NVIDIA Jetson TX2. However, the purpose for running YOLOv3 in the system was not just for the ***Mapper Master***, but also for the data coming from the ***Mapper Agents***. The

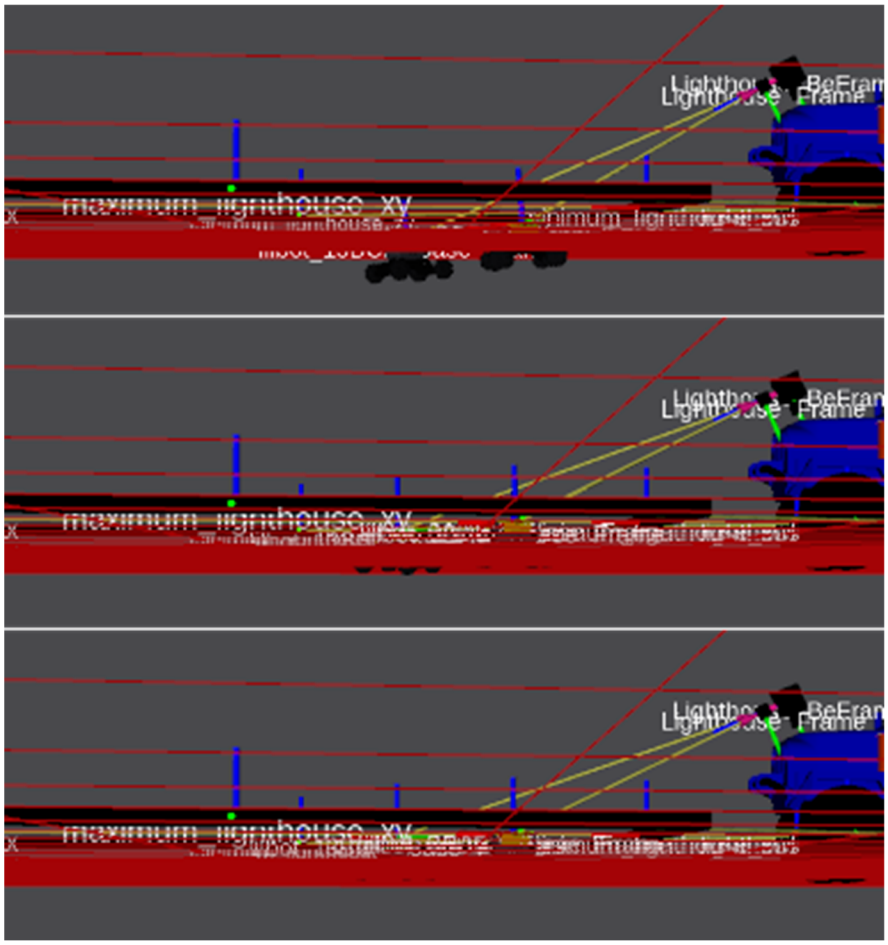


Figure 65: Alignment of the HTC Vive Lighthouse in the URDF at the front position to the red colored grid representing the ground plane. Top: Bad case where tracked vehicles protrude through ground plane, Middle: Tracked vehicles not protruding as much through ground plane, Middle: Tracked vehicles not protruding as much through ground plane, Bottom: Vehicles on ground plane.

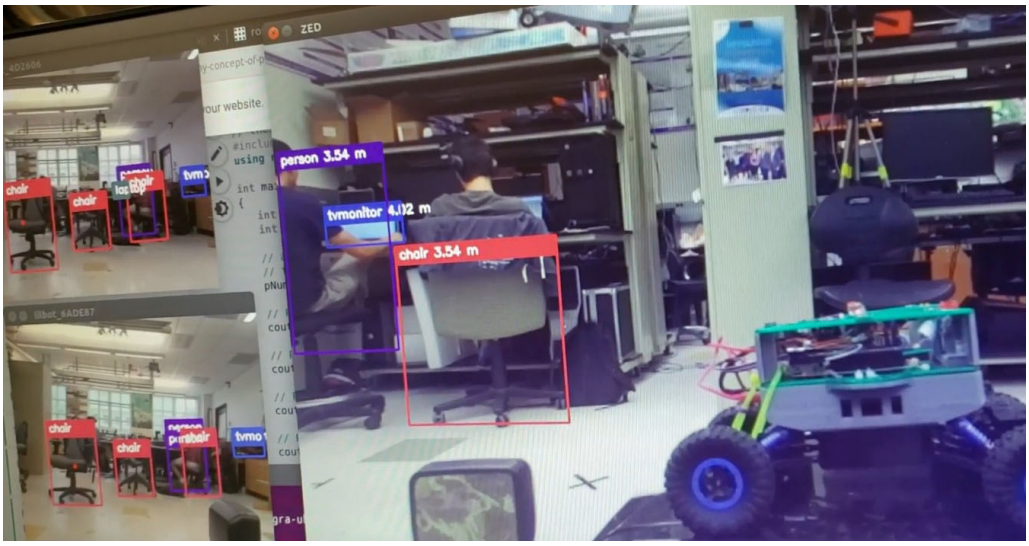


Figure 66: Output of multi-image source YOLOv3 operating on ZED Camera and Raspberry Pi Camera images.

software components to make the Python program into a ROS node were added to enable multiple camera feed data processing in the ***zed_darknet*** package. Support was also added for processing subscribed compressed image feeds. With these components in place, the system is able to process multiple image feeds per second for object classes of interest. Figure 66 shows a sample where the ***Mapper Master*** and two ***Mapper Agent***s are capturing video at one time. Tests of six ***Mapper Agents*** were performed with limited additional overhead due to the limited size of the small frames and compressed image feeds. Compressed image outputs are available for each of the agents in the system with indicators of where the detected objects are. The next step will be to correlate objects among the multiple perspectives obtained from the numerous camera views.

Simulation of Multi-vehicle System in ROS and VR Compatible Unity Software – In the development of the URDF files for RVIZ, visualizations of the robots were made. These visualization files, or mesh files, were exported to Unity. Unity is the next logical step in developing a realistic simulation of the system for testing controllers, multi-vehicle formations, and camera fields of view. Unity has a ROS bridge component that can be used to control the simulation with the same software used to control the robot. Unity also has a component where the HTC Vive VR headset can be used as a controller for the cameras allowing for a person to move around inside the virtual world giving a unique perspective to the performance of the autonomous vehicles. Figure 67 shows scenes from a Unity simulation of multiple vehicles driving in a formation to cover greater areas using multiple cameras.

Cooperative Localization using Monocular Cameras When an object is detected by two "Mapper Agents" then determination of the corresponding intersection of the two vectors oriented from the two cameras to the object can be calculated. The following equations are implemented in python using ROS "PoseStamped" messages, which are time stamped objects that contain a $<x, y, z>$ pose and a quaternion orientation. A pose object located at the camera frame on the vehicle is generated for each object to be cooperatively located. On the particular vehicles, the

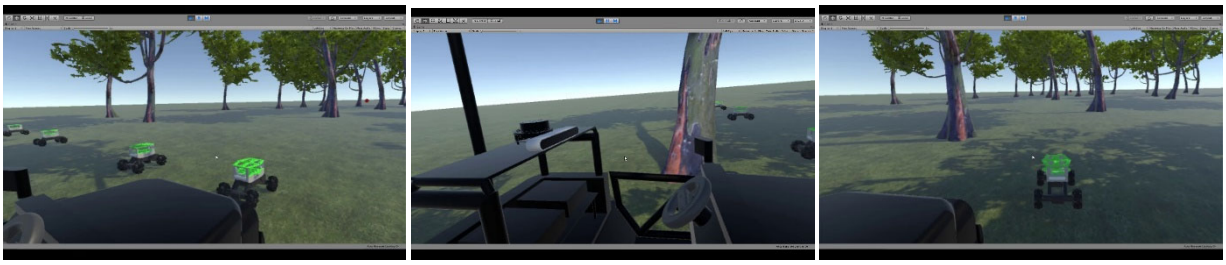


Figure 67: Scenes from a multi-vehicle simulation in Unity with vehicles driving in a line formation.

camera used is a monocular type which does not provide depth information. From the two poses and orientations, x_1, x_2, Q_1 and Q_2 can be obtained through ROS "tf" functions to lookup the transform to the front of the vehicle and perform the rotation of a pose vector towards the object. where the quaternions have the following representation:

$$Q_1 = a_1 + b_1\hat{i} + c_1\hat{j} + d_1\hat{k}$$

$$= a_1 cos\frac{\theta_1}{2} + (b_1\hat{i} + c_1\hat{j} + d_1\hat{k})sin\frac{\theta_1}{2}$$

$$Q_2 = a_2 + b_2\hat{i} + c_2\hat{j} + d_2\hat{k}$$

$$= a_2 cos\frac{\theta_2}{2} + (b_2\hat{i} + c_2\hat{j} + d_2\hat{k})sin\frac{\theta_2}{2}$$

and by setting the coefficients $b_1 = c_1 = b_2 = c_2 = 0$ as we are interested in the rotation about the z-axis, we get:

$$Q_1 = a_1 cos\frac{\theta}{2} + d_1 sin\frac{\theta}{2}\hat{k}$$

$$Q_2 = a_2 cos\frac{\theta_2}{2} + d_2 sin\frac{\theta_2}{2}\hat{k}$$

To simplify use of the quaternions components, we define the following terms r_1, r_2, s_1, s_2 to represent the transform from the quaternion to a more convenient vector compatible format:

$$r_1 = a_1^2 - d_1^2$$

$$r_2 = a_2^2 - d_2^2$$

$$s_1 = 2a_1d_1$$

$$s_2 = 2a_2d_2$$

Positions of the target within the two camera frames, $\boldsymbol{x_1}$ and $\boldsymbol{x_2}$, are given by:

$$\boldsymbol{x_1} =< x_1, y_1 >$$

$$\boldsymbol{x_2} =< x_2, y_2 >$$

with the corresponding instantaneous trajectory vectors are $\boldsymbol{p_1}$ and $\boldsymbol{p_2}$:

$$\boldsymbol{p_1} =< r_1, s_1, 0 >$$

$$\boldsymbol{p_2} =< r_2,\ s_2,\ 0 >$$

The vectors normal to the trajectory vectors are $\boldsymbol{n_1}$ and $\boldsymbol{n_2}$.

$$\boldsymbol{n_2} =< s_1,\ -r_1,\ 0 >$$

$$\boldsymbol{n_2} =< s_2,\ -r_2,\ 0 >$$

Line vectors described by the following two equations are given in d_1 and d_2,

$$s_1(x - x_1) - r_1(y - y_1) = 0$$

$$s_2(x - x_2) - r_2(y - y_2) = 0$$

which can be rewritten as the following in matrix format:

$$\begin{bmatrix} s_1 & -r_1 \\ s_2 & -r_2 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} r_1x_1 - s_1y_1 \\ r_2x_2 - s_2y_2 \end{bmatrix}$$

solving for the intersection of the two vectors, which are located at $< x, y >$:

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} s_1 & -r_1 \\ s_2 & -r_2 \end{bmatrix}^{-1} \begin{bmatrix} s_1x_1 - r_1y_1 \\ s_2x_2 - r_2y_2 \end{bmatrix}$$

The following condition prevents two agents from having the same orientation with respect to the object:

$$s_2r_1 - s_1r_2 \neq 0$$

which expanded with the definitions of s_1, s_2, r_1, r_2:

$$cos(\theta_1)sin(\theta_2) - cos(\theta_2)sin(\theta_1) \neq 0$$

reduces to:

$$sin(\theta_2 - \theta_1) \neq 0$$

Figure 68 shows two vectors from the two camera frames intersecting with one another. In the figure, the resulting intersection point is marked with another vector pointing back towards the y-z plane with magnitude only in the x-component of its vector. The intersection position (x, y) can then be recorded with the ROS "tf" transform broadcaster. Measured dimensions of the object are represented with markers generated for visualization in RVIZ, which are centered on the frame of intersection's transform. With the detection bounds from the deep neural network object detector, dimensions can be estimated more correctly with the two visual sources of information. Ellipses can represent the measurements of object size with the expected uncertainty of the measurement. There are some issues to be worked out with this approach. As the method for determining the class of an object uses deep neural networks, the co-localization of a unique item in an environment should be simple. For more complex environments where an object is commonly found, a disambiguation routine will need to take place to sort out what each robot is seeing where. This routine will need to take into account relationships between clusters of object classes for tracking and discrimination between detected items. Another issue that can arise is that the agents may see multiple objects of a certain class in the same area with different perspectives as one unique thing through one of

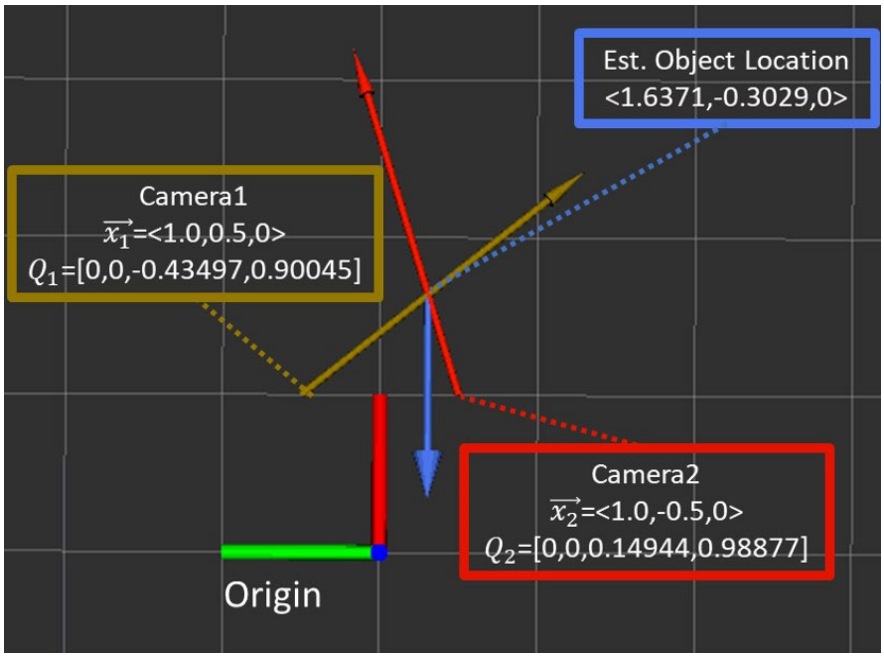


Figure 68: Intersection of two vectors coming from two camera frames. The intersection marks the estimated location of a detected object's centroid.

many problems. First, there could be a physical barrier between two objects and both agents see something of the same class on both sides of the separator. Another problem can come from really small items that look like they are one item as they are close to each other. These problems are issues as we intend to utilize multiple robots to perform context-aware mapping of detected objects for reuse by mobile embedded systems.

Simulation and Visualization of Camera Properties – Field of view and maximum range are handled in the simulation as an angular sweep in front of the "Mapper Agent" robots, where objects must fall within this area to be detected by the particular agent. A "PolygonStamped" type message was used to draw the area in RVIZ for each of the "Mapper Agent" robots. One message is published for each simulated robot.

Simulating Detection of Objects – Objects of interest were simulated by using a "PoseS- tamped" ROS message type to hold a pose that was used as a centroid location. Transformations from the "odom" frame to the "Mapper Agent" robot pose and the various object poses were used in calculations to determine an estimate of the object location. Object pose estimates are set at the same position the robot camera pose, but with an orientation towards the object of interest. Validity of estimates are checked against camera properties and mathematical restrictions of the functions utilized. With the possible valid estimations calculated in the simulation, all pairs of esti- mates are collected using the function "combination" from the "itertools" package in python. Each pair is input to the algorithm and the output pose estimate is recorded using a visualization marker, and optionally saved to a file for offline plotting of a map. Plans to improve the process include clustering the object pose estimates for filtering multiple observations into a single data structure

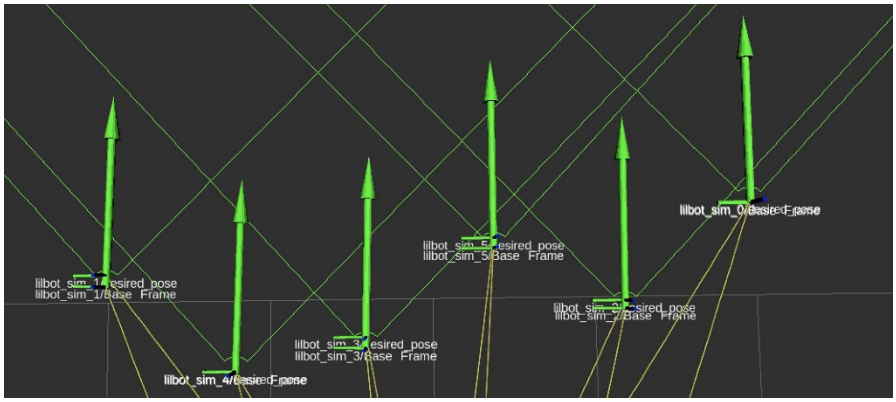


Figure 69: Improved simulator performance at eliminating steady state error in the heading.

to store in the map that provides context of the detection.

Simulation of Robot Motion – Improvements were made to the controller used in the "Mapper Agent" 's Ackerman steering dynamics simulator to correct the sources of heading data used for alignment of the vehicle to both the desired position and the final heading once at the destination. Figure 69 shows various simulated agents reaching their desired poses with better performance than previously acheived.

Simulation of Noise and Quantization Errors – Orientations towards the objects of interest are quantized using a rounding scheme to model anticipated effects of measuring centroids from images using pixel measurements. Noise in pose measurements of the location is achieved through applying noise to the estimated position of the "Mapper Agents" for simulating the measurement from the light sensors measuring the HTC Vive Lighthouse infrared light sweeps. Noise in the measurement tends to have a larger effect on the x-axis (forward is positive) than the y-axis (left is positive). This noise characteristic is present in the simulation.

Visualization of Algorithm Calculations – To visually mark poses in the simulation, a "VisualizationMarkerArray" type message was used to efficiently render multiple shapes. Cylinder shaped markers were used to visualize the known object locations and estimates for those object's locations. Arrow shaped markers were used for the directional estimates of "Mapper Agents" positions and orientation estimates towards objects of interest. One single message is published for all objects, resulting in one batch of changes per rendering update. Figure 70 shows an instantaneous capture of the algorithm output visualization RVIZ. Figure 71 shows an instantaneous view of the output when only one pillar is detected by two agents.

Object location data published by "darknet-ros" is then subscribed to by the python program that aggregates data and manages the "Mapper Agent" platforms as a group. In the group manager software the bounding box topic of "darknet-ros" is used to examine the detected objects. Objects detected by "darknet-ros" were filtered for the list of reliable objects that could be detected each frame. In the particular area of the laboratory that we tested, the class of "bottle" was the most reliably detected object. Figure 72 shows an example of the output from YOLO V3. It has some less than desirable detections based on a number of factors including the camera resolution and distance from the objects being detected.

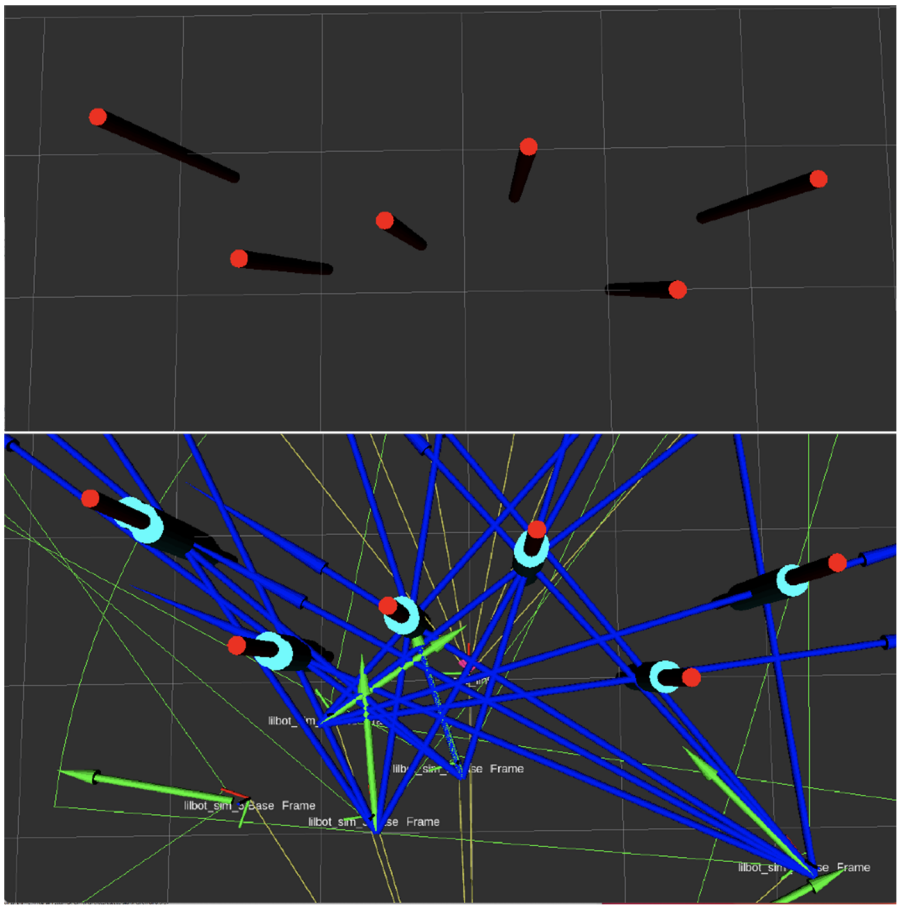


Figure 70: : Scene from RVIZ showing an instantaneous view of multi-robot simulation cooperative detection of multiple objects. The top image shows the location of 6 detectable objects, visualized as red pillars. The bottom image shows the objects as red pillars, various orientation estimates as long blue arrows originating from the robot pose, detected object pose estimates as light blue cylinders (multiple pillars are overlapped), "Mapper Agent" headings as short green arrows, and camera detection range as thin green outlined arc shaped areas.

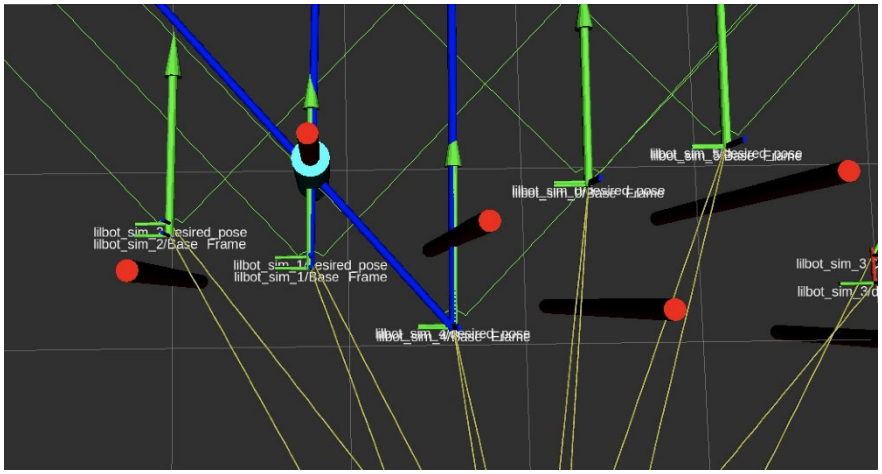


Figure 71: Scene from RVIZ showing an instantaneous view of object detections where two agents are currently detecting one object. The remainder of pillar objects have either one or no current detections.



Figure 72: Image from "Mapper Agent" showing multiple detections using "darknet-ros" package for YOLO V3 from two separate agents' viewpoints. The left image shows the view from the robot on the right looking towards the left, which shows one positive identification of a bottle. The right image shows the positive detection and one false-positive detection of a bottle on the right image.

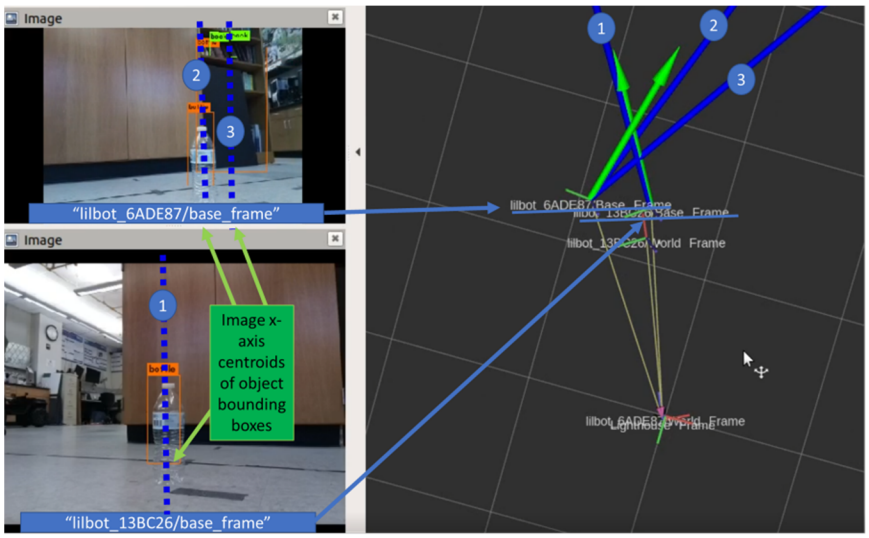


Figure 73: Scene from RVIZ showing an instantaneous view of object detections where two agents are currently co-detecting one object (vectors marked "1" and "2" and tracking a false-positive object with a vector marked "3".

Solely from the detections provided in the figure without depth data, there are issues in resolving proper co-location of a single object. Possible methods to resolve the issues in co-detecting objects is 1) use depth camera, 2) use pattern matching techniques (previously used oriented-fast and rotated brief (ORB) features in the first two years of the project) and 3) more "Mapper Agents" pointing towards an object. The first method would likely be selected if emitted radiation (directed light-based depth cameras and LIDAR emit light) is allowable. It can also be done by one agent, which could be the "Mapper Master" robot. The second method mentioned would be applicable if the objects of the same classes do not share much of the same features. The third method is a promising option if the additional agents do not also detect the same false-positive class identifications, unless the false-positives are quite reliable data to be obtained. Locations of vector intersections can be clustered together to verify that an observation is being made at the correct location.

Of the various methods suggested to reduce the effect of the false-positives, filtering and clustering and depth estimation were identified as promising methods for the monocular camera use case. An effort was taken to characterize the false-positives from the vector intersection logic with a larger count of agents using simulation. The ROS-based multiagent simulator was modified to exhibit the characteristics that were experienced in the physical experiment, but with six virtual vehicles instead of two physical vehicles. Assumptions were made in the simulator to account for performance of the object detector with a typical image resolution which affects the minimum and maximum detection distance for a simulated observation and the vector intersection logic. Additional measures were taken to ensure only vector intersections in front of the vehicles were considered as being correct. Detection vectors from six simulated vehicles to six objects created a large number of vector intersections as can be seen in Figure 74. Important observations from this trial run are the objects at the extremities of the work environment have few if any vector intersections, and there are a significant number of false-positives in the vector intersections.

Figure 75 shows the cumulative candidate object detections from the simulated agents driving

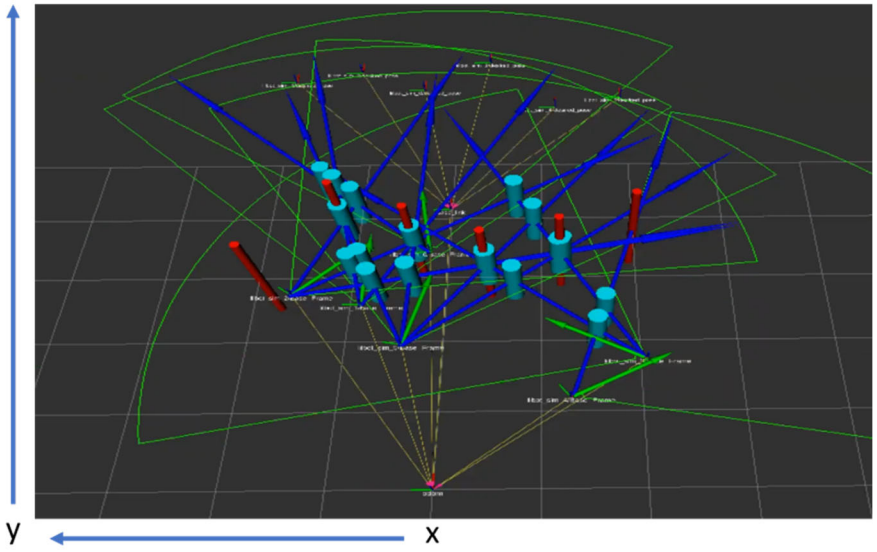


Figure 74: Image from RVIZ showing a view from the simulation where red cylinders are true object positions, dark blue arrows are vectors towards detected objects, light blue cylinders are object detection candidates (detection object vector intersections).

through the object area. The tallest peaks correlate with objects detected most reliably. Small peaks generally match with the false positives. The two true object locations not detected in the freeze frame of the simulation shown in Figure 74 do not show up as a significant number of observations. When the vehicles have a non-zero velocity, changes in observations of vector intersections between two steps for positive detections are minimal about a center location (relatively static in position) and false-positives move along in a linear fashion.

Methods that were researched for removing false-positives from the pool of detected object candidates included the following:

- Changes in detection characteristics were examined as a potential route to filter the observations.
- A rule-based filtering process to ensure every detection vector only is included in one candidate detection or intersection, while still allowing multiple detection vectors to result in one intersection.

The goal of the rule based filtering is to remove the extra candidate vector intersections. This is depicted below in Figure 76 where all red circles with a slash indicate an incorrect result. Correct results are open randomly colored circles at the middle of vector intersections. The colors of circles and vectors indicate a unique object to be tracked. In the figure most, if not all, incorrect candidate vector intersections can be thrown out based on the change in time step with the assumption that an agent vehicle is moving.

The assumption of a moving vehicle failed in testing of the time-based method for filtering data, where the failure is attributed to the computational load of rendering the markers in RVIZ. Two reasons were identified for the problem, running the simulation in a virtual machine environment

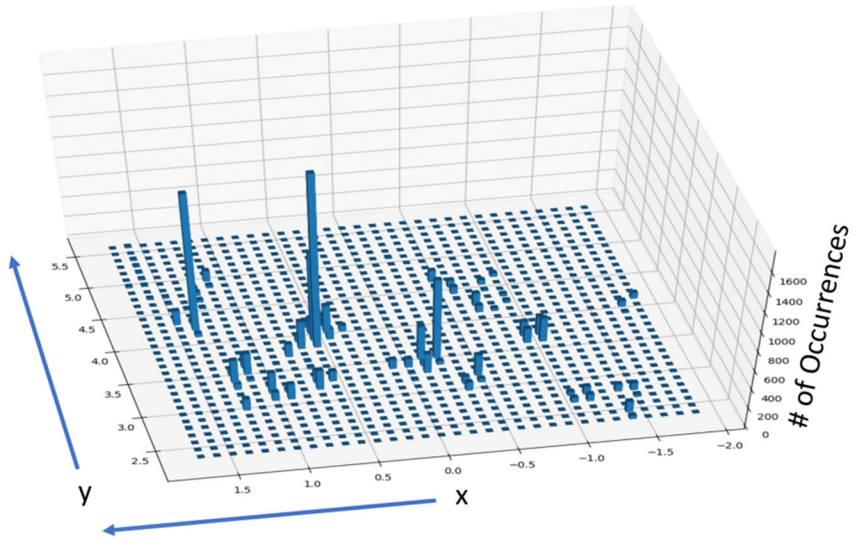


Figure 75: 3D histogram of recorded data of all vector intersections recorded where x and y axes have unit in meters.

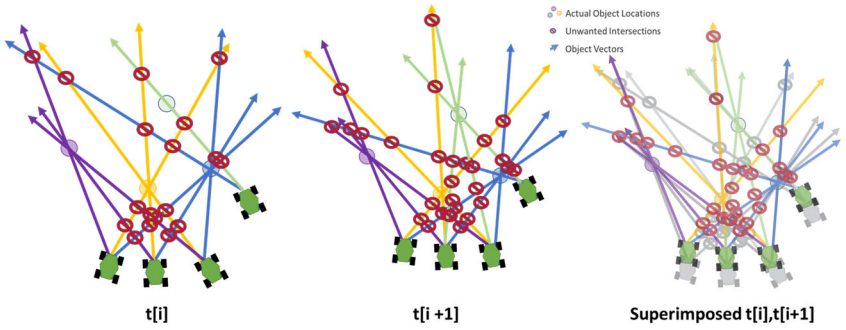


Figure 76: Detection and intersection vectors for four simulated agents over two time steps.

(limited resources) and rendering too many candidate intersections. To fix the first problem, the computer running the simulations is being reconfigured to be more capable of running the simulation. The second problem requires a re-work of the algorithm to satisfy technical limitations of the simulation and improve robustness of the filter process.

At the point where three or more simulated mapping agents are in the same region with the six simulated objects the number of candidate intersections from detections (rendered objects) becomes too high for RVIZ's visualization markers to be shown in the virtual machine without bogging down the simulation. As the simulation gets slower due to more complex rendering instructions, the time-based filtering method fails to acquire good results as the simulated agents travel less by taking more time in each simulation step. It allows small motions in a longer time span to undesirably be considered as detection candidates by the filtering technique. Changes in value for both time and distance must be measured to make the filter technique be compatible with both the real world and simulations. An assumption that distance would be sufficient would not account for spurious noise events in pose readings that can occur if line of sight is lost between our current localization signal producer (HTC Vive Lighthouse) and sensors (TS3633). A constraint requiring a change in position is not rare in mapping techniques.
The process for each iteration is outlined as follows:

- For each observation (vectors from a pair of agents):
  - Pairwise computation of intersection of two agent's observations
  - Calculate distance between intersection location and both agents
  - If within reasonable limit (i.e. $dist < 3m$):
    - Distance calculation from past observations (pair of agents)
    - If within small threshold (i.e. $dist <= 3cm$):
      - Same object as before, confidence_vote += 1
    - else if within large threshold (i.e. $dist <= 10cm$):
      - Likely not the same object, confidence_vote -= 1
    - else:
      - New object, confidence_vote = 0
- For object candidates:
  - if confidence_vote > min_confidence_threshold:
    - Plot observation in RVIZ
    - Register in map

The next steps in this process for future work is the following:

- Implementation of the rule based filtering process in ROS python code
- The addition of logic to filter candidate detections by ratios of detected object pixel width to source image width may improve weeding out false-positive vector intersections.

3.1.4.3.5 Summary A set of approaches to handle cooperative localization and mapping on a cloud based environment. Multiple SLAM algorithms were investigated over the course of the project and detailed in quarterly reports. Cloud and edge-based mapping approaches were explored with a focus on reaching a real-time or near real-time mapping capability. Notably, the RTAB-MAP package produced visually rich point cloud maps at a cost of heavy local GPU load. This approach could be useful with advancements in local and edge compute capability with GPU, and advancing communication standards for offloading data to the edge for computation. Many of the existing approaches would fail to be useful on a resource constrained device that needs to utilize the map generated by the algorithm. Promising approaches for future exploration were developed for multi-agent mapping using highly accurate beaconing for formation control.

In the multi-agent deep neural network enhanced localization and mapping objective, advancements were made in the tools and methods used to manage formations and the information gained from them. Formations were simulated in more depth using ROS and RVIZ visualization tool. Mapper Agent controllers were developed and tuned for reaching desired poses relative to the Mapper Master. A manager for the odometry of the Mapper Master vehicle was developed which will be used to localize detected objects. A method for finding estimated object poses was introduced for co-locating objects between multiple agent's cameras which required localization, coordination and image processing capabilities. Localization of multiple vehicles was obtained by careful calibration of modeled coordinate transforms on the master vehicle which is the signal positioning source. Coordination of multi-vehicle motion is enabled through use of ROS coordinate transform listeners, where each vehicle in the network has access to system-wide positioning information on other agents. A test of system-wide image processing capabilities was performed onboard the master vehicle with six mapping agents using YOLOv3 on seven image streams. Characterization of false-positives for multiple vehicles was performed in simulation.

3.1.5 Project Progress for Task T1-5 (Hierarchical Hybrid Cooperative Control of LSASV)

3.1.5.1 Period of Performance under

Task T1-5 Start Date: August 2014

Conclusion Date: April 2021

Faculty lead: Dr. Ali Karimoddini

3.1.5.2 General Description of Task T1-5

Although a cooperative strategy will enhance the reliability and effectiveness of the team, the overall structure will be a highly complex system with many interacting subcomponents. One of the main sources of the complexity is the hybrid nature of the system due to the coexistence of the decision-making mechanism (with event-driven dynamics), and the low-level control of the system (with time-driven dynamics). In such a hybrid structure, the event-driven and time-driven dynamics of the system not only coexist but also interact with each other, and their coupling effect cannot be ignored. In addition, this complex structure must include humans in the control loop. Hence, given a team of heterogeneous autonomous systems and humans in the control loop, a challenging

problem is how to design and analyze such a hybrid decentralized control structure to comprehensively capture the dynamics of all subcomponents and their interactions. To address this problem and to overcome the complexity of the system, we propose a hierarchical hybrid control structure to distribute the complexity and control tasks among the layers and among the components. Hierarchical control systems have been studied for quite some time; however, considering the concept of hierarchical control within the hybrid modeling and control framework and its application to autonomous systems have not yet been addressed. Moreover, the role of the human in terms of high-level supervision and goal setting has not been captured. Recently, we have developed a hi- erarchical hybrid control structure and applied it to a helicopter. The TECHLAV, however, will leverage our previous work by developing a decentralized hierarchical hybrid structure for control and coordination of a team of heterogeneous autonomous vehicles to cooperatively accomplish an assigned mission under human supervision.

The subcomponents of this control hierarchy would include but not be limited to the vehicle dynamics, continuous low-level controller, planning unit, and supervisory unit. In this hierarchy, each layer by itself can be modeled as a hybrid system. A hybrid system typically consists of a set of discrete states standing for operating modes, each of which has time-driven (continuous or discrete time) dynamics. To develop this control hierarchy, we first will use a formal method to connect the hybrid subcomponents and then synchronize them to collectively form a composed structure. This can be done by properly capturing the input and output of the layers, and by synchronizing and treating the discrete states, discrete transitions and continuous transitions. In this structure, the human operator assigns the mission. The assigned mission given in the form of Linear Temporal Logic (LTL), which is very close to the human language, will be translated to a finite state machine for which we will design and implement a supervisory controller in a decentralized way. This project aims at developing a decentralized hierarchical hybrid cooperative control framework for task allocation and coordination of a team of autonomous vehicles to address more complex tasks.

3.1.5.3 Objectives for Task T1-5 over the Course of Project

- **Objective Name:** Development of algorithms to design a hybrid controller for autonomous vehicles to accomplish a high-level and complex task

  Objective Type: Research

  Objective Description: Reach-avoid problems, with either dynamic or static formulation, is a very complex problem. To solve this problem, we proposed to first obtain an equivalent abstract model of the continuous dynamic problem by formulating it into an LTL formula with boolean variables, and then we developed a game-theoretic algorithm to solve the high-level problem. I is proven that this high-level solution can be correctly converted into a continuous domain solution. A complete discussion on the process, proof of existence and correctness of such solutions, and complexity analysis of the algorithms are also provided.

  Impact: By designing automated algorithms to solve a very high-level objective like “Reaching a set of states and avoiding known or unknown set of states”, we accomplished the objective of keeping human out of the design loop. This design process for the complex dynamic systems including autonomous vehicles, and the complex objectives used to be almost impossible for a human or machine. The derivable of this task, on the other hand, has made this task possible.

3.1.5.4 Progress Against Planned Objectives in Task T1-5

The objective of this task is accomplished. Several algorithms are developed to:

- formulate the reach-avoid problem into a LTL formula representing the high-level problem and objectives
- solve the high-level problem and provide a solution that guarantees achievement of the high level objective
- convert the high-level solution of the problem into its original continuous domain

Besides, several formulations of the problems are addressed to illustrate the powerful capability of the problem in solving complex problems with a slight modifications. These scenarios are single target reach-avoid problem and multi-target reach-avoid problem.

3.1.5.5 Technical Accomplishments in Task T1-5

To accomplish the objective of this task, a correct-by-design controller is developed for the vehicle under control to respond reactively to adversarial actions of the opponent vehicle, while also avoiding static obstacles and no-fly zones. A particular targeted problem is path planning and control of autonomous vehicles involved in a dynamic adversarial reach-avoid scenario. In the studied scenario, there are two non-cooperative vehicles with the competitive objectives "reaching a target and avoiding the other vehicle" for one of them, called attacker, and "protecting the target and capturing the opponent vehicle" for the other one, called defender. To address this problem, the environment is partitioned into finite number of disjoint regions, which results in an abstract discrete version of the problem, allowing for formally capturing the assumptions about the environment and requirements of the vehicle under control using Linear Temporal Logic (LTL) formulas in the form of General Reactivity(1). The developed method can handle the reactive nature of the reach-avoid problem while synthesizing a discrete supervisor that is guaranteed to win the game. Using a temporal game structure and mu-calculus formulas, two algorithms are developed to find discrete strategies that guarantee both safety and reachability. The approach is also extended to the multi-target dynamic reach-avoid problem. Then, a novel correct-by-design hybrid controller is proposed to generate control signals for executing the winning strategies by driving the attacker to win the reach-avoid game while respecting the dynamics of the system. A salient feature of the proposed hybrid controller is the smoothness of the generated control signals, which is achieved by the proper control of the vector field when the system moves over the edges of partitions while guaranteeing the execution of the high-level discrete commands. The method is also extended to incrementally forward complete systems to handle autonomous vehicles with more complex dynamics.

We consider the dynamics of the attacking autonomous vehicle modeled with a particular class of multi-affine nonlinear systems of the form $\dot{x} = f(x) + Bu$, which describe well-known models like Euler, Volterra, Lotka-Volterra equations, attitude and velocity control systems for autonomous vehicles such as aircraft and underwater vehicles. For the projection of the discrete strategies to continuous trajectories in a symbolic control framework over a partitioned region, when the system's trajectory enters a new region, a new discrete command is issued to the system to drive the system to the next desired partition. The resulting vector field over the partitioned space is then changed, which may cause a discontinuity of the control signal. Such discontinuity in multi-affine nonlinear

systems has been observed and discussed in [10]. The abrupt control signal changes may cause a discomfort, negatively impact the actuators and reduce the reliability of the system [11–13]. Our proposed solution is also able to handle this problem efficiently.

Finally, a framework is development to design a hybrid controller for a more general class of nonlinear systems. The previously proposed hybrid controller has been developed for the autonomous vehicles with modeled dynamics modeled as multi-affine systems. However, many systems dynamics cannot be modeled as a multi-affine function. Therefore, we propose to developing a bi-level abstraction technique which could be feasible for many systems which are incrementally forward complete [14]. This include many important class of robots including nonholonomic systems.

In summary here is a list of the main accomplishments for this task:

- Formulating the dynamic reach-avoid problems using temporal logic formulas to apply formal method techniques for developing correct-by-design techniques,
- Obtaining the guaranteed winning initial regions for the attacker vehicle to start the reach-avoid mission,
- Developing formal efficient algorithms to generate winning strategies for an autonomous vehicle in a dynamic adversarial environment,
- Introducing a novel interface for generating smooth control signals for the system to accomplish high-level commands, Implementation of the results on actual systems,
- Extending the introduced framework to a multi-target dynamic reach-avoid scenario when the attacker is required to visit a sequence of targets while avoiding the defender.

Next we discuss multi-target dynamic reach-avoid scenario.

3.1.5.5.1 Problem Formulation The vehicle's dynamic in multi-target dynamic reach-avoid (MTDRA) scenario, is considered as follows:

$$\dot{x}(t) = u(t) \tag{24}$$

where $x(t) \in P \subset \mathrm{R}^2$ is the position of the vehicle within a bounded 2D operation region P, and $u(t) \in U \subset \mathrm{R}^2$ is the control input. We assume that both vehicles move with maximum velocity of V_m.

The operation takes place in a region represented as a bounded set $P \subset \mathrm{R}^2$. To manage the complexity of the problem and to employ symbolic motion planning techniques, we partition the environment P into finite disjoint rectangular regions P_{ij} such that:

$$P = \bigcup_{i\in\{1,\cdots,n\}, j\in\{1,\cdots,m\}} P_{ij} \tag{25}$$

$\cap \quad = \emptyset \qquad (\quad) \neq (\quad)$

where $P_{ij} \quad P_{lk} \qquad$ for all $i, j \qquad l, k$.

To capture real-time discrete position of both vehicles over these partitions, we define attacker Boolean sensors a_{ij} and defender Boolean sensors d_{ij}, over partitions, P_{ij}, $i \in \{1, \cdots, n\}$, $j \in \{1, \cdots, m\}$. The truth evaluation of these Boolean propositions changes whenever the vehicles move to a different partition. Since they can only be in one partition at a time, the sensor variable with the same index

of that partition, where they are located in, is *True* and the rest are *False*. We define sets A and D containing these variables as follows:

$$\begin{aligned} \mathrm{A} &= \{a_{ij}\} \quad , \quad i \in \{1, \ldots, n\}; j \in \{1, \ldots, m\} \\ \mathrm{D} &= \{d_{ij}\} \quad , \quad i \in \{1, \ldots, n\}; j \in \{1, \ldots, m\} \end{aligned} \tag{26}$$

We assume that the number of targets is N_t and they should be visited in a pre-determined order. For each target $k \in \{1, \cdots, N_t\}$ located in P_{i_k,j_k}, we define a Boolean proposition t_{i_k,j_k}, and the set T, which contains all these propositions. Finally, we define a proposition "*Accomplished*" which is *True* when the attacker visits all the targets. Having this information, we can then capture all the assumptions, requirements and objectives of the vehicles as LTL formulas in the form of General Reactivity(1).

The multi-target dynamic reach-avoid (MTDRA) problem can then be described as follows:

Problem 1. Consider an attacking vehicle with the continuous dynamics in (24) driven over the region P described in (25). Also, consider the attacker objective is to visit targets with positions $x_{t_1} \cdots x_{t_N} x_{t_k} \in \mathrm{R}^2$, in order, while avoiding the defender which tries to capture the attacker before it accomplishes its objective. Design a controller, $u(t)$, for the attacker so that the attacker's path, $x(t) \in P \subseteq \mathrm{R}^2$, satisfies its desired objective.

3.1.5.5.2 Specification of a Multi-target Reach-avoid Scenario Linear Temporal Logic: The requirements and high-level constraints of the vehicles can be captured by Linear temporal logic (LTL) formulas [15]. An LTL formula, φ, is constructed over a finite set of atomic propositions, Σ, using the standard boolean operators (negation, $\neg$, disjunction, $\vee$), and the temporal operators (next $\bigcirc$, until U).

$$\varphi ::= p \mid \neg\varphi \mid \varphi \vee \varphi \mid \bigcirc\varphi \mid \varphi \,\mathrm{U}\, \varphi \tag{27}$$

Other Boolean operators (such as conjunction, $\wedge$, and implication, $\rightarrow$) and other temporal operators (such as eventually, $\Diamond$, and always, $\Box$) can be constructed based on the aforementioned list of operators in (27).

Consider Σ as the set of all propositions and $\sigma = \sigma_0, \sigma_1, \ldots$ as a sequence of truth assignments to propositions in Σ, where σ_i is the set of propositions at position i. We say $\sigma, i \vDash \varphi$ if φ is true at position $i \geq 0$ of σ.

A special class of LTL formulas is General Reactivity(1) (or simply GR(1)), which provides an appropriate format to describe the specifications in a dynamic environment, where there are interactions between the system and its environment [16,17]. A GR(1) formula can be described as:

$$\varphi = \varphi_e \rightarrow \varphi_s \tag{28}$$

where φ_e contains all the assumptions about the environment, and φ_s represents the assumptions on the system and its desired behavior. Formulas φ_e and φ_s are the conjunction of some sub-formulas in all three forms of B, $\Box B$ and $\Box \Diamond B$, where B could be a Boolean or temporal formula.

Assumptions and Requirements of the Vehicles in a MTDRA Scenario: Considering the defender vehicle as part of the environment of the attacker, we use GR(1) over the proposition set $\Sigma = \mathrm{A} \cup \mathrm{D} \cup \mathrm{T} \cup \{Accomplished\}$ to describe the MTDRA problem in the form of:

$$\varphi = \varphi_d \rightarrow \varphi_a \tag{29}$$

where φ_d and φ_a capture the assumptions and requirements of the defender and the attacker, respectively. Formulas φ_d and φ_a can be represented as the conjunction of five subformulas:

$$\varphi_d = \varphi^d_{init} \wedge \varphi^d_{sing} \wedge \varphi^d_{term} \wedge \varphi^d_{rul} \wedge \varphi^d_{obj} \tag{30}$$

$$\varphi_a = \varphi^a_{init} \wedge \varphi^a_{sing} \wedge \varphi^a_{term} \wedge \varphi^a_{rul} \wedge \varphi^a_{obj} \wedge \varphi^a_{opt} \tag{31}$$

where for $r \in \{a, d\}$ we have,

- φ^r_{init} describes the initial value of all propositions in $A \cup D \cup T \cup \{Accomplished\}$,
- φ^r_{sing} describes the *singularity requirement*, which requires each vehicle to be only in one of the partitions,
- φ^r_{term} describes the *termination condition*, which requires that no change occurs after the game is over (one of the vehicles reaches its objective),
- φ^r_{rul} describes the *transition rules*,
- φ^r_{obj} describes the *objectives* of the vehicles,
- φ^a_{opt} describes the *optimal discrete transition rules* for the attacker.

The following example explains how to capture these formulas for the vehicles in a multi-target dynamic reach-avoid scenario described in Problem 1.

Example 1. *We consider the environment that is partitioned into* 68 *disjoint cells with targets* T_1, T_2, *and* T_3 *located in regions* P_{23}, P_{35} *and* P_{57}, *respectively. Assume that the initial positions of the defender and the attacker are* P_{41} *and* P_{11}, *respectively. This configuration has been shown in Fig. 77.*

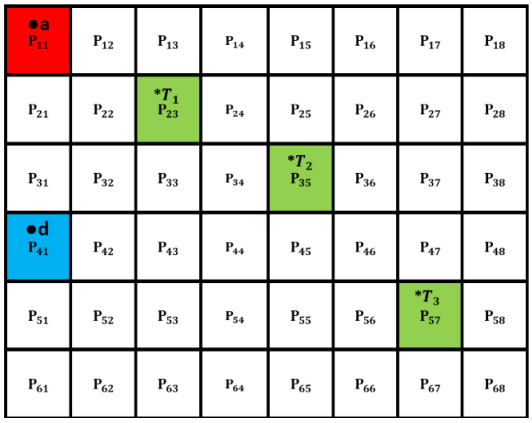


Figure 77: Configuration of an example of multi-target dynamic reach-avoid scenario. The targets are in regions P_{23}, P_{35} and P_{57}, and the attacker and the defender are initially in regions P_{11} and P_{41}, respectively.

The defender is initially in partition P_{41}. *Therefore* φ^d_{init} *will be:*

$$\varphi^d_{init} = d_{41} \wedge (\neg d_{11} \wedge \cdots \neg d_{35} \wedge \neg d_{42} \wedge \neg d_{68}) \tag{32}$$

The attacker is initially in P_{11}, and the first target to be visited is T_1. Therefore φ^a_{init} will be:

$$\varphi^a_{init} = a_{11} \wedge (\neg a_{12} \wedge \cdots \wedge \neg a_{68}) \wedge t_1 \wedge \neg t_2 \wedge \neg t_3 \wedge \neg Accomplished \quad (33)$$

According to the singularity requirement, each vehicle can physically be in only one partition of P at a time, and hence, only one of the vehicles vector propositions can be True. The singularity requirement for the defender can be captured by the following temporal formula:

$$\begin{aligned}\varphi^d_{sing} = \Box\ [\ & d_{11} \wedge (\neg d_{12} \wedge \neg d_{13} \wedge \cdots \wedge \neg d_{68})) \\ & \vee (d_{12} \wedge (\neg d_{11} \wedge \neg d_{13} \wedge \cdots \wedge \neg d_{68})) \\ & \vdots \\ & \vee (d_{67} \wedge (\neg d_{11} \wedge \neg d_{12} \wedge \cdots \wedge \neg d_{66} \wedge \neg d_{68})) \\ & \vee (d_{68} \wedge (\neg d_{11} \wedge \neg d_{12} \wedge \cdots \wedge \neg d_{67}))]\end{aligned} \quad (34)$$

Similarly, for the attacker, we have:

$$\begin{aligned}\varphi^a_{sing} = \Box\ [\ & a_{11} \wedge (\neg a_{12} \wedge \neg a_{13} \wedge \cdots \wedge \neg a_{68})) \\ & \vee (a_{12} \wedge (\neg a_{11} \wedge \neg a_{13} \wedge \cdots \wedge \neg a_{68})) \\ & \vdots \\ & \vee (a_{67} \wedge (\neg a_{11} \wedge \neg a_{12} \wedge \cdots \wedge \neg a_{66} \wedge \neg a_{68})) \\ & \vee (a_{68} \wedge (\neg a_{11} \wedge \neg a_{12} \wedge \cdots \wedge \neg a_{67}))]\end{aligned} \quad (35)$$

Once either of the vehicles achieve its goal, the game is over and both vehicles stay at the place they are and do not make any new decision. This occurs if the attacker visits all the targets, which means Accomplished is true, or if the defender captures the attacker, which happens when both the attacker and the defender are in the same region, i.e. $(a_{ij} \wedge d_{ij})$ is true for some $i \in \{1, \cdots m\}$ and $j \in \{1, \cdots n\}$. The termination requirement for the defender, therefore, can be described as:

$$\begin{aligned}\varphi^d_{term} = \Box[((& Accomplished \vee a_{11}) \wedge d_{11} \rightarrow \bigcirc d_{11}) \\ & \wedge (\ (Accomplished \vee a_{12}) \wedge d_{12} \rightarrow \bigcirc d_{12}) \\ & \vdots \\ & Accomplished \vee a_{67}) \wedge d_{67} \rightarrow \bigcirc d_{67}) \\ & \wedge ((Accomplished \vee a_{68}) \wedge d_{68} \rightarrow \bigcirc d_{68})]\end{aligned} \quad (36)$$

and the termination requirement for the attacker is:

$$\begin{aligned}\varphi^a_{term} = \Box[((& Accomplished \vee (d_{11} \wedge a_{11})) \rightarrow \bigcirc a_{11}) \\ & \wedge (\ (Accomplished \vee (d_{12} \wedge a_{12})) \rightarrow \bigcirc a_{12}) \\ & \vdots \\ & Accomplished \vee (d_{67} \wedge a_{67})) \rightarrow \bigcirc a_{67}) \\ & \wedge ((Accomplished \vee (d_{68} \wedge a_{68})) \rightarrow \bigcirc a_{68})]\end{aligned} \quad (37)$$

We assume that the vehicles can transit to the adjacent regions only through the edges not the vertices of the cells.

So, for the transition rules of the defender we have:

$$\begin{aligned}\varphi^{d}_{rul} = \square[&(d_{11} \rightarrow (\bigcirc d_{12} \vee \bigcirc d_{21})) \\ &\wedge (d_{12} \rightarrow (\bigcirc d_{11} \vee \bigcirc d_{22} \vee \bigcirc d_{13})) \\ &\wedge (\ d_{13} \rightarrow (\bigcirc d_{12} \vee \bigcirc d_{14} \vee \bigcirc d_{23})) \\ &\quad \vdots \\ &\wedge (d_{67} \rightarrow (\bigcirc d_{66} \vee \bigcirc d_{57} \vee \bigcirc d_{68})) \\ &\wedge (d_{68} \rightarrow (\bigcirc d_{67} \vee \bigcirc d_{58}))]\end{aligned} \tag{38}$$

Similar rules hold for the attacker. In addition, we assume that initially the attacker aims at visiting the target t_1. Once it reaches t_1, then it targets for t_2, then t_3, and finally, when it visits t_3, the proposition Accomplished becomes true. Considering all these requirements for the attacker, φ^{a}_{rul} will be:

$$\begin{aligned}\varphi^{a}_{rul} = \square[&(a_{11} \rightarrow (\bigcirc a_{12} \vee \bigcirc a_{21})) \\ &\wedge (a_{12} \rightarrow (\bigcirc a_{11} \vee \bigcirc a_{22} \vee \bigcirc a_{13})) \\ &\wedge (\ a_{13} \rightarrow (\bigcirc a_{12} \vee \bigcirc a_{14} \vee \bigcirc a_{23})) \\ &\quad \vdots \\ &\wedge (a_{68} \rightarrow (\bigcirc a_{67} \vee \bigcirc a_{58}))] \\ \wedge\ \square[&(d_{23} \wedge t_1 \wedge \neg t_2 \wedge \neg t_3) \rightarrow \bigcirc t_2) \\ &\quad a_{35} \wedge t_1 \wedge t_2 \wedge \neg t_3) \rightarrow \bigcirc t_3)] \\ &\wedge ((a_{57} \wedge t_1 \wedge t_2 \wedge t_3) \rightarrow \bigcirc Accomplished)]\end{aligned} \tag{39}$$

To address problem 1, the objective of the attacker is to capture the targets in a defined order. Using LTL, this complex mission can be easily expressed as:

$$\varphi^{a}_{obj} = \square \diamond (a_{23} \wedge (\square \diamond (a_{35} \wedge \square \diamond a_{57}))) \tag{40}$$

The opponent (defending) vehicle, however, is not under our control, and hence, its objective can be trivially written as $\varphi^{d}_{obj} = \square \diamond True$.

3.1.5.5.3 Optimal Discrete Decision Making for the Attacker The formula φ^{a}_{rul} in (31) describes the feasible transitions of the attacker which are based on this assumption that the players should transit to the next region through the edges of their current region. φ^{a}_{rul} then, is conjuncted with optimal transition rules, φ^{a}_{opt}. To find these optimal strategies, we propose to utilize a finite two-player zero-sum game in a matrix form to make a decision at each step for the attacker, according to the current status of the vehicles. For this purpose, we define the game's objective function as:

$$L(x'_a, x'_d) = \begin{cases} \infty, \text{ if } x'_a = x'_d \\ 0, \text{ if } x'_a = x_t, x'_d \neq x_t \\ \alpha \| x'_a - x_t \| + \beta / \| x'_a - x'_d \|, \text{otherwise} \end{cases} \tag{41}$$

where, x'_a x'_d are possible next regions of the attacker and the defender, respectively; x_t is the position of the next target; $\|.\|$ is norm 2, and α and β are tuning coefficients factors.

In this game, the vehicles share the same objective function (41). However, the attacker wants to minimize this objective function, while the defender wants to maximize it. By using this game configuration, the attacker will try to make the best decision by minimizing the cost function in (41), which requires the attacker to avoid the defender by maximizing the distance between the two vehicles, $\|x'_a - x'_d\|$, and to reach the target by reducing its distance from the target, $\|x'_a - x_t\|$. If the attacker and the defender are in the same region ($x'_a = x'_d$) (the attacker is captured by the defender), or the attacker reaches the target ($x'_a = x_t$), the game is over. As the first player, the attacker tries to independently and conservatively minimize its loss to make an optimal decision. To illustrate this procedure, consider the following example:

Example 2. *In Example 1, consider an arbitrary step of the game in which the defender and the attacker are at P_{61} and P_{11}, respectively, and the attacker aims to visit target T_1 located in P_{23} (See Fig. 77). Let $\alpha = 1$ and $\beta = 5$. Requiring to stay within P, and assuming that the vehicles can only go to their immediate vertical or horizontal neighbors, both vehicles have two options: the defender can go to either P_{62} or P_{51}, and the attacker can go to P_{12} or P_{21}. Using the objective function in (41), this game is represented in the following matrix form:*

		attacker	
		P_{12}	P_{21}
defender	P_{52}	2.25	2.58
	P_{41}	2.58	3.5

in which the optimal solution for the attacker, as the first player, will be transiting to the region P_{12} which minimizes the maximum loss as:

$$\min \max \{\{2.25, 2.58\}, \{2.58, 3.5\}\} = 2.58$$

Therefore, when the target is in P_{23}, and the current position of the attacker and the defender are P_{11} and P_{61}, the optimal decision for the attacker is to choose P_{12} as its next destination. The temporal formula which describes this optimal strategy will be:

$$\Box[t_1 \wedge (a_{11} \wedge d_{61})) \rightarrow \bigcirc a_{12}] \tag{42}$$

In general, the complete formula for the attacking agent will be:

$$\varphi^a_{opt} = \bigwedge_{i,j,l,k,s} \Box[t_s \wedge (a_{ij} \wedge d_{lk})) \rightarrow \bigvee_{(f,g)\in Q_{ij}} \bigcirc a_{fg}] \tag{43}$$

$$[(i,j) \in M, (l,k) \in M - (i,j), s \in \{1, \cdots k\}]$$

where t_s is the current target to be reached; s is the index of the current target; Q_{ij} contains the set of index of all desired regions in the neighborhood of a_{ij} (which are actually the solutions of the finite zero-sum game with the objective function in 41), and $M = \{(i,j) | i \in \{1, \ldots n\}, j \in \{1, \ldots n\}\}$ is the set of all indices for the regions in P.

3.1.5.5.4 Hybrid Controller Synthesis We provide the solution of Problem 1 in two steps. First, we design a discrete controller to derive discrete strategies satisfying the specification φ in (29). Then, we design a hybrid controller to generate continuous winning paths satisfying this specification.

Discrete Controller Design: Here, we synthesize the discrete controller as a finite state machine which satisfies the GR(1) formulas (if they are realizable). For reactive formula of type GR(1), [16] introduced an algorithm which has two steps. It first checks the realizability of the specification by evaluating a fixed-point equation, and then for the realizable formula, the algorithm synthesizes the winning strategy satisfying that formula. For detailed information about this process, we refer readers to [17].

Applying this algorithm to Problem 1 with the specification $\varphi = (\varphi_d \Rightarrow \varphi_s)$, the attacker always wins if φ is realizable, resulting is an automaton $G = (Q, q_0, \mathrm{D}, \delta, \mathrm{L}, \mathrm{X})$ where:

- $Q = \mathrm{N} \times \mathrm{A}$ is the set of discrete states,
- $q_0 = (1, a_0)$ is the initial state,
- D is the set of input propositions (defender's moves),
- $\delta : Q \times \mathrm{D} \rightarrow Q$ is the transition relation,
- $\mathrm{L} : Q \rightarrow \mathrm{A}$ is the labeling function,
- X is the set of final states.

If the automaton G is at state q_k, the current position of the defender $d_{k+1} \in \mathrm{D}$ is considered as input to G, and causes a transition from q_k to q_{k+1}, denoted by $\delta(q_k, d_{k+1}) = q_{k+1}$, and shown by $q_k \xrightarrow{d_{k+1}} q_{k+1}$. The newly generated output label at state q_{k+1} will be $a_{k+1} = \mathrm{L}(q_{k+1}) \in \mathrm{A}$. Based on the moves of the defender, a set of input labels will be received by G in the form of $d_0, d_1, \ldots$. The automaton G, then, executes a run in the form of $r = q_0, q_1, \ldots$, which is a sequence of states starting from the initial state q_0, where $q_0 \xrightarrow{d_0} q_1 \xrightarrow{d_1} q_2 \ldots$. Correspondingly, a sequence of labels $L = \mathrm{L}(q_0), \mathrm{L}(q_1), \mathrm{L}(q_2), \ldots = a_0, a_1, a_2, \ldots$ will be generated as the discrete path for the defender. Since the formula φ_d has considered all possible transitions (decisions) of the defender, all admissible sequences of decisions of the defender can be reacted by a sequence of actions of the attacker, leading the attacker to win.

Hybrid Controller Design:

In order to convert the generated discrete path to smooth continuous signals driving the attacker over the partitioned space, we use the method we proposed in [18] to construct a hybrid controller. Due to the bisimulation relation between the original system with a multi-affine dynamics and its abstract model, shown in [18], it can be guaranteed that the generated continuous signals preserve the properties of the discrete path.

Results: For Example 1, the automaton for the attacker is synthesizable and it is constructed using the described method as shown in Fig. 78. In this figure, the nodes represent the activated (true) attacker sensors, which means the position of the attacker. The defender sensors are not shown on the edges of the automaton graph.

A hybrid controller is designed where the simulation results are shown for two different actions of the defender in Fig. 78. Based on the derived automaton G, the control strategy is computed for the initial position of region P_{11} for the attacker, and region P_{41} for the defender.

In Fig. 78, we show a behavior of the defender as a red curve, which is interpreted for the attacker as the discrete input sequence d_{41}, d_{31}, d_{21}, d_{22}, d_{32}, d_{33}, d_{34}, d_{44}, d_{45}, d_{46}, d_{47}. Observing these discrete decisions of the defender at each position, the attacker makes an optimal deci- sion, about its next transition, according to the automaton G, which generates the discrete path a_{11}, a_{12}, a_{13}, a_{23}, a_{24}, a_{25}, a_{35}, a_{36}, a_{37}, a_{47}, a_{57}.

This discrete path is translated to a continuous signal, shown in blue, driving the attacker within P. As it can be seen, the proposed algorithm is capable of creating a winning strategy, so that the attacker can visit all three targets before being captured by the defender and win the game. Figure 78 shows the result of the same procedure for a different behavior of the defender d_{41}, d_{31}, d_{21}, d_{22}, d_{23}, d_{24}, d_{25}, d_{35}, d_{36}, d_{46}, d_{47}, which is reacted by the attacker's decision sequence a_{11}, a_{12}, a_{13}, a_{23}, a_{24}, a_{25}, a_{35}, a_{45}, a_{46}, a_{56}, a_{57}, and again confirms victory for the attacker.

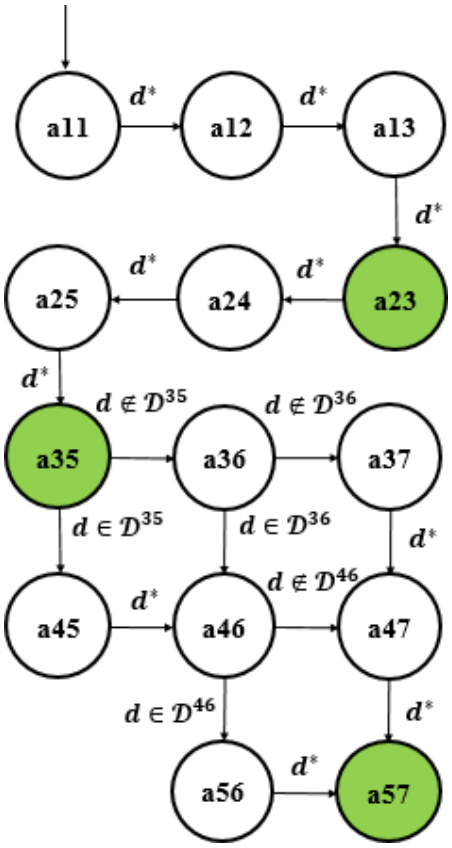


Figure 78: Some parts of the Synthesized automaton G for the attacker in Example 1. The nodes show the true attacker sensor, which means the position of the attacker. The edges are defender sensors, where 35 , 3 , 46 {d25, d26, d35, d36, d37, d38, d45, d47}, and d represents all possible defender's position information

.

3.1.5.5.5 CONCLUSION In this paper, we developed a novel effective symbolic framework for motion path planning of an autonomous attacking vehicle involved in a multi-target dynamic reach-avoid (MTDRA) scenario. This problem was formulated within reactive synthesis framework

GR(1) to describe all the assumptions and requirements of the vehicles as well as their interactions and conflicting objectives. Illustrative examples were provided to describe the implementation of the proposed approach.

3.1.6 Project Progress for Task T1-6 (Cloud-Based Control of LSASV)

3.1.6.1 Period of Performance under

Task T1-6 Start Date: Q3 2015

Conclusion Date: Q2 2022

Faculty lead: Patrick Benavidez

3.1.6.2 General Description of Task T1-6

Cloud computing infrastructure (compute, storage and network) in cloud data centers will be the platform for hosting the models and algorithms researched in this Thrust. The computing resources located in cloud data centers will be used to keep a virtual copy of each vehicle in the LSASV. Storage resources will be used to provide the optimal configuration for Data Analytic research ac- tivities (vSLAM, world-maps, etc.). Network resources will be used to create team boundaries for the LSASV swarm performing cooperative missions (shared information). The benefit of using a virtual vehicle will be to offload processing tasks that otherwise would overwhelm the vehicle's local processing capabilities. Since the computational power of remote vehicles is constrained by avail- able battery and processing power and physical characteristics of the vehicle, complex calculations can more optimally be made using the parallel computational facilities offered in the cloud. This approach allows for greater flexibility, such as including multiply simulated (or virtual) vehicles into a swarm of physical vehicles. Large-scale systems analysis can then be performed using as many vehicles as required to verify the correct functionality of the developed models and algorithms. Hu- man participants can also directly use the cloud to interact with the swarm of vehicles and override autonomous behavior as necessary. One of the necessary research challenges of using the cloud as part of the control loop for autonomous vehicles is in assuring adequate bandwidth and availabil- ity of a network connection between the LSASV and the cloud. We anticipate that some amount of local processing on each vehicle will be necessary to mitigate mission failures due to increased latency or intermittent outages in the network. After all, this is a key role and advantage of au- tonomy. Cloud-based robotic control has recently been researched and successfully used to process image data captured by robots. The image point cloud data can then be uploaded and processed in remote cloud data-centers. The result of the image processing can be used to update a common world-map of individual robots in the swarm.

3.1.6.3 Progress Against Planned Objectives in Task T1-6

The object of this task was accomplished. Serveral applications of cloud computing were explored, including:

- Implement a peer-based and cloned-based model for testing networked control systems.
- Control a swarm of unmanned boats to rescue passengers of a sinking ship floating in water.

- Gather data from a mobile sensor network to model an unknown signal and optimize sensor placement.
- Cluster autonomous vehicles on a road network.
- Place roadside units along a roadway to optimize the connection of autonomous vehicles to the cloud.

Alongside the expolation of applications for cloud computing with respect to LSASV, the following algorithms and architectures were also developed:

- Implement a cloud-edge architecture for the cloud layer of a networked system.
- Implement two types of peer-based model called Agent-VM and Task-VM.
- Develop a fuzzy algorithm to automatically place applications in the cloud to optimize performance.
- Develop algorithm to cluster vehicles on a roadway to enable platooning.
- Develop architecture for utilizing cloud and fog computing with a robotic swarm to process live data.

3.1.6.4 Technical Accomplishments in Task T1-6

3.1.6.4.1 Small-Scale Testbed A comprehensive survey was conducted to explore the existing techniques for implementing robotics algorithms in the cloud in parallel of learning software and programming languages. Significant benefit of using cloud will be to offload processing tasks that otherwise would devastate the vehicle's local processing capabilities (consequently more battery life of the vehicle) and implementation of multifaceted calculations in parallel computational can speed up the more complex algorithms. In addition, cloud allows for greater flexibility, such as running multiple algorithms in parallel. The process separated into two major steps: first running the one robot in the cloud, second running more than one robot in the cloud.

A java application that would map the movements of a robot in a GUI was developed. In order to achieve this, Robot Operating System (ROS) was used; ROS is a software infrastructure that relies on the passing of messages through the publisher/subscriber relationship. Originally ROS was intended to execute code written in C++ or Python, but was later extended to what is known as ROSJAVA. ROSJAVA allows java applications to be written in ROS with full access to messages being passed regardless of the coding language. The program retrieves specific inertia measurement unit (IMU) data (velocity in x, velocity in y, yaw) from a Parrot Bebop and performs calculations on them to determine the robots change in position. This data is published to the odometry topic */odom*. This topic has a message format specific to odometry type messages and must be referenced in a certain way. We will see this formatting in the Java program when it accesses those odometry messages. Overall, this node's task is to continuously calculate and publish the robots current x and y position. For mapping, a java program subscribes to the */odom* topic and uses this data to plot a circle according to the received coordinates. This is done repeatedly and the outcome is a line that represents the path the robot has taken. In order to control the drone, a keyboard controller was written in python. This program constantly reads input from the keyboard and when a key is

pressed, it enters a lookup table that sets velocity/orientation commands accordingly. Along with basic control of the robot, the python program allows software pan and tilt of the image stream coming from the drone's wide angle camera.

A ROS Multimaster network was set up that allows each Kobuki robot to have its own master node if it so pleases. This allows unique topics to be published for each robot whereas without the Multimaster network, each Kobuki would be publishing data under the same topic name (since Kobuki software is the same for all the robots) causing a clash in data. So far a network of two Kobukis and the Bebop drone can all see each others data on the ROS network, and use this data as each robot sees fit. Scripts were written to record and sensor and image data into the Hadoop Distributed Filesystem (HDFS). The sensor data is comprised of inertial measurements and time- stamps which is written into a csv file for storage. The images utilize the HIPI software for storing into the HDFS. Each script saves their respective data locally on the machine running it, and then once a size threshold has been reached, currently 100 lines of text and 100 images, the scripts then send the data to the cloud for long term storage and calculations. The amount of data sent can be changed to optimize Hadoop's parallel processing. Python scripts were also written to store the data from ROS Nodes into CSV files. These python scripts would take a keyword as an input and automatically find all ROS Topics with that given keyword. The program would then automatically subscribe to all of those topics and then write the data to comma separated values file (CSV). The CSV file would be named based on the ROS topic name, and the first row of the CSV file would names of data that is associated with that topic such as time-stamp, linear velocity x, etc. Then the data at each time-stamp would be written to the CSV file. Along with that, the CSV files were store in an organized file directory system. Inside of the root data folder is a group of folders of all the keywords used. Inside each keyword folder is a folder for each topic name without any name-spaces. Lastly inside each topic name folder is a CSV file named based on the full topic name with the associated name-space. This file system allows the data to stay organized, while making it easier to have a script store each CSV file into Hadoop.

3.1.6.4.2 Peer-based and Clone-based Models The backbones of the system to develop a peer-based and clone-based cloud model were developed. Two different types of systems were completed to accomplish this task; 1) Virtual Machine (VM) Based System and 2) Container-Based System. The VM based System uses KVM and Virsh to automatically set up VMs on the cloud network. An image was created to contain the libraries and packages that are needed to interface with ROS, and can be easily extended to contain anything else that would be needed by the system. This image can contain a program to run a specific service (Peer Based Model) or can contain a copy of the entire agent (Clone Based Model). The Container-Based System uses Docker in Swarm Mode. Docker in Swarm mode does the same thing as the VM based System, but uses containers instead of VMs. The testing and verification for these models was developed using the RTABMAP program from Task T1-4, which was modified to operate on the cloud network. This was used to decouple the processes that need to run on-board the autonomous system from the processes that need to be run on the cloud network. This decoupled program allows the autonomous system to publish all of the necessary topics such as odometry and image data, and allows the cloud network to subscribe to them for use with RTABMAP. By moving the map creation process to the cloud network, the cloud will be able to store many more images to create better maps for future use. The testing environment consists of autonomous ground agents and autonomous aerial vehicles working together to rescue people after a cruise ship disaster. The process of rescuing the people requires

many different algorithms to run, such as clustering and auctioning algorithms. The algorithms vary in both computational necessity and the amount of bandwidth needed to pass the information to the next process. All of these algorithms and processes have been dockerized into containers and the containers can easily be spun up on different machines. The containerized environment to test the various cloud models can be seen in Figure 79. The data collection processes have also been implemented into each of the containers. This will allows the containers to collect the necessary data to analyze each of the cloud models. The data collected by the containers will then be stored into Hadoop for post-processing and analysis. This data includes processing time, size of data needed to be sent, and communication delay.

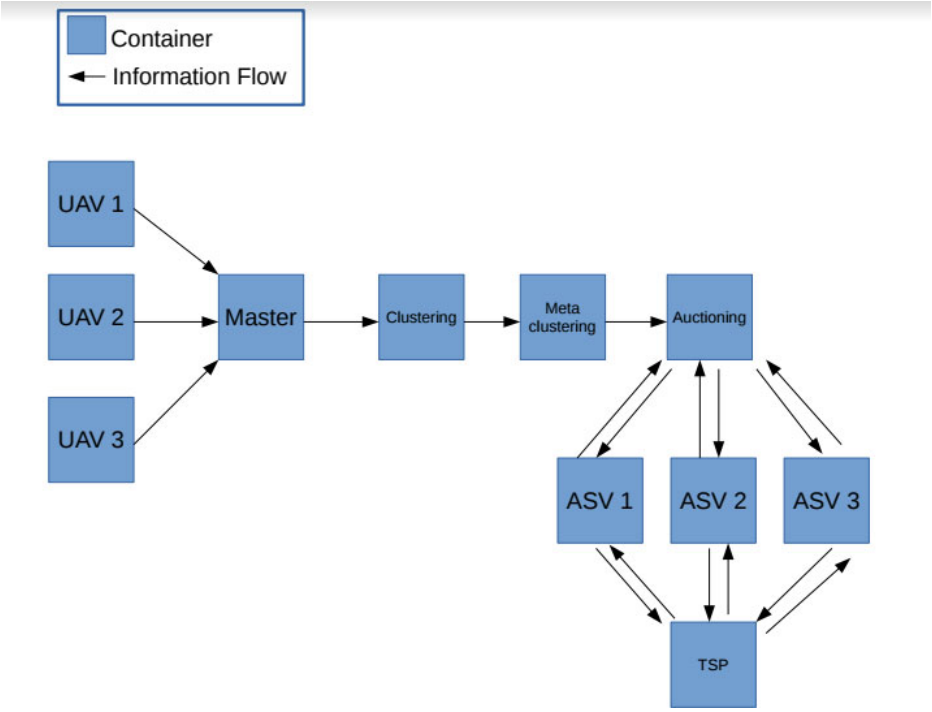


Figure 79: Containerized environment to test various cloud models.

A hardware experiment for testing data transfer rate was developed based on the previously discussed scenario involving a boat swarm for a cruise ship disaster. The drawback of the previously reported scenario is that it utilized VMs to represent the end and edge devices. This is not realistic because the networking latency between VMs will be small compared to the actual system. By using a better hardware setup, the delays will be able to be measured more accurately. End devices in the experiment are comprised of Raspberry Pi 3 Model B microcomputers running a Linux-based operating system. These end devices are used to interface to the control hardware of the robotic boat agents along with the control hardware for the UAVs. Additionally, these devices are used to implement the tasks which previously were simulated in software for obtaining timing data on planned experimental tasks. This use of physical hardware introduces realistic task durations to the experiment. Docker is used on the microcomputers to containerize the software components as it was implemented before in the previous software-based experiments. This hardware setup also

demonstrated the benefits of using Docker due to the easy deployment of containers on various devices by allowing us to move the Docker containers that were used on the VMs to the Raspberry Pis. These end devices communicate via IEEE 802.11n WiFi with an edge device. The edge device are capable of managing connections to the end devices. Edge devices were implemented with both portable laptop computers with software-based routing, such as Zeroshell, and a WiFi router with Linux installed on it for additional customization of capabilities, such as VyOS. This allowed us to have complete control of the edge device, and allow the edge device to be used for computation. Finally the edge device communicates to the cloud, which is hosted by Chameleon Cloud in Austin, Texas, via Ethernet (IEEE 802.3). RabbitMQ messaging queues are used to communicate with the cloud since ROS makes it difficult to communicate with nodes hosted on the cloud. RabbitMQ is an industry standard publisher/subscriber messaging queue for cloud applications. The RabbitMQ and MySQL servers were setup in the cloud. The RabbitMQ server is the host of all of the communication between nodes, and the MySQL server is used to store all of the data collected during experiments. Lastly, a Kubernetes cluster was created in the cloud. Kubernetes allows for the automatic management of all of the nodes in the swarm. Kubernetes automatically scales a node if the CPU or memory load becomes too high for a particular node, and it also will restart nodes that have crashed. This allows for the swarm to be very robust.

3.1.6.4.3 Cloud and Cloud-Edge Architectures The previously described system was first implemented in a cloud only architecture to verify the functionality of the components. Once the system is completely functional, we collected data on the runtime of each algorithm, the communication latencies, and the runtime for the entire system. This data acted as a baseline for a second test, where the system was implemented with the Cloud-Edge architecture. The Cloud-Edge Kubernetes System addresses some of the drawbacks of the Cloud Only Kubernetes System. Since not all of the algorithms require a high amount of computational power, it may not be necessary to perform the algorithm on the cloud. Instead some of the moderately computational expensive algorithms can be placed on the edge device. Since the edge device is significantly closer to the end devices, the latency will be much lower than performing the algorithm on the cloud. To setup the Cloud-Edge Kubernetes System, we installed Docker and RabbitMQ on the router, allowing for it to communicate with the system and perform calculations for the Cloud-Edge model. The algorithms in the Cloud-Edge system, seen in Figure 80, consists of a MAV search algorithm, people location combiner, two clustering algorithms, an auction algorithm, and a traveling salesman solver. The Cloud-Edge architecture was applied to the system through the clustering algorithm. The original clustering algorithm was split into two, one for the speed layer and one for the batch, and both fed into the metaclustering algorithm.

The results for the Cloud-Edge system were not as expected. The updated rate and runtimes of the Cloud and Cloud-Edge architectures were about the same. This is caused by the fact that the algorithms that were placed on the edge device were slowed down by the lack of processing power on the edge. This decrease in speed was about equal to the time saved due to the faster communication speeds, which resulted in the two architectures operating at about the same speeds. From these results, it was determined that the following need to be considered when choosing to use the original Cloud architecture versus the Cloud-Edge architecture: computational power of edge device, application's tolerance to latency, application specific hardware requirements, the storage requirement for the applications, intra-subnet communication, and communication load of the application.

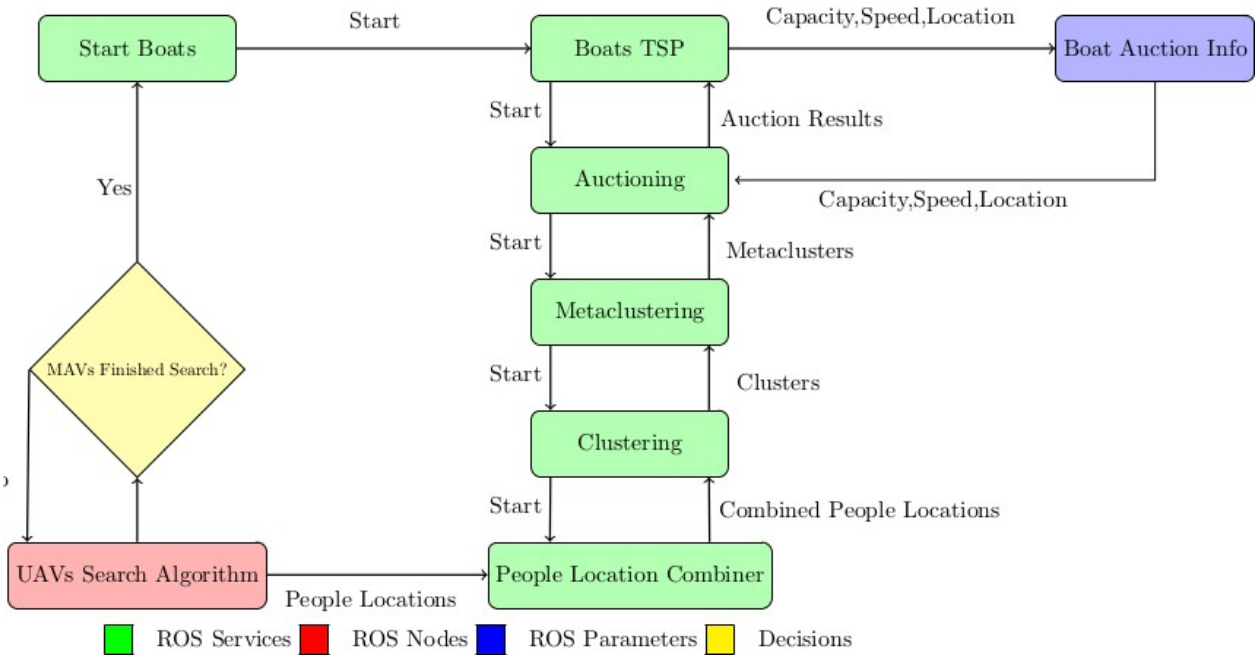


Figure 80: High level overview of cyber-physical system used.

The first thing that needs to be considered when choosing to use cloud or cloud-edge is the computational power of the edge device. Currently, the processing power of edge devices are increasing, but still are not nearly as powerful as small micro-computers. This severely limits the number of applications, and the computational need of an individual application that can be placed on the edge. As a result, if a large number of applications needs to be placed on the speed layer, or if a single computationally expensive application needs to be placed on the speed layer, the traditional cloud architecture may be more desirable. However if the speed layer consists of only a few lightweight applications, then using the cloud-edge architecture may be better. Another consideration that needs to be made when deciding which architecture to use is the end devices' application tolerance to latency. For example, if the end device's application is a control algorithm for a robot, then its tolerance to latency will be very low, because typical control algorithms need to operate at a minimum of 50Hz. This means that the total runtime of the control algorithm will need to be less than 20ms including communication time, making it difficult to place the application in the cloud. In a scenario like this, it may be very beneficial to use the cloud-edge architecture. Other considerations that must be made are the hardware and storage requirements needed for the application. Some applications, such as machine learning, operate much faster when a graphical processing unit (GPU) is present. Since majority of edge devices do not have GPUs, using the cloud-edge architecture is not always ideal. Furthermore, applications that require a large amount of storage, such as one that builds a map out of many images, may not run well on an edge device due to their limited storage capabilities. Many applications need information from other applications or sensors that are not connected to the same edge device. Therefore, that application would have to receive data from a different edge device than the application is connected to. If this application was hosted on

the edge device, then the application would still have to get the information for the other devices through the cloud. If the application was instead host on the cloud, the application could access the information directly from the cloud. This makes it ideal to use the traditional cloud architecture if the applications need information from sensors or applications that connect to many different edge devices. The final consideration when choosing which architecture to use is the communication load the application will have on the network. For example, if an application needs to send large images to an end device, then you may want that application to exist on the edge device, because sending images from the cloud to the router, and then to the end device may take a long time. However if the application is sending very small packets of data, then placing the application in the cloud will not be an issue, because it will only take a very small amount of time to send the small packets of data from the cloud to the end device.

3.1.6.4.4 Fuzzy Application Placement Controller Because of the multitude of net- work configurations, and consequent combinatorial explosion, the cloud-edge model has issues with optimization at scale. Randomness and non-negligible, exogenous shocks further complicate the analysis. More importantly, these are unknowable a priori, and typically can only be classified post hoc. Accordingly, a fuzzy logic application placement controller is developed, for it does not depend on a model.

This fuzzy logic application placement controller is designed to optimize the placement of containerized applications such that the total application loop delay is minimal. This includes the communication delay to send data from the sensors to the application, and the delay from the application to the actuator. The inputs to the controller include:

- **Cloud Speedup:** The speedup factor that would be attained by running the application on the cloud, excluding the loop delay, compared to running it on a target end device, used as a reference for the system, calculated as the product of the speedup attained due to hardware differences between the cloud and the reference end device, and the speedup attained due to the ability to parallelize sections of the algorithm
- **Edge Speedup:** The speedup factor that would be attained by running the application on the edge device, such as a router or switch
- **Magnitude of Runtime:** The amount of time it takes to run the algorithm on the reference end-device
- **Latency:** The base delay between the edge device and cloud, irrespective of message size
- **Requires Other's Info:** Boolean stating whether or not the sensor and actuator are con- nected to the same end device
- **Bandwidth:** The bitrate of that the network is capable of sending from the end device to the edge device
- **Throughput:** The bitrate the application actually sends through the network

After taking in all these factors, the fuzzy application placement controller decides on which node the application should be placed. A block diagram overview of this fuzzy logic controller can be seen below, in Figure 81.

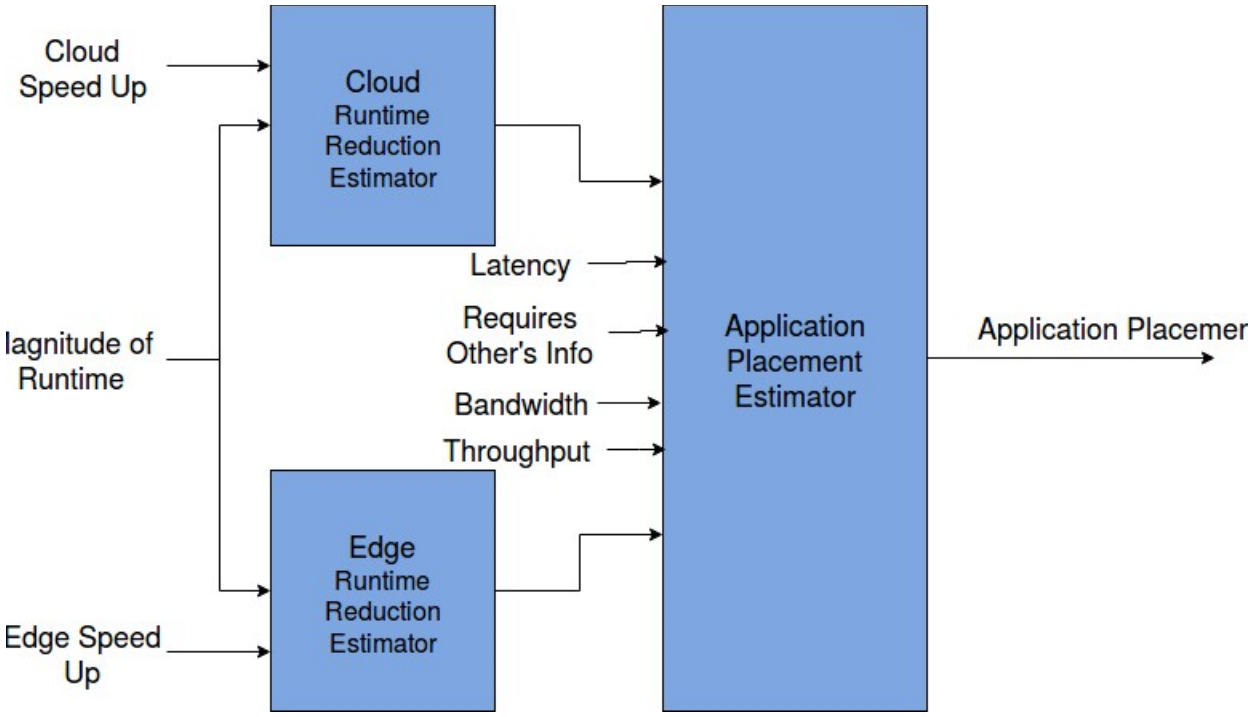


Figure 81: Overview of Fuzzy Application Placement Controller

Instead of comparing the decisions of the application placement controller to those of an expert, they were tested in Yet Another Fog Simulator (YAFS). YAFS is a highly configurable, Python-based simulator for cloud, edge and fog computing. Using conventions from graph theory, each computational device in the network is represented by a node, and each connection between two computational devices is represented by a link. The network is modelled as an undirected graph, and for the particular network under investigation, the two end devices are identical except for the sensors and actuators they contain. A simple network is shown below in Figure 82, where node 0 is a cloud node, node 1 is an edge node, and nodes 2 and 3 are end devices.

The network was then simulated for all possible placements of the application. The placement with the smallest total loop delay was selected as the analog to the expert's decision in evaluating the performance of this fuzzy logic-based application placement controller. From preliminary membership function settings, the placement controller successfully placed the application on the optimal node 64% of the time. Moreover, for 80% of the cases, the placement decision of the controller took no more than 20% extra time. For example, suppose there were 100 trials, and for each trial, the network was configured randomly but in such a way that the total loop delay for the optimal placement was 10 seconds. It would be the case that for about 80 trials, the placement decision would result in a runtime that was less than 12 seconds.

3.1.6.4.5 Improving the Cloud-Edge Architecture The results of the Cloud-Edge architecture showed that it did not improve the speed of the cruise ship disaster scenario. This was due to the delay lost by moving the speed layer to the edge being the same as the computation time gained from the slower edge device. While the speed was not improved in the experiment, other

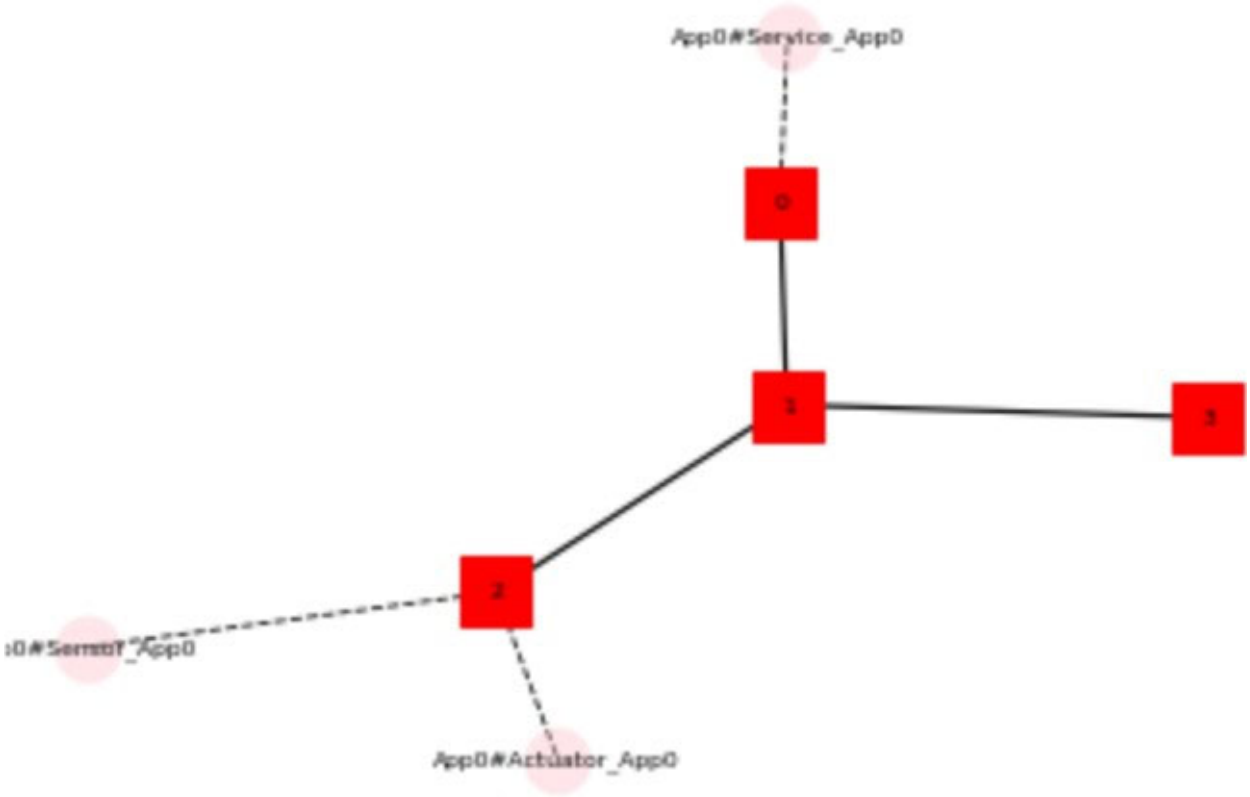


Figure 82: Graph of simulated network

benefits were found in the form of cost savings and more efficient use of the system's total resources.

We proposed that the speed layer of the Cloud-Edge architecture be split among the cloud, edge, and end devices of the system. Making this change allows the user to optimize the runtime of the system while maintaining the benefits of the original Cloud-Edge architecture. The placement of each part of the speed layer is made with the variables discussed previously. In order to test the effectiveness of this architecture, a new scenario was developed instead of the cruise ship disaster. This decision was made based on the results of the application placement controller and the inability to further split the clustering algorithm. The new scenario developed is the sensor placement optimization problem, which involves finding the optimal placement of mobile sensors to monitor a signal. The scenario assumes that the signal is unknown to the mobile sensors and must be modeled before optimal positions can be found. This scenario combines numerous fields including optimization, consensus control, and modeling, which make it ideal to test the updated Cloud-Edge architecture. Two other architectures were also implemented with this new scenario, a control architecture where all components are calculated onboard the mobile sensors and an alternative to the Cloud-Edge called the Kappa architecture. The Kappa architecture still splits an algorithm into two layers like the Cloud-Edge, but they are the exact same and are only provided different inputs as opposed to the two unique methods in the Cloud-Edge.

The scenario was simulated in Gazebo, which is a simulation environment complete with physics and ROS support. The models that have been used for the mobile sensors was the ROS Turtlebots,

which are similar in function to a Roomba. The tests were run using four separate devices; one for the end devices simulation, one for the end devices, one for the edge device, and one for the cloud. The simulation for the end devices was with Gazebo using four Turtlebot 2 models. The simulation was run on a desktop with a Intel Core i5-6600 running at 3.30 Ghz and 25 Gb of RAM. The end device was controlled by an ODroid XU4, which has a Exynos 5422 Cortex A15 running at 2Ghz and 2GB of RAM. The ODroid is the processor that was used to control the Turtlebot2, which is why it was chosen. The edge device was a LIVA X Mini PC with an Intel Celeron CPU running at 1.58 Ghz and 4 GB of RAM. Lastly, the cloud layer was an AWS EC2 t2.micro instance. All devices were running Ubuntu 16.04. The Robot Operating System (ROS) was used as a message broker between local devices, and RabbitMQ was used as a message broker between the edge device and the cloud.

The results are based on two sets of tests. The first analyzed the network connection between each device in the system in order to compare latency. The second asseses the time it takes to calculate a solution for the speed and batch layer for each architecture. This information is used to assess which architecture has the best performance.

Test 1: The first test used Ubuntu's ping command to record roundtrip time (rtt) statistics between different components of the test setup. The command sent 1000 pings at a rate of 10 Hz and a size of 80 bytes each. This is the same speed and size as the messages between the mobile sensors and the edge device, which is of comparable size to messages sent between the cloud and edge. Measurements were recorded between the simulation and the end device, the end device and the edge device, the simulation and the edge device, and the edge device and the cloud. The results are shown below in Table 8; The results show that the time to send a message to the cloud from

Table 8: The minimum, maximum, mean, and variance of the tts of 1000 pings between the end device, simulation, edge device, and cloud

	min (ms)	max (ms)	mean (ms)	variance (ms)
End - Edge	1.22	323	6.41	661.62
End - Sim	1.27	312	4.66	184.14
Sim - Edge	.2	.75	.40	.006
Edge - Cloud	39.1	155.0	40.14	46.69

the edge device is much higher then messages between the edge device and the end device. The 33.73 ms saved by not sending the speed layer solution is non-negligible, which will be shown in the following subsection. The table also shows the difference having a wired connection can make. Both the simulation and edge device were connected by Ethernet to the same local network and had an average tts 0f .4 ms, an order of magnitude smaller then a WiFi connection in the private network. Based on this fact, the Edge-Cloud tts is a best case scenario, additional time would be added if the edge device had to connect to the local network over WiFi like the end device.

The ping command also records the packet drops of each test, which are shown in Table 9. The

	End-Edge	End-Sim	Sim-Edge	Edge-cloud
packet drops (%)	7	10	0	0

Table 9: Percentage of packets dropped over 1000 pings between end device, simulation, edge device, and cloud

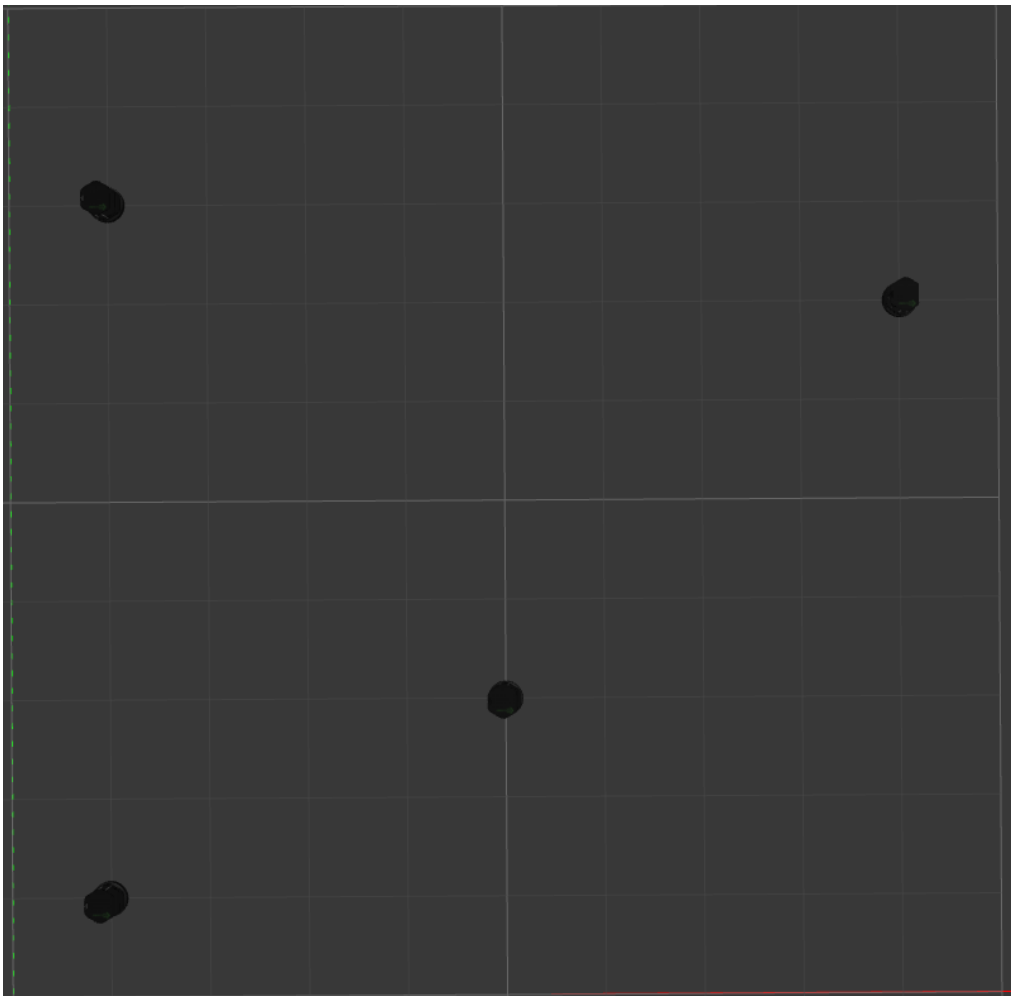


Figure 83: Gazebo simulation starting configuration, with 4 agents located at (5,3), (9,7), (1,8),

packets dropped by the end device are very large at around 8%. This may be caused by the WiFi adapter being used, which is also manufactured by Odroid. The other two connections have small enough packet drops to be functionally 0, which is expected for two systems that are physically connected. While these results indicate that using a different WiFi adapter may be advised, the one being used is provided with the Turtlebot2, and therefore a realistic expectation for the system.

Test II: The second test involved measuring the run time of different programs for each architecture. The run time was measured using Rospy's get_time function. The time it takes for the speed and batch layers to calculate the next desired position and update their model was recorded. Each architecture was run for some time after it reached equilibrium, then the data from each was truncated to the length of the shortest trial to ensure that the same amount of iterations was compared for each algorithm. Each architecture was run on the same simulation, which has a area of 10x10 meters and four agents at the coordinates (5,3), (9,7), (1,8), and (1,1). Figure 83 shows the simulation environment.

The runtimes of the desired position algorithms are plotted below in Figure 84. Looking at the plots, it is clear that most of them have a correlation between time (iterations) and the runtime. This can be explained by the runtime increasing as more data is collected but decreasing as each robot gets closer to equilibrium, triggering the stop condition sooner. Both the Lambda and Kappa architectures use the same batch desired position algorithm and on the same device, thus their results are very similar. The runtime of both decrease as the stopping criteria is met faster. For the speed desired position algorithm, the Control and Kappa architecture both use the same simulation as the batch layer, but instead of simulating each agent until it reaches equilibrium it calculates only the next position. Both architectures performed similarly, averaging out to around 3 seconds each. This speed is extraordinarily slow compared to the batch layer's equilibrium of around 30 ms due to the relative weakness of the Odroid and Liva computer. The Lambda architecture's speed desired position algorithm was able to calculate the position in about 10 ms. This can be attributed to the Odroid only having to calculate it's own desired position and that being its only task.

Because the other modeling algorithms have little or no trend over time, their summary statistics can be analysed instead in order to show trends in the runtimes. Figure 85 displays the runtimes of both the batch and speed layer modeling as boxplots. Boxplots were chosen to represent the data because it is resilient to varying lengths of recorded data, unlike similar methods such as histograms. The first of the two plots shows the runtime of the batch modeling algorithm used in the Lambda and Kappa architectures. The two architectures had very similar results, with a mean runtime of 1 ms and a similar variance. The Kappa architecture has a few more outliers then the Lambda architecture, but the amount is negligible. These results make sense, both architectures use the exact same algorithm for the batch modeling and both run on the same system, therefore they should have very similar results. The second plot displays the results of the speed modeling algorithm for the three architectures. Of the three, the Lambda architecture is clearly the fastest. This is a result of using best-fit plane, which was designed to be extremely quick to calculate. The Control and Kappa architecture on the other hand use the same modified RBF as their speed models. It can be seen that the Control architecture has a longer average runtime and a far greater variance compared to the Kappa architecture. This can be attributed to the weaker computational ability and higher load on the Control architecture, which is required to calculate both the modeling and desired position algorithms.

3.1.6.4.6 Implementing a Model Predictive Control Algorithm The key shortcom- ings of cloud-based direct control of LSASVs include the effects of time delay and packet loss. In response to these shortcomings, direct control can be managed by a model predictive control algo- rithm in order to generate interim control schedules for actuation before the next control message is received. For this implementation of MPC, the predictions were generated according to a dis- cretized, nonlinear model as given by 4th-order Runge Kutta approximation, and the prediction accuracy and runtimes were compared to those of an adaptive model predictive control algorithm, where the system is linearized about the current operating conditions at each time step. As shown in Figure 86, the propagation of the error from linearization causes considerable divergence from the system's simulated response to the same inputs. The test scenario was the UGV driving on a circular path for 1 second, at a control frequency of 10 Hz, and the predictions are shown where "RK4" is the 4th order Runge Kutta prediction, and "LDT" is the Linearized, Discretized predic- tion. Each was compared to a much higher resolution numerical solution labelled "groundtruth." Further, the predictions could be generated significantly quicker with the nonlinear model than with

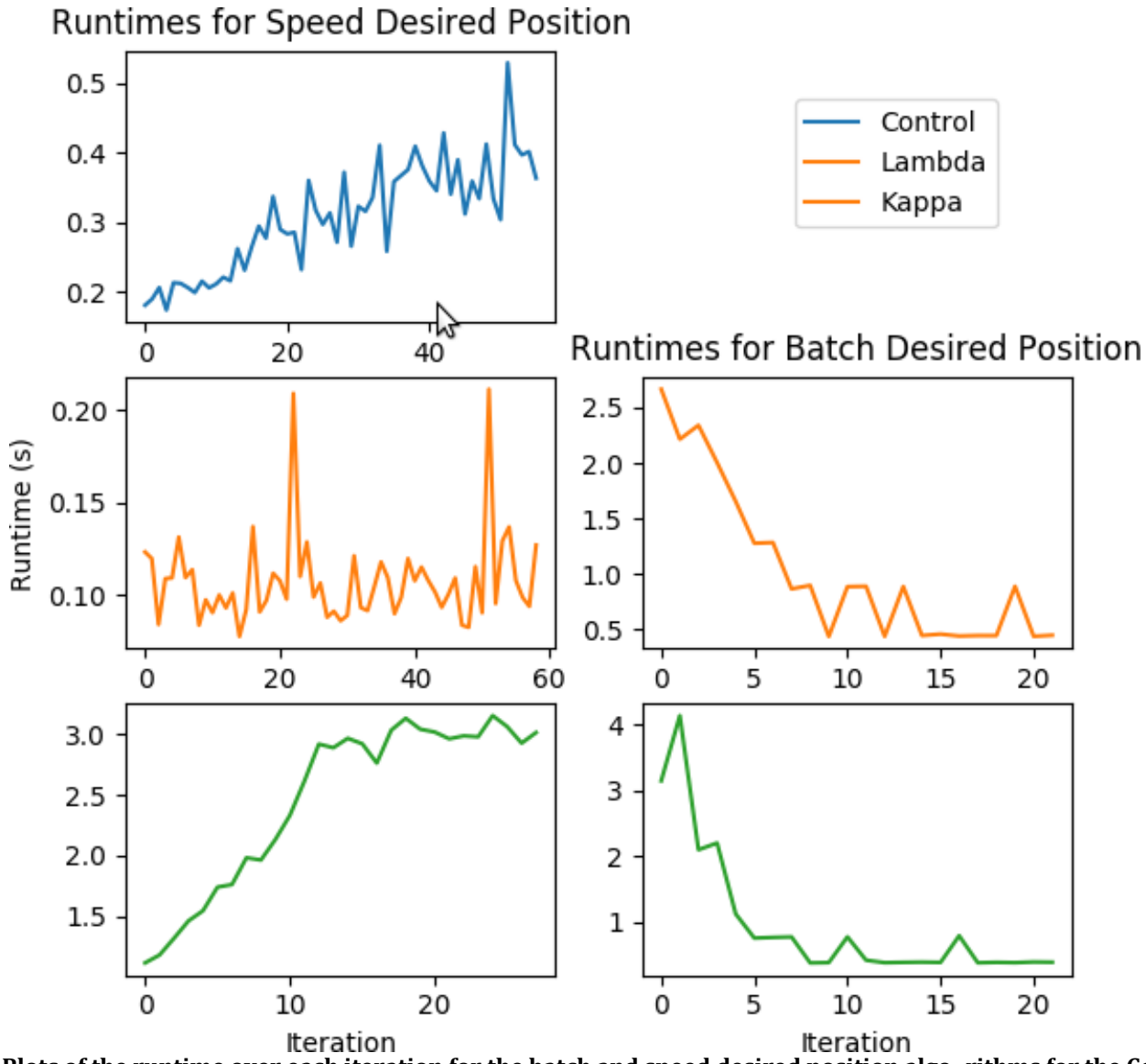


Figure 84: Plots of the runtime over each iteration for the batch and speed desired position algo- rithms for the Control, Lambda, and Kappa architectures

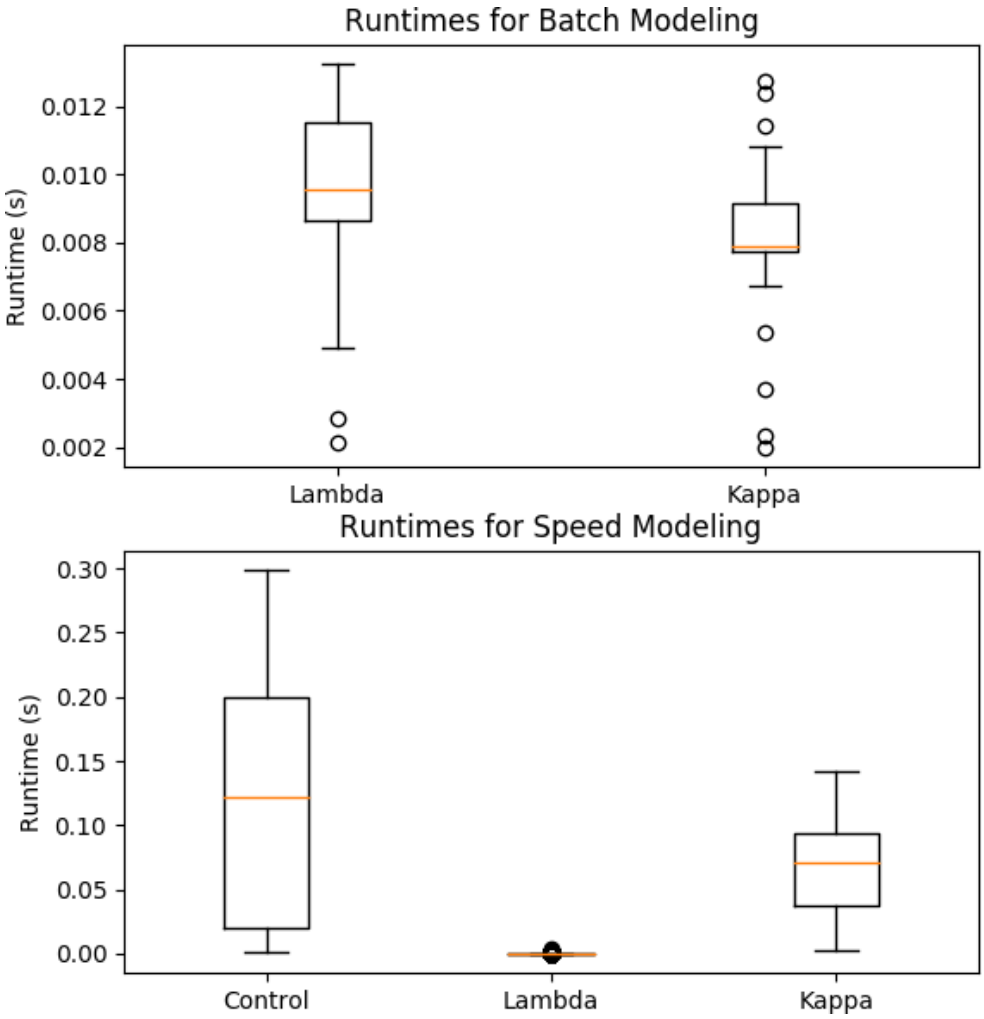


Figure 85: Boxplots from the runtimes of the batch and speed layer modeling for the Control, Lambda, and Kappa architectures

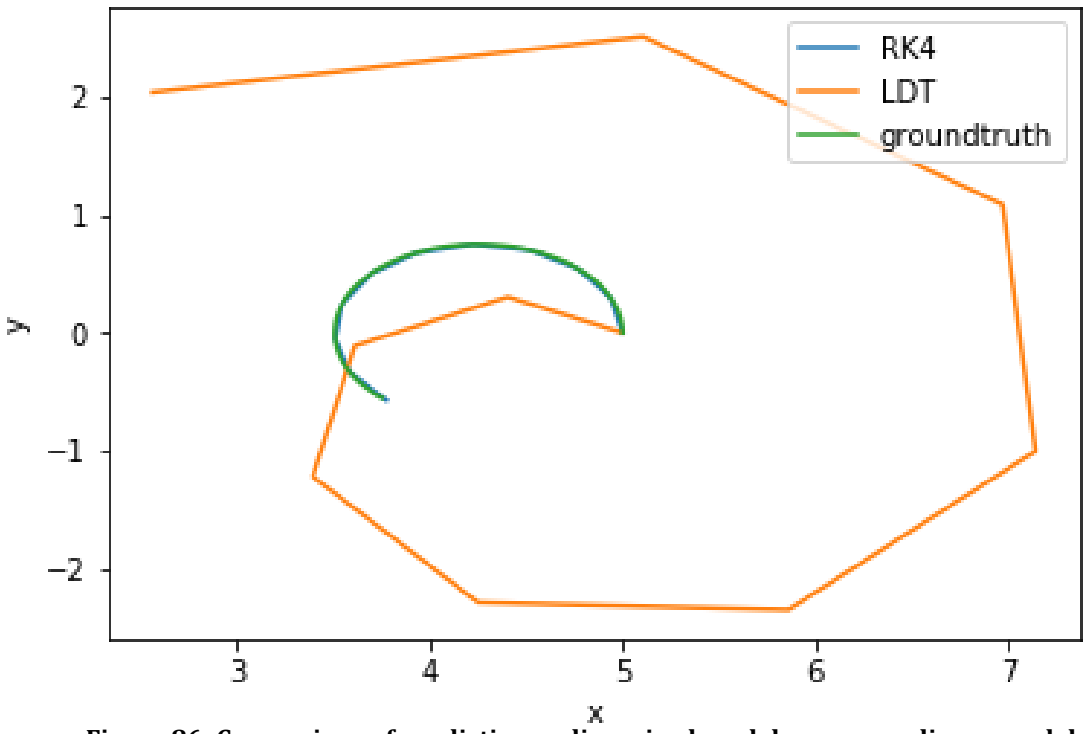


Figure 86: Comparison of prediction on linearized model versus nonlinear model

the linearized, discretized system, as shown in Table 10. Table 10 was generated by calculating the prediction horizon 1000 times, and the mean runtime and standard deviation of the runtimes are reported from the sample. Both the nonlinear prediction algorithm and the linearized prediction algorithm were compared in speed to a numerical solver (ODEint) of equivalent resolution. It is

Table 10: Comparing the runtime of different prediction algorithms

Method	$\bar{x}$[ms]	s_x[ms]
ODEint	1.896	0.322
RK4	1.880	0.286
LDT	2.615	0.386

important that these prediction runtimes are as small as possible because the optimization solver uses sequential least squares, and as such, calls upon the employed prediction algorithm multiple times.

Both prediction systems were then tested by subjecting it to up to 80% packet loss, (i.e. 80% of the time, the agent would not receive the new control schedule and would have to use the last received control schedule in the interim, until it received a new one). This condition was tested several times, with random initial poses and target poses within a 10m by 10m square centered on the origin. Performance was measured by the ability to drive the agent within 10cm of the target pose within 20 seconds. Beyond 30% packet loss, the MPC algorithm using the linearized model for predictions was unable to successfully drive to the target pose. The MPC algorithm using the nonlinear model

was able to successfully drive to the point in each trial. A representative trial is shown below in Figure 87. Several estimation methods were compared in order to reduce the computational load of

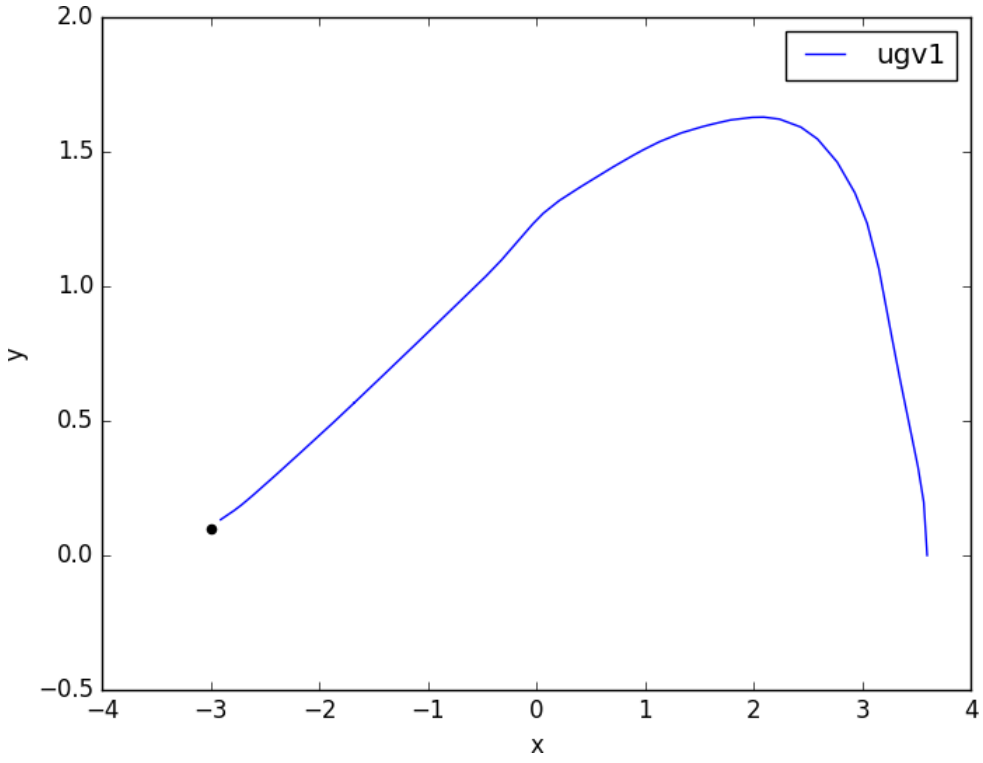


Figure 87: Drivepoint scenario for UGV subject to 80% Packet loss

the model predictive controller on the UGV. The most resource intensive component of the nonlinear model predictive controller is the optimizer, for it requires the approximation of solving a single differential equation many times over. The methods tested were:

- Sequential Least Squares Quadratic Programming (SLSQP)
- Constrained Optimization By Linear Approximation (COBYLA)
- Limited Memory Bounded BFGS (L-BFGS-B)
- Truncated Constrained Newton's Method (TNC)
- Interior Point Optimizer (IPOPT)

The results found the two most competitive paradigms to be SLSQP and L-BFGS-B, as shown by Table 11, where $\bar{x}_T$ is how much time it took for that algorithm to solve it on average, S_T is the standard deviation of the solution time, $\bar{x}_J$ is the average percent of the optimal solution for that scenario (i.e. 100 is optimal), and S_J is standard deviation of that metric. The metrics chosen were for concise representation of each's performance, though they do betray a prior assumption

Method:	$\bar{x}_T$ [s]	S_T [s]	$\bar{x}_{\bar{J}}$ [%]	$S_{\bar{J}}$ [%]
SLSQP	0.1358	0.1326	100.88	4.0658
COBYLA	0.2481	0.0386	107.35	21.940
L-BFGS-B	0.1006	0.0229	100.93	4.4646
TNC	0.2236	0.0959	100.67	1.5679
IPOPT	2.2999	11.145	103.16	23.185

Table 11: Comparison of estimation algorithm runtimes

that both variables come from gaussian distributions which is hardly the case. As such the notation refers to the empirical mean and empirical standard deviation, as opposed to estimates of the population distributions' true parameters. Experimentally, the IPOPT and COBYLA, had a much greater frequency of outliers in both time and suboptimality than would be suggested by exponential attenuation of likelihood thereof, much less exponential squared.

3.1.6.4.7 Downsampling using Simulated Annealing Downsampling the recorded dataset of the previously discussed mobile sensor optimization problem can be used to reduce the amount of storage needed and decrease the runtime of the modeling algorithm. Because the data is used to model the unknown signal, it is important that the reduced version be accurate because it directly affects the placement of all agents in the system. Downsampling was chosen as the reduction paradigm that will be applied to the dataset. Downsampluing removes unnecessary samples from the dataset in order to minimize its size. Feature reduction was also attempted through Principle Component Analysis(PCA), which gave the results shown below; In the figure, the left plot shows the signal estimated from the whole dataset and the right plot the estimated signal from the PCA reduced dataset. The plots show that PCA completely removes the information from the data, showing that the number of features cannot be reduced by this method.

In order to implement downsampling, an optimiation function was developed that could reduce the dataset while maintaining the information of the system. The optimization function is shown below;

$$\text{minimize} \quad \|\Psi\Lambda - \Psi_s\Lambda\|_2^2 + \sigma \|w\|_1 \tag{44}$$

Where $\Psi\Lambda$ is the estimation of the signal using the whole dataset, $\Psi_s\Lambda$ is the estimation of the signal using the reduced dataset, σ is a scalar, and w is a mask with values that can be 1 or 0 that is used to signify which samples in the dataset will be retained. The equation has two components, a mean squared error between and a LASSO function. The mean squared error adds a cost for a large error between the estimation of the signal using the whole data set and the estimation based on the reduced data set. The error is measured based of a grid of points read from each estimation, then averages the errors between those readings. This component is designed to retain the information of the system in the solution because the cost increases as the models drift apart. The second function, the LASSO function, is designed to make smaller datasets more favorable then large ones. this is done by adding up the number of samples in the reduced data set then applying a scaling function. The smaller the dataset the smaller the sum, which reduces the cost of that solution.

The second part to implementing the downsampling is to determine how new solutions to the problem will be found. There are two methods to do this, gradient based methods and gradient free methods. Because the derivative of the optimization function is very hard to compute, a gradient

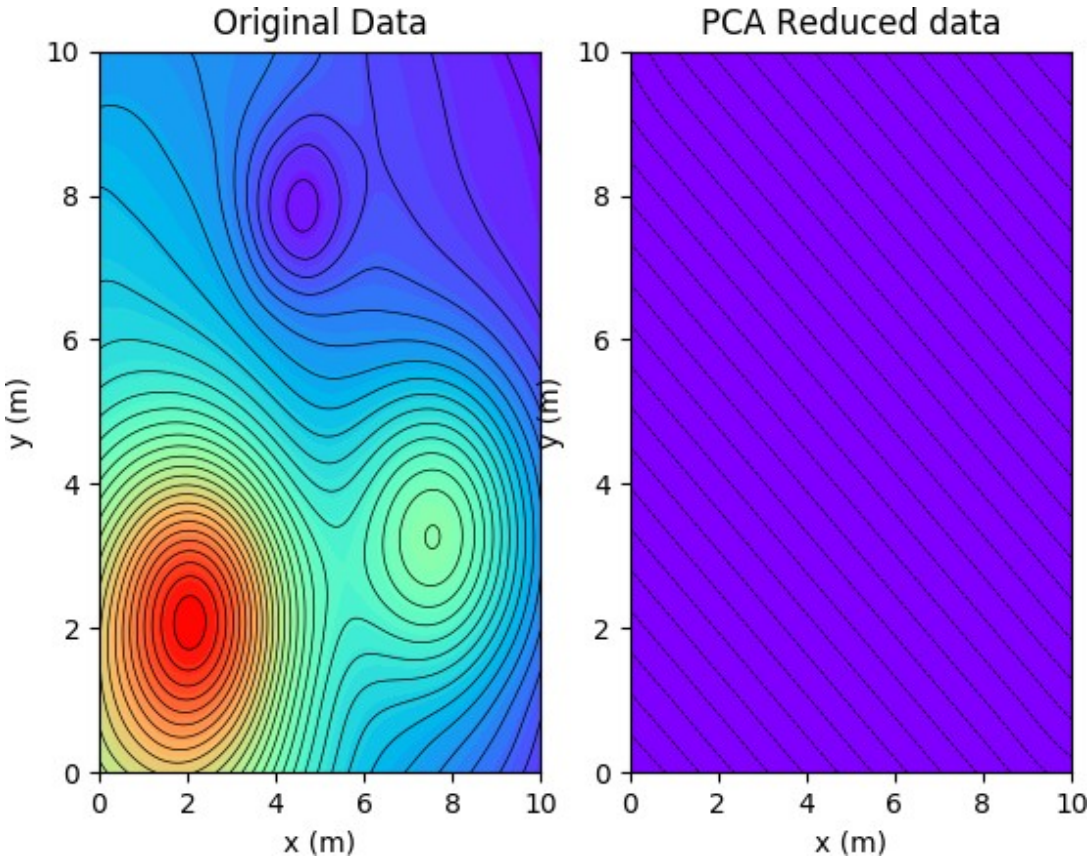


Figure 88: Comparison of models produced from original data set vs. PCA reduced data set with 2 eigenvalues

free method was chosen. The gradient free method that was decided on was simulated annealing. Simulated annealing is based on the process of annealing metal, which involves heating up the metal then slowly cooling it in order to make it stronger. In simulated annealing, an 'energy' function is used to represent the temperature, and instead of optimizing the crystal structure of the metal the goal is to optimize a given objective function. The general algorithm runs by first creating an initial energy along with an initial random solution. The algorithm then randomly generates a new neighbor, which can be anywhere in the sample space. This neighbor is found by changing one on more elements of the current solution, in this case by setting some data values to 0. Once the neighbor is generated, the cost function of that neighbor is found, and the algorithm then uses the energy function to determine if it will move the current solution to this new neighbor solution. This step is why simulated annealing is able to find an approximate global solution, it allows itself to occasionally move to solutions with a worse cost function then the current solution, enabling it to not be caught in local minimums. The decision on whether it will move to smaller values is based on the energy function, the higher the energy the more likely it will accept a worse cost function. Lastly, the energy function is decreased by some amount and the process is repeated. The simulated

annealing algorithm will be applied to this problem by randomly switching a value in w between 1 and 0 to form a neighbor. Because of the LASSO function the final solution should favor sparsity, ensuring that the algorithm minimizes the problem to the best of it's ability.

The algorithm has six variables that can be edited by the user during initialization. The first two are the number of epochs and number of trials. Each epoch will have a different temperature value, while the trials are the number of runs using each temperature value. The next variable is the temperature schedule to use, with four schedules being allowed. These four equations are shown below:

$$\text{Exponential}: y = a^t,\ t = 0, \cdots, epochs \tag{45}$$

$$\text{Linear}: y = -\sigma t,\ t = 0, \cdots, epochs \tag{46}$$

$$\text{Log}: y = \frac{1}{1 + \sigma \log(1 + t)},\ t = 0, \cdots, epochs \tag{47}$$

$$\text{Quadratic}: y = \frac{1}{1 + \sigma t^2},\ t = 0, \cdots, epochs \tag{48}$$

The two variables a and σ are constants for the temperature, and can be adjusted to change the

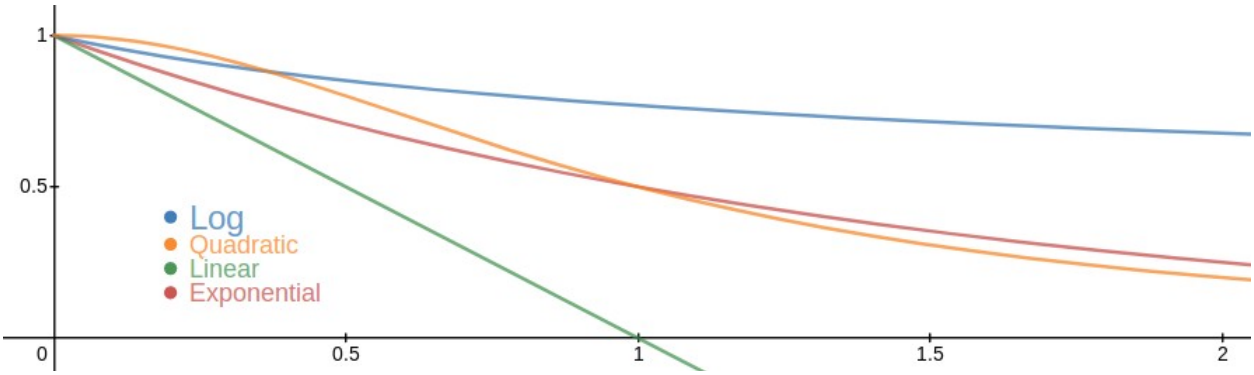


Figure 89: Temperature schedule plots for exponential, linear, logarithmic, and quadratic functions.

shape of each plot. The variable a is used to determine the base of the exponential schedule and σ is a scaling factor used for the logarithmic and quadratic schedules. The final variable is to determine how verbose the output should be, ranging from 0 to 3, with 0 being no output and 3 being a detailed description of how the system is running. When the algorithm is initialized, the temperature schedule is formed automatically. Once the initialization is complete, the user can provide a data set to be minimized. When data is provided, the model of that full data set is formed. For each epoch, the neighbors that are evaluated for each trial are formed in one batch. This is done so the evaluation of their costs can be calculated in parallel using Python's multiprocessing library. The formation of the neighbors cannot be parallelized in this implementation because each neighbor is based on the previous. It would be possible to further parallelize this code by making each trial independent. Each parallel process returns a float value representing the cost of that model. The costs are collected in an array, which is iterated through to determine whether each solution will be accepted. The costs are chosen in this way to emulate how the sequential code would see the same data. The criteria for accepting a solution is that it has a lower cost or that a randomly generated number is larger then a generated probability based on the difference between the two solutions and

the current temperature, shown below:

$$p = e^{\frac{dCost}{temp}} \tag{49}$$

After each epoch is run, the final weights are returned to the user to apply to the data.

Three tests were conducted to evaluate the performance of the algorithm. The first involves using various combinations of schedules, number of epochs, and number of trials in order to assess the performance of each. The second involves using different amounts of data to assess how the system's performance changes. The final test will use a data set from the mobile sensor optimization problem to assess how well the algorithm is able to minimize data from an actual experiment.

The first test involves analyzing all combinations of the the schedules, number of epochs, and number of trials in order to show general trends in the performance of each. Each combination was run on the same data set of 200 points 20 times. Furthermore, sigma was set to 1 and the exponential schedule had an a value of 0.5. The number of points removed, the error of the final model, and the time to run were all saved in an xarray data set. Xarray is a Python library that allows the creation of n-dimensional arrays, which is necessary for this data set because of the numerous variables. Data was collected for the four previously mentioned schedules along with 5, 10, and 25 epochs and trials.

The points removed by each combination can be seen below in Figure 90.

The box plots from Figure 90 reveal some general trends from the combinations. First, increasing either the epochs or the trials results in more points being removed from the data. With only a few exceptions, increasing the number of trials resulted in 75% of the data being greater then the medium of the previous amount of trials. In many of the plots, increasing the number of trials also has the effect of increasing the spread. Increasing the number of epochs has a greater effect on spread in the logarithmic and linear schedules then the exponential and quadratic. Out of the four schedules, linear resulted in the best medium for 25 epochs and trials at a value of 64, while quadratic had the best minimum for 25 epochs at 41, excluding an outlier.

The average error of each schedule for all combinations can be seen below in Figure 91.

From the errors it can be seen that the linear and logarithmic schedules almost always have the highest error, with linear becoming comparatively much higher then the others. Though it is high compared to te other schedules, the maximum error of 2.8 is still very low, which will be further discussed later. When viewing these errors alongside Figure 90, it is obvious that the higher spread in the box plots and the average errors of the bar graphs are correlated. This doesn't come as a shock, in order to remove more points from the data some error must be added to the system.

The final variable that was recorded was the average time to calculate each combination, which is shown in Figure 92. From the plot it can be seen that the run-time is completely independent from the schedule chosen. Furthermore, the run-time of the algorithm is very slow, ranging from 5 seconds up to 100.

The second test is aimed at assessing how the algorithm functions with different amounts of data. For this test, only the exponential and linear schedules will be used at 10 epochs and 5 and 25 trials. The same data will be recorded as the previous test. Finally, the same σ and a value will be used as the previous test.

The amount of points removed is displayed below in Figure 93. The figure shows that as the number of points in the data set increases, the spread of the data tend to increase with it. This can be attributed to a drastic increase in the possible solutions to the problem.

Lastly, Table 12 shows the time to compute each combination of data points and trials. The

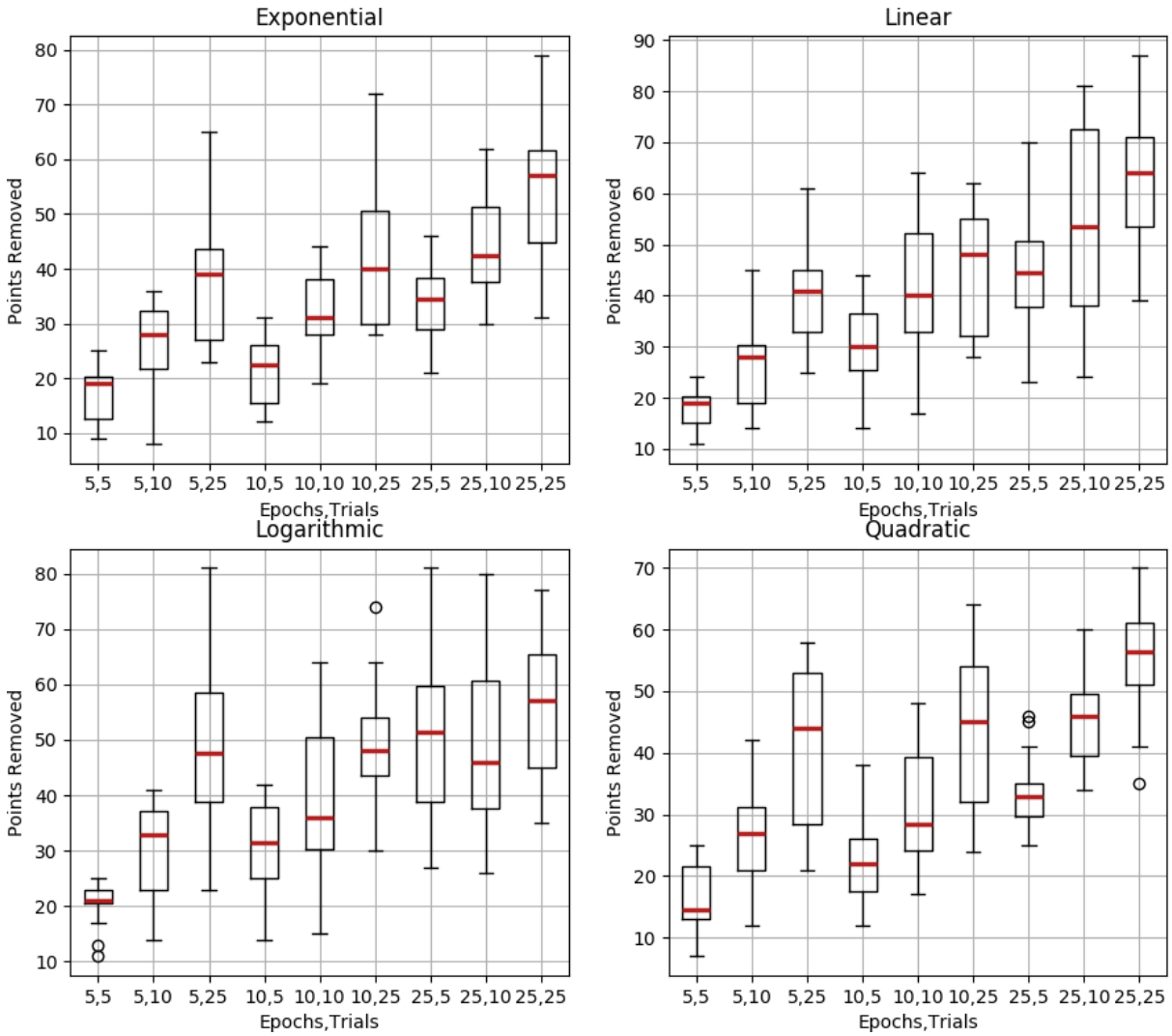


Figure 90: Box plot of the number of points removed by the exponential, linear, logarithmic, and quadratic schedules for 5, 10, and 25 epochs and trials. The y-axis is the number of points removed and the x-axis is the number of epochs and trials.

ratio between the time to calculate each solution and the number of data points being processed decreases as more data is added. This shows how using multiprocessing benefits the system more as more data is added.

The final test gives an example of the system being used in an actual application. The data collected is from a simulation of the robotic swarm discussed in the introduction. As a reminder, the robots collect data from an area, then use that data to form a model. The simulated annealing algorithm will take that recorded data and reduce it while maintaining the model. The example uses the simulated annealing algorithm with a linear schedule, 25 epochs, and 25 trials. The run has 183 data points, of which 70 were removed with an error of 1.263. From the figure it can be

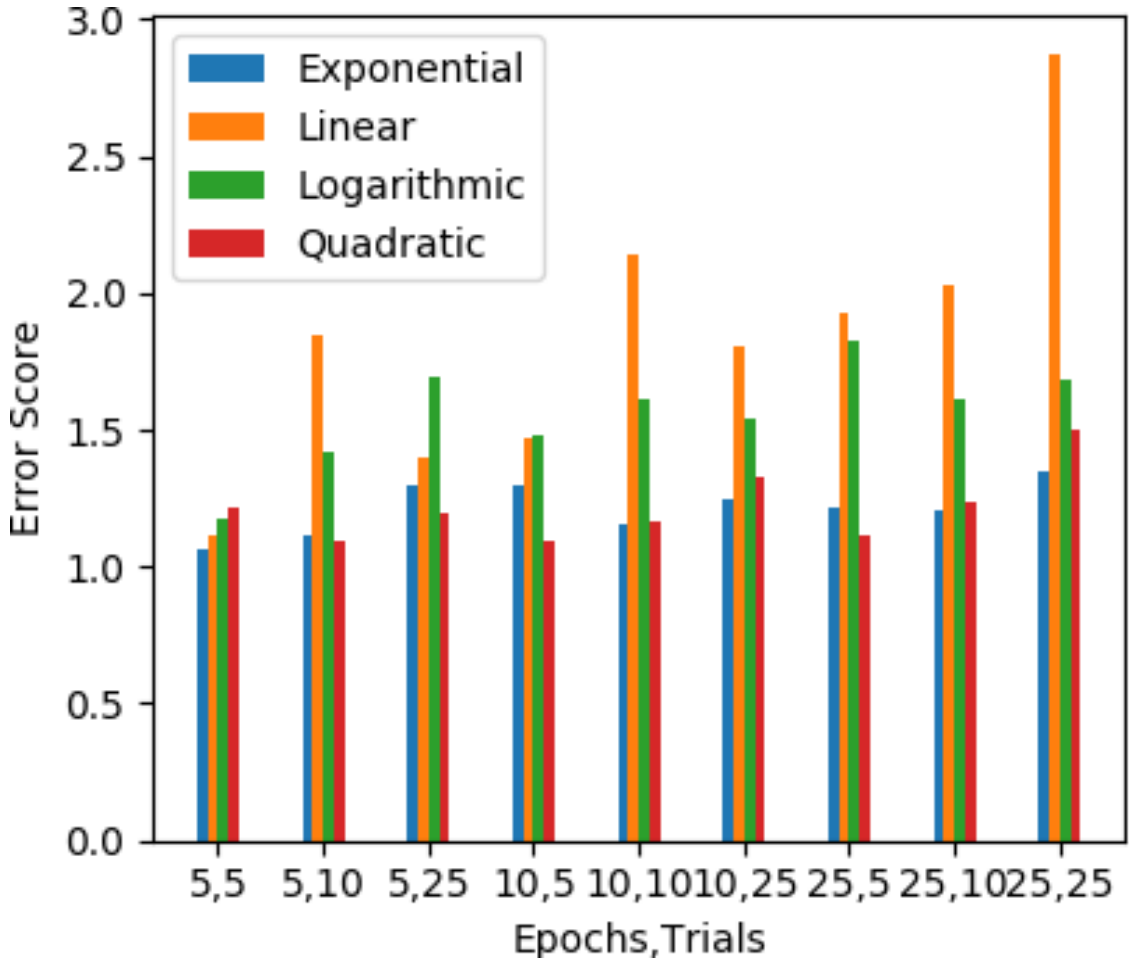


Figure 91: Comparison of exponential, linear, logarithmic, and quadratic schedule's error score for 5, 10, and 25 epochs and trials.

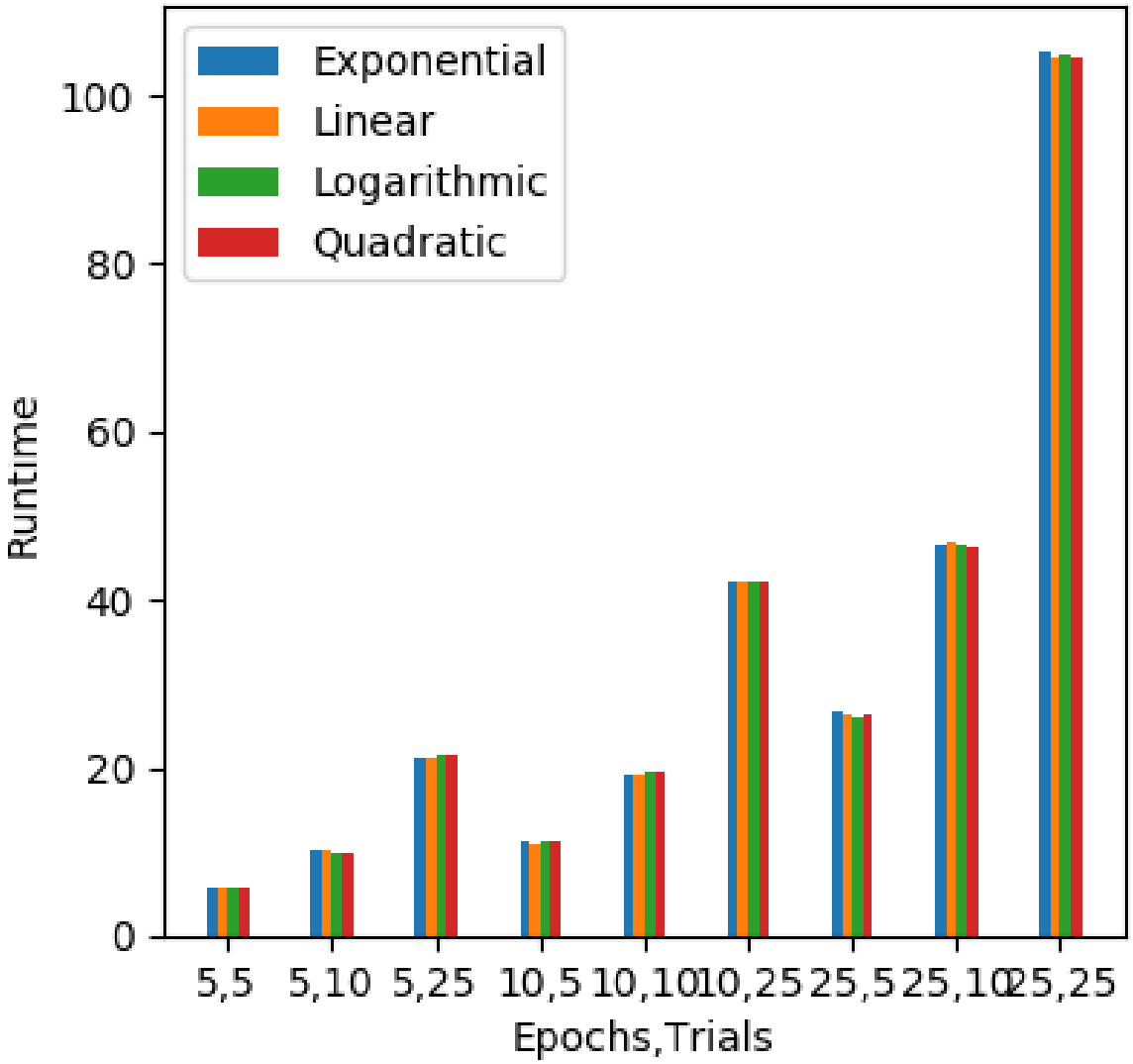


Figure 92: Run-time of exponential, linear, logarithmic, and quadratic schedules with 5, 10, and 25 epochs and trials.

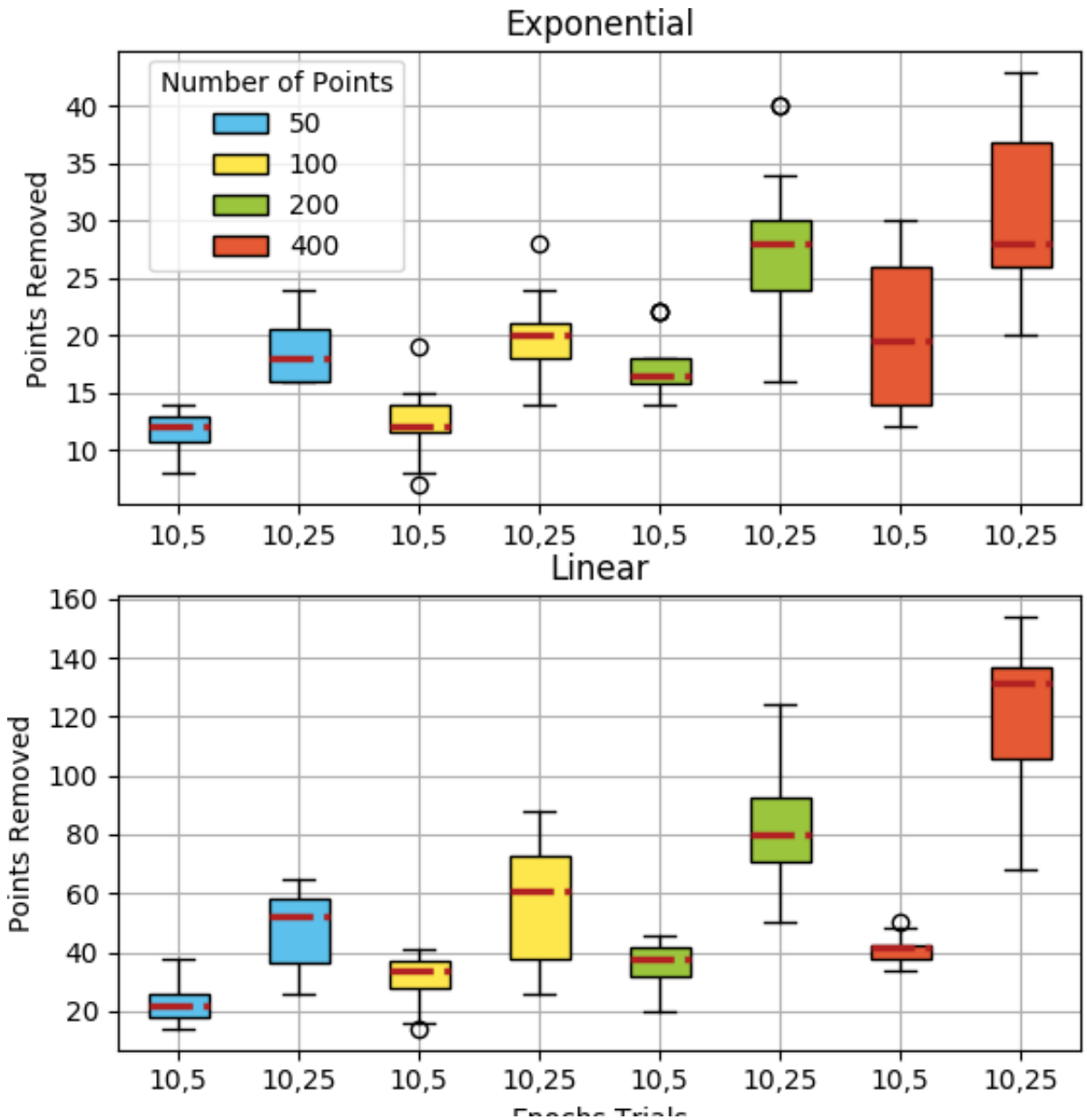


Figure 93: Box plots of points removed from 50, 100, 200, and 400 points by the exponential and linear schedules with 10 epochs and 5 or 25 trials.

seen that the model is retained well, even with 61% of the data being removed. Figure 94 shows the results of the run in the form of a topographical map of the sensor function. The left plot shows the original model with the recorded data points, while the right plot shows the reduced data points and model. From the figure it is apparent visually that the models are similar and would be able to be used interchangeably.

From these tests, it is shown that the algorithm does a good job at reducing data, but at the

Table 12: Time to calculate for 50, 100, 200, 400 data points with 5 and 25 trials in seconds.

	5	25
50	2.28	12.15
100	5.45	25.62
200	11.04	42.63
400	20.07	73.29

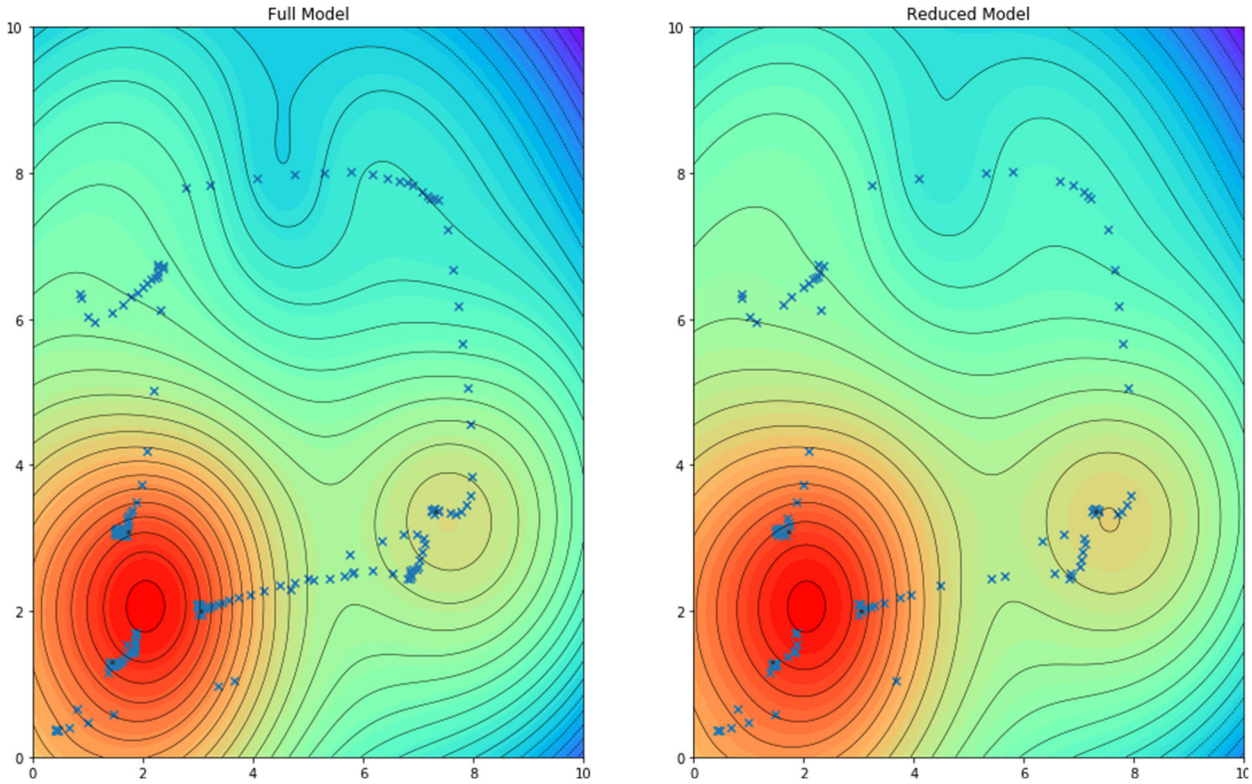


Figure 94: Comparison of original (left) and reduced models (right) from mobile sensor optimization simulation.

cost of being very slow.

3.1.6.4.8 Platooning of Autonomous Vehicles A platoon is a group of cars traveling together on the highway as a single unit. In this case, the vehicles would be autonomous and controlled by a mixture of local and cloud controls. The goal of platooning is to increase the safety and efficiency of cars on the highway. A platoon can increase safety on the highway by allowing the autonomous cars to send warning about hazards to other vehicles in the platoon, allowing them to better avoid the issue. Furthermore, the cloud can send information to a platoon about alternate routes that better balance traffic, which can help prevent congestion in cities and construction zones. Efficiency is increased through the use of drafting, which allows a line of cars to be more efficient by traveling in the wake of the first vehicle. A report released by the National Renewable

Energy Laboratory estimates that approximately 65.6% of total miles driven by semi-trucks can be platooned, resulting in a savings of "4% - 5% for the leading truck and 10%-14% for the following trucks." [19] Platooning can be applied in both the civilian and defense fields, where it can be used to improve supply chains for a company's fleet or a military force. Both of these fields present different challenges for the same problem. In civilian use, the platoon will always be on well paved roads, allowing for easier detection of lanes and other vehicles. Civilian platoons must also be compliant with more government regulations because they are on public roads. On the other hand, military platoons must drive in dangerous environments where hostiles are difficult to detect. Furthermore, military platoons will have less access to cloud infrastructure, forcing them to be more reliant on on-board computing. This project will focus on civilian platooning, which while simpler is still a difficult challenge that is a hot topic in current research.

In order to implement platooning, three basic problems must be solved; clustering, consensus control, and communication. Clustering is used to determine which vehicles should be in each platoon, which allows for the size of the platoon to change as vehicles enter and exit a highway. The clustering algorithm must be able to be calculated quickly because the number of vehicles on the highway is constantly changing. Due to the needed high speed and low latency, clustering is a good candidate for the Cloud-Edge architecture, which may allow for better clusters while still allowing for the necessary performance. The second problem that needs to be solved is consensus control, which is needed to ensure that each vehicle is collision free and maintains the same distancing and speed. Each vehicle will take information from it's own sensors, information passed from it's neighbors, information from infrastructure, and finally information from the cloud. Each vehicle must then determine how it will avoid collisions with it's neighbors and what controls it needs to reach its destination. Maintaining distance and speed can be implemented by adapting current adaptive cruise control technology to take into consideration information from other vehicles. The final problem is communication, which must be able to send information reliable between each vehicle in the platoon and from vehicles to the cloud. Communication between vehicles is already standardized by IEEE under IEEE 802.11. The standard focuses on Vehicle Ad Hoc Networks (VANET), which is the spontaneous creation of wireless networks between vehicles. This type of network changes rapidly as vehicles enter and exit the highway or travel in and out of the range of a vehicle's communication area. This project will focus on improving communication between vehicles through a method called Event-Triggered Scheduling Mechanisms (ETSM). ETSMs send messages only when a event threshold is met, which allows for messages to only be sent when needed. This is preferable to constantly sending messages or sending them at a set rate because it saves bandwidth for the network. ETSMs can be further improved by dynamically adjusting the threshold for sending a message, which can further fine tune how often messages are sent. By using ETSMs, the limited bandwidth of a VANET can be used more efficiently and the performance of the control can be maintained with less messages because only important ones are sent.

We decided to focus on the field of clustering. There are a number of topics that are relevant to clustering vehicles on a highway, such as clustering Mobile Ad-hoc NETworks (MANET) or clustering based on information exchange. The subject of MANETs is very similar to clustering vehicles, which use a VANET to communicate between vehicles. Both VANETs and MANETs deal with agents that are moving with respect to each other, but in VANETs the agents are specifically vehicles and roadside infrastructure while MANETs are any wireless devices. Furthermore, VANETs have more regular movement because they are confined to a roadway. Because of their similarities, clustering methods developed in MANETs can be modified to function in VANETs. Many MANET

clustering methods involve the use of a cluster head, which acts as a leader for a cluster. The clusters are formed from the cluster heads, and information between clusters is passed through their cluster heads. Cluster head election can be based off it's speed, direction, neighbors, ability to forward packets to other cluster heads, and more. Clusters are then formed using the cluster heads based on variables such as the destination of vehicles, the size of the platoon, and the estimated platoon lifetime. In many cases, the clusters act as both the vehicles in a platoon and as a topology for network communication between agents, so optimal communication between vehicles can also be taken into account when forming clusters.

Simulation of traffic will be an important component to this project, and as such a robust and accurate simulator will be needed. Two possible software suites that can be used are Paramics [20] and Sumo [21]. Paramics has been used by both researchers and industry to analyze traffic flow. It is developed by SYSTRA, and is designed for use by governments and industry to provide simulations of roadways. Sumo is another traffic flow simulation, but unlike Paramics it is open source and free to use. Sumo is developed by Eclipse, and focuses more on research then industry. Both methods are microsimulations of traffic, meaning that they simulate individual cars in the system. For this project it is recommended to use Sumo because it is open source. This allows Sumo to be modified as needed for the project, and has allowed other researchers to combine different simulation environments with Sumo in order to simulate vehicle to vehicle communication, vehicle to infrastructure communication, and platooning.

A number of different clustering methods were investigated, which will be discussed below. These methods were fuzzy based clustering, location based clustering, mobility based clustering, and finally destination based clustering.

Fuzzy based clustering uses fuzzy logic to determine clustering. Fuzzy logic is a method that allows uncertainty to be integrated into clusters by letting individuals be part of multiple sets. Fuzzy logic also allows for uncertain expressions to be quantified, such as most, least, a little, ect. With respect to clustering in VANETs, fuzzy clustering allows vehicles to identify with multiple clusters on the road. A lot of research has been conducted using fuzzy clustering to determine cluster heads, which are vehicles that will represent a whole cluster. In these applications, each vehicle has a probability to become a cluster head based on some metrics. These metrics vary between papers, and include measurements such as distance to other vehicles, velocity, energy, lifetime, and density of its region. If the probability for the vehicle crosses a threshold, then it may become a cluster head.

Location based clustering use the position of vehicles to determine if they should be in a cluster. Algorithms that fall under this category often choose a cluster head based on the vehicles position, then limit the vehicles that can be a member of its cluster based on a max distance. While these methods can perform well, knowing the absolute position of all vehicles on a roadway can be a difficult task, especially in a city or covered roadway such as a tunnel, where GPS does not function well.

Mobility based clustering methods use metrics such as velocity, acceleration, and direction to determine clusters. Mobility based clustering algorithms will often use these metrics to estimate the cluster head lifetime of a vehicle, which is the estimated amount of time a vehicle will be a cluster head. The higher this value is, the longer a vehicle can be a cluster head and the more stable the system is.

Destination based clustering uses the final destination of a vehicle to determine clusters. By knowing the final destination of a vehicle, its route there can be estimated. The vehicles are then

clustered based on the similarity of their routes, allowing for clusters to last longer. Algorithms in this category will often use metrics such as velocity and direction to determine clusters. It is worth noting that destination based clustering creates a security risk for users by allowing other vehicles to know their destination.

Each of these methods provide a unique solution to the problem of clustering vehicles on a roadway. Based on the research though, it was decided that mobility based clustering will best fit the goals of this project. The metrics needed for mobility based clustering are readily available with current vehicle technology and can be recorded reliably. Furthermore, the information is capable of being collected in all environments and requires no other infrastructure.

The implemented mobility-based clustering algorithm uses the vehicles velocity and location to determine what cluster the vehicle belongs in. The clustering algorithm that was implemented was designed by Rawshdeh and Mahmud in [22]. In their implementation, vehicles are traveling in a single direction down a multi-lane highway and pass their information between each other. The authors assume that each vehicle will send periodic messages to their neighbors about their current state and that vehicles will also gather information they receive. A cluster head is determined by a suitability function, which takes into account the number of neighbors the vehicle has, its distance from the center of its neighbors, and its relative speed to its neighbors. The goal is to maximize the time the cluster head stays connected to its neighbors, therefore increasing the stability of the system. The authors simulate vehicles driving on a single direction, multi-lane highway for 10km and vehicles are allowed to be in a cluster if the cluster head is within 200m.

In this implementation, some changes have been made to the work of Rawshdeh and Mahmud. Vehicles are simulated on an intersection of two interstate roadways instead of a single highway, and each highway is bidirectional. Furthermore, the clustering range was modified to better represent the capabilities of an on-board transmission unit, from 200m to 500m. The vehicles and roadways were simulated using SUMO and a realistic roadway was generated using OpenStreetMaps.

The simulation is run for 1000 time steps, which is equivalent to 1000 seconds or approximately 16.5 minutes. Vehicles are randomly placed along the roadways as the simulation runs and are removed if they reach their destination before the simulation terminates. For each time step, the vehicles update their databases and determine if they should become a cluster head or join a new cluster. The results show that the mobility clustering algorithm was able to successfully group vehicles on the roadway, but with some drawbacks. Vehicles with similar locations and velocities are clustered, but there is no test that they are traveling in the same direction, meaning that vehicles in opposite lanes may be clustered. Clusters also change often using this method, which can lead to instability in the system from vehicles constantly having to adjust to new clusters.

3.1.6.4.9 Roadside Units and their Placement Research in the field of cloud-based control for autonomous vehicles on a roadway revealed a field of study that will provide better capabilities to vehicle platooning then decentralized, mobility-based clustering methods. This field involves infrastructure supported intelligent transportation systems (ITS), which use roadside units (RSUs) to communicate with connected vehicles and pass information between them. Roadside units (RSU) are stationary infrastructure that are built along a roadway that are capable of communicating with connected vehicles on-board unit (OBU). RSUs can communicate with vehicles through a number of methods such as with cellular infrastructure or with dedicated short range communications. A common method in research is to use IEEE 802.11p, a network standard that is designed for wireless local area networks for vehicles. RSUs can communicate with each other over a

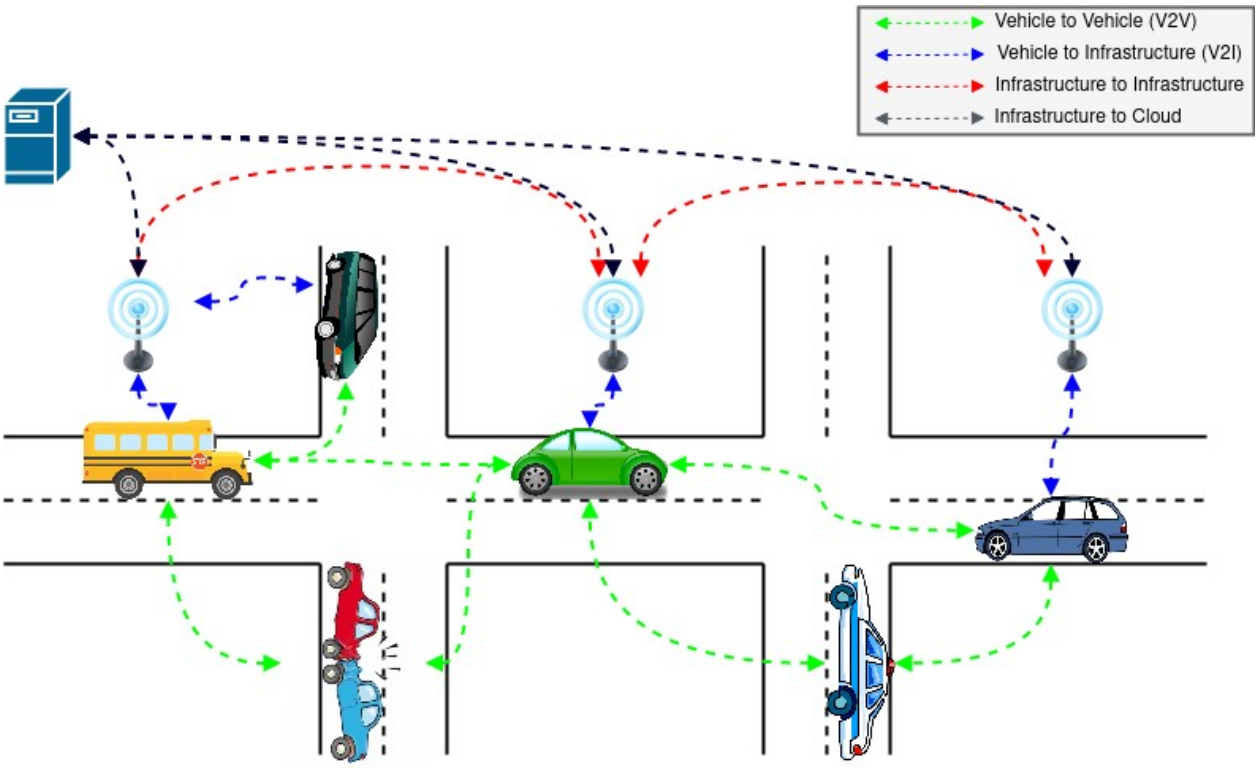


Figure 95: Example of an intelligent transportation system, showing the communication between vehicles, RSUs, and a cloud server. V2V communication is in green, V2I communication is in blue, infrastructure to infrastructure is in red, and finally infrastructure to cloud is in black.

wired or wireless connection, which allows them to share information. Figure 95 shows an example of an ITS with RSUs. In the figure, the communication channels between vehicles, infrastructure, and the internet is shown. Vehicles on the roadway detect a crash and pass that information to the nearest RSU, which can then send it to other RSUs in the network to maximize the number of vehicles aware of the accident. RSUs can be used for more then just communication, they can act as access points for cloud systems or even become components in a fog network by adding some computational ability to the RSU. This allows for more complex and advanced algorithms to be used then is possible with decentralized methods with the further benefit of collecting information from many more vehicles. RSUs can also be useful with respect to vehicle platooning by allowing for vehicles communicate through the centralized RSU, allowing for information needed for determining platoons to be passed easier. Using the fog computing framework, the RSU could even determine platoons on their own using information gathered from the roadway.

While it is possible to place RSUs at every intersections, the cost of implementing such a network would be astronomical. This leads to the problem of finding the optimal balance between number of RSUs and costs. This issue can then be expanded to determine not only the optimal number of RSUs, but the best location to place the RSUs to minimize the number of RSUs but maximize the performance of the RSU network. This problem is nontrivial to solve due to the sensitivity of an RSUs performance to the environment around it. Furthermore, a non-optimal placement for an RSU can lead to poor connection to vehicles which would reduce the benefit the RSU adds to

the ITS. Finding the optimal placement for RSUs is called the RSU placement problem. The RSU placement problem is a combinatorial optimization problem, meaning the problem must find an optimal subset from a finite set of objects. The RSU placement problem is similar to problems such as the set cover problem and knapsack problem, both of which are NP-hard. Using this similarity, it has been proven that the RSU placement problem is NP-hard, meaning that the optimal solution cannot be found in polynomial time.

With this information in mind, it was decided that the final task for thrust 1-6 would be to implement an algorithm to solve the RSU placement problem, which would allow for connected vehicles to easily share information and access cloud services. A survey of the literature was conducted, revealing a gap in the research. In the surveyed works, the majority of authors utilized either greedy or genetic algorithms to solve the problem. These algorithms have similar benefits; they are both generic and can be applied to a large range of problems, they are common algorithms that have a lot of documentation, and finally they are capable of solving NP-hard problems like the RSU placement problem. While these are significant benefits, there is also some downsides to the algorithms. With greedy algorithms, they are extremely prone to settling on a locally optimal solution when applied to a highly nonlinear problem such as the RSU placement problem. This can mean that the solution found does not meet the service requirements needed. Genetic algorithms can have a similar issue called premature convergence. Genetic algorithms work with a population of potential solution to the problem, and premature convergence is when these potential solutions all converge to a single non-optimal solution. There are some methods to attempt to avoid this problem, but they can take time to fine tune. There is a more modern field that is capable of handling similar problems that did not appear in the literature, which is neural networks.

Neural networks (NN) are a serious of algorithms that aim to learn relationships in data by analyzing a large set of examples. NNs are based on the human brain, where a set of neurons learn and remember relationships in information recorded by the human body. NNs are made up of three basic sections: the input layer, the hidden layer, and the output layer. The input and output layers are straight forward, they are responsible for reading in data and printing out the result of the NN. The hidden layers are responsible for processing data and learning the relationships in the data between the input and output. NNs can be applied to a wide range of problems and inputs, for instance it can segment an image, translate a text between languages, and detect faults in a machine through sensors. Based on the ability of a NN to learn complex relationships in data and take in a wide variety of input data, it is not unreasonable to apply the topic to the RSU placement problem. Using a NN to solve the RSU placement problem provides some unique benefits that makes it a tempting alternative to the previously described methods. Because NN do not need to be explicitly told how to solve the problem, they can find trends in the data that may not be apparent to a researcher. A model of the system also does not need to be made, the NN only requires sensor readings to determine an output. This is especially useful in the RSU placement problem where many of the variables are extremely difficult to model, such as the relationship between the geometry of a roadway and the optimal location to place an RSU. NN take a long time to train but they are very quick at generating a solution from input data, meaning that a trained NN could quickly create an RSU network for a provided road network. Lastly, a NN process input data that would not be possible with traditional optimization algorithms like the greedy and genetic algorithms such as time series.

The goal of this work is to prove that a NN can be used to find the optimal position of an RSU in a road network. This will be accomplished using images of the road network and a NN

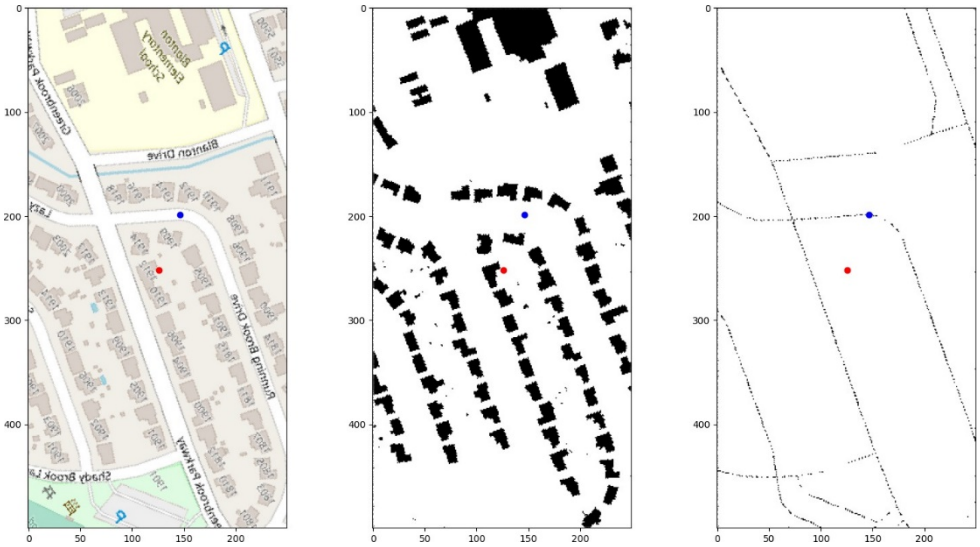


Figure 96: Example CNN input

that specializes in image processing called the convolutional neural network (CNN). The goal of a CNN is to take an image as input and condense in such a way as to retain only data needed to solve the problem. This is done by repeating two processes in the hidden layer; convolution and pooling. The convolutional layer applies convolution to the input image, which is responsible for extracting high level features from the input image. This extracted data is then reduced using the pooling layer, which shrinks the image into a smaller image. This process can be repeated multiple times to further extract information and compress the image further, with the final result being a compressed image containing important information. This compressed image is then flattened and fed into a traditional NN, which attempts to learn how to transform the compressed image into an output for the problem.

There is not a dataset available to train a CNN for the RSU placement problem, so one must be generated instead. It was decided that three images would be used to train the CNN; the map of the sample, the road traces of the sample, and finally the building traces of the sample. An example of the images can be seen below. Data on the roadways is obtained through OpenStreetMap, an open source geographic database of the world. Through OpenStreetMaps, information about roads and buildings can be queried and used to create the images needed for the data set. Before samples can be generated, the user must determine an area that they will be pulled from, for instance we used downtown Austin as the source for all our data. The size of the samples must also be determined, in this implementation samples are 250x500 pixels. The process for generating a sample is illustrated below: Each sample has four coordinates for its four corners and a random angle that the sample is rotated with. The rotation is necessary for increasing the variation in the data, allowing for more samples the be recorded from the same boundary. Using the coordinates of the sample, the road network and building traces are queried and similar images are created. Each sample contains these



Figure 97: Process for generating a sample from OpenStreetMaps

three images, the four coordinates of the sample, the coordinate of the center of the sample, the angle the sample is rotated, and finally the coordinate of the optimal intersection for the RSU. For this implementation, it was decided that the optimal intersection would be the one closest to the center, which was simple to calculate but complex enough that the CNN would still have to learn the relationship. A total of 5,000 samples were generated from the boundary in downtown Austin.

A simple CNN was created to solve the RSU placement problem. As mentioned before, the CNN takes in the three images of the roadway and outputs the coordinates that the RSU should be placed. The input for the CNN has the shape $(250, 500, 5)$ where 250 and 500 are the size of the sample and 5 is the number of layers; the R, G, and B layers for the map plus the two black and white images for the roadways and building traces. The input is passed through the convolution layer and max pooling layer, both with a 3x3 pixel filter. In order to reduce overfitting in the model, dropout is also added to the output of the max pooling layer. The CNN then branches out into two paths, one for the x coordinate and one for the y coordinate which both have the same structure. Each branch contains a dense layer with 256 neurons, an additional dropout layer, and an output layer of with one neuron. The mean squared error is used to train the CNN, with the error being the distance between the inferred coordinate and the solution.

The CNN architecture discussed above is trained for 200 epochs using the 70-30 split in the dataset. The model is trained with batches of 64 samples, and the Adam optimizer is used for training. Figure 98 shows the training history of the CNN model, and the results can be seen in Table 13.

The results show that the model is capable of successfully identifying the optimal intersection to place an RSU using just the images of the map, roadway, and building traces. Furthermore, based on the results of the validation data, the model is not severely overfitting the data. Figure 99

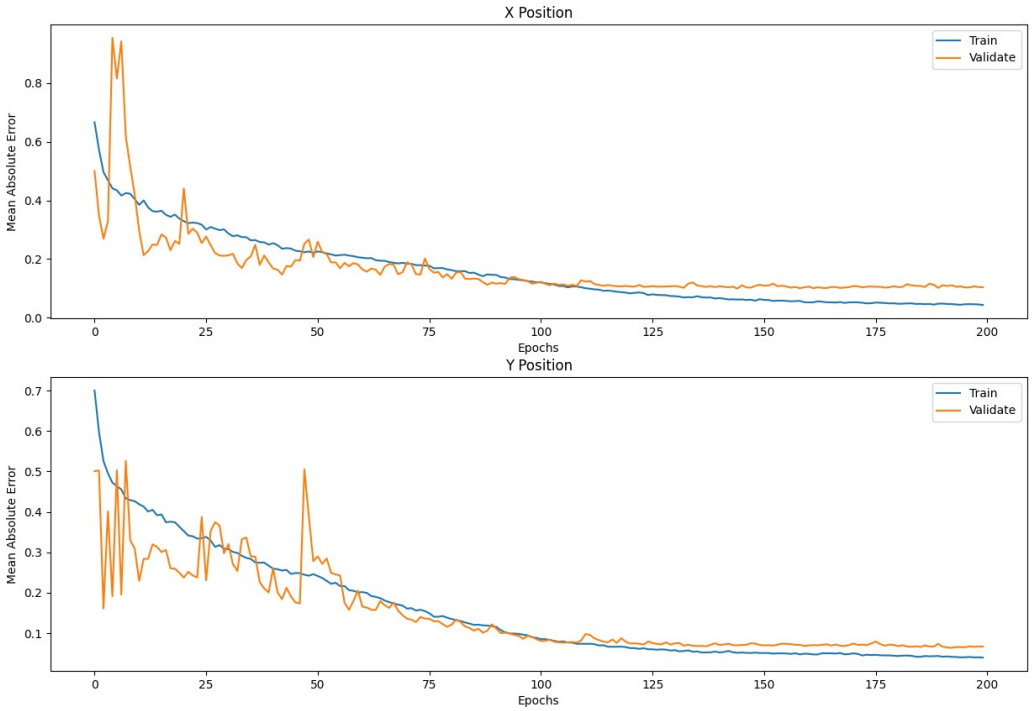


Figure 98: The mean absolute error of the CNN model as it trains over 200 epochs.

Table 13: Results of training CNN over 200 epochs with batch size of 64 samples.

X coordinate MAE	Y coordinate MAE	X validation MAE	Y validation MAE
0.0437	0.0398	0.1038	0.0667

demonstrates an example of the model predicting the correct intersection to place an RSU. These results will open the door for new research in the field of RSU deployment, and through it the use of cloud computing in the field of intelligent transportation systems.

3.1.6.5 Summary

Task 1-6 was responsible for researching and implementing cloud computing into the field of LSASVs. Different algorithms related to cloud based robotic control were investigated, along with methods for collecting data and transmitting it to the cloud. Furthermore, small-scale testing environments were researched and applied, including a robot-in-the-loop simulation. Industrial cloud computing infrastructure was utilized in the research to help verify the performance of the algorithms. The effect of the architecture of the cloud based controls was also investigated, with a focus on the cloud-edge architecture. The cloud edge architecture dynamically switches between two different algorithms, one which provides a quick solution and one which provides and accurate solution. This architecture allows the cloud based controller to react quickly to the environment while also providing accurate solutions. Three different applications were researched and implemented; autonomous boats, mobile sensor networks, and intelligent transportation systems. For each of these fields the uses of cloud computing were researched and implemented, and tests were done to assess the ability of cloud computing to control the agents in each environment.

3.2 Project Progress and Project Plans for Thrust 2: Resilient Control and Communication of Large-scale Autonomous Vehicles (RC2LAV)

3.2.1 Project Progress for Task T2-1 (Developing a decentralized fault detection mechanism)

3.2.1.1 Period of Performance under

Task T2-1 Start Date: May 2015

Conclusion Date: Dec 2021

Faculty lead: Dr. Ali Karimoddini

3.2.1.2 General Description of Task T2-1

The first step to handle a fault is to accurately detect the occurrence of the fault, and its nature and location. It is not possible to consider a sensor to detect any fault, as it dramatically increases the overall costs and more, importantly, not all faults are predictable. Instead, it is reasonable to diagnose a failure through system behavior and limited observations. Fault detection and isolation is part of the decision-making unit of the system in which the system behavior has been abstracted to a discrete model and the fault occurrence can be considered an event which changes the operational mode of the system. As described above, most current research addresses fault detection and isolation for a single agent. To address fault detection for a team of agents, we will first characterize the nature of fault for cooperative tasks. We will then propose a diagnoser for a multi-agent system and investigate its decomposability into local diagnosers which form a decentralized fault detection structure in combination with local supervisors. Abstracted data received in the supervisory layer of the control structure will be used to detect a fault. Due to the discrete nature of the supervision

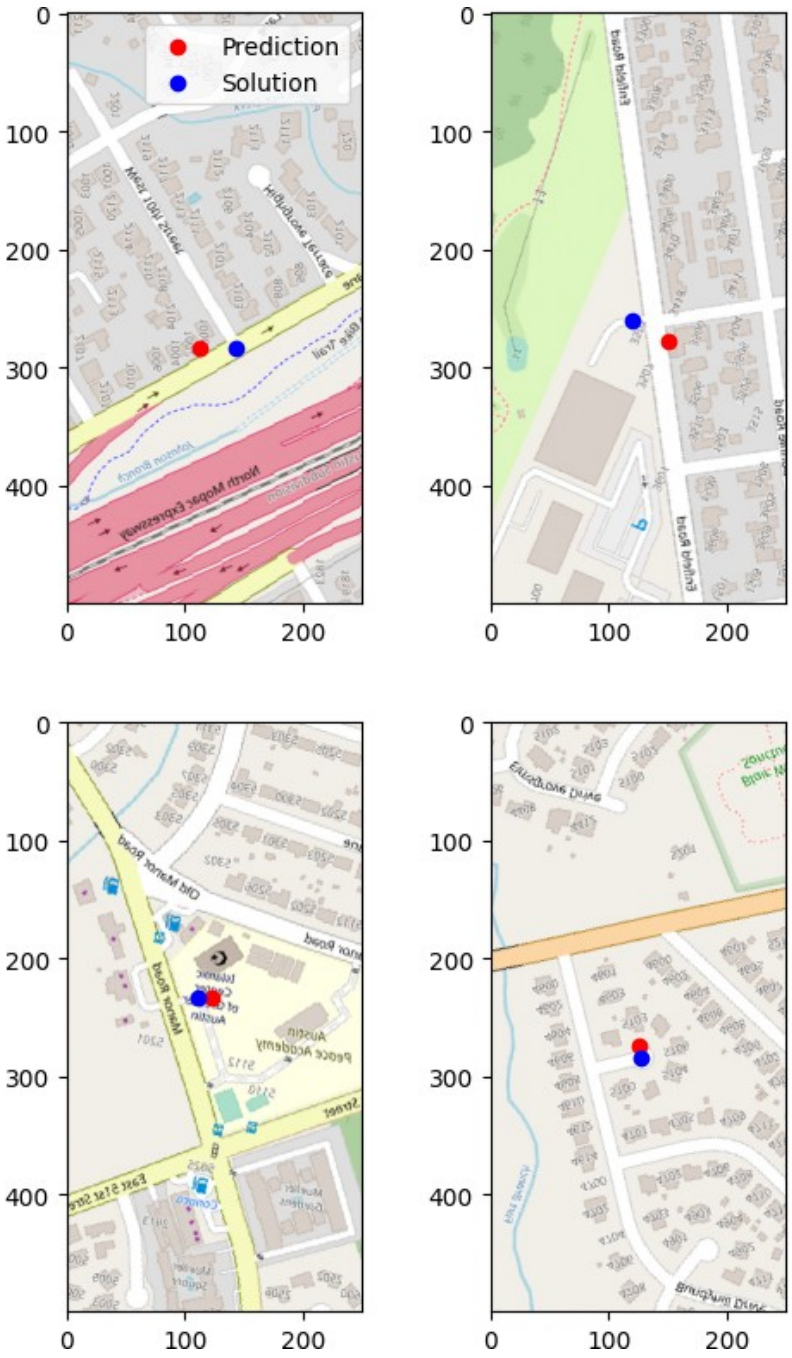


Figure 99: Four example outputs from CNN the error between the estimated and actual solutions for optimal RSU placement.

layer, we will study fault detection in the context of discrete event systems, automata theory, and supervisory control of discrete event systems.

3.2.1.3 Progress Against Planned Objectives in Task T2-1

The objective of this task is accomplished. Several algorithms are developed to:

- Formulated fault diagnosis for uncertain/unknown/partially-known
- Developed fault diagnosis techniques for uncertain/unknown/partially-known
- Developed the Diagnosability Concept and Conditions for uncertain/unknown/partially-known
- Applied the developed active-learning diagnosis algorithm to a aicraft flight control systems
- Investigated the time complexity of the developed diagnosis algorithm

3.2.1.4 Technical Accomplishments in Task T2-1

By relaxing the synchronous initialization and operation of the diagnosis tools with the DES systems, the state and condition of the monitored system upon activation of the diagnoser become uncertain. Then, the problem is that how can one distinctively characterize a system's behavior (system state and condition) solely based upon a finite number of future successive external system observations? To address this problem, a systematic procedure will be developed resulting in a diagnosis tool, so called diagnoser. Because the diagnoser is not synchronously activated with respect to the original system, upon its activation, the diagnoser is completely unknowing of the original system's state and condition. In this situation, the proposed diagnosers start wide and narrow down their estimate of the original system's state and condition as they receive information through their observations. Since the system and diagnoser are not synchronously initialized with each other, the diagnoser must narrow down its estimate of the monitored system's state and condition with less information than it would have acquired if the diagnoser was synchronously initialized with the system. Upon activation, the diagnoser is unaware of the state of the system, as well as the trajectory of states that the system may have visited. This creates a void in past knowledge of the monitored system's behavior. For this problem, two main situations can be considered: (1) *Asynchronous Fault Diagnosis* for which there is no information about the state and condition of the system understudy at the diagnoser's activation instance, and (2) *Semi-asynchronous Fault Diagnosis* in which the information about the state and condition of the system understudy is uncertain at the diagnoser's activation instance.

The second problem that is addresed in this task is the *diagnosability* of DES systems under asynchronous and semi-asynchronous activation of the diagnosers. The diagnosability refers to the feasibility of fault diagnosis for a given DES system. The diagnosability problem is to determine if all system fault occurrences can be definitively diagnosed within a finite number of system observations. For asynchronous and semi-asynchronous diagnosable plants, the diagnoser can diagnose any particular fault type (e.g., sensor, actuator, plant) denoted by F_i. If only a particular type of fault, e.g. F_i, is of interest, then the problem is reduced to the F_i-asynchronous and F_i-semi-asynchronous diagnosability, respectively.

It is also desired to know how long does it take for the developed diagnosers to diagnose system fault occurrences. The diagnosis delay can be characterized by the number of system transitions

required to diagnose an occurred fault, as DES systems are time-abstract event-driven. An accurate number of transitions for detecting any particular type of fault depends on the state of system understudy, which is not available to the diagnoser. Therefore, it is not meaningful to find the diagnosis delay at all states of the system. However, it is possible to find an upperbound for the number of transitions required to diagnose an occurred fault, as will be addressed in this dissertation under both asynchronous and semi-asynchronous conditions.

The other problem we are addressing in this task is one where the discrete event system is completely unknown. From the observable set of generated events (*strings*), we should determine if a fault has materialized and in the occurrence of a fault, we must then diagnose the type of fault that has transpired. To address this problem a systematic active-learning technique is developed for constructing a fault diagnosis tool for an unknown finite-state Discrete Event System (DES). The developed tool, called diagnoser, detects and identifies occurred faults by monitoring the observable behaviors of a plant. The proposed algorithm utilizes an active-learning mechanism to incrementally complete the information about the system. This is achieved by completing a series of observation tables in a systematic way, leading to the construction of the diagnoser. It is proven that the proposed algorithm terminates in a finite number of iterations and returns a correctly conjectured diagnoser. The resulting diagnoser is a deterministic finite-state automaton and is proven to consist of a minimum number of states. A sufficient condition for diagnosability of the system under diagnosis is derived along with illustrative examples to provide further clarification in detailing the steps of the proposed algorithm.

The next problem that we are addressing in this task is the case where we consider a discrete event system in which partial-knowledge is known in the form of an automaton. Given the known part, we must construct a language-equivalent model of the complete system. To address this problem a systematic active-learning method for realizing a partially-known Discrete Event System (DES). The proposed technique takes the available information about the system into account by tabularly capturing the known data from the system, and then, discovers the unknown part of the system via an active-learning procedure. For this purpose, a series of tables is constructed to first infer the information about the system from the available data, and if unavailable, the developed algorithm collects the information through basic queries made to an oracle. It is proven that the developed technique returns a language-equivalent finite-state automaton model for the system under identification after a finite number of iterations. A real-world illustrative example is provided to explain the details of the proposed method.

The last problem that we are addressing in this task is one where the discrete event system under analysis is partially known. Given the known part of the system, we must first utilize the readily available information and then through analysis of the observable set of generated events (*strings*), determine if a fault has occurred. Upon the discovery of an occurred fault, we must diagnose the type of fault that has transpired. To address this problem we develop a novel active- learning technique for fault diagnosis of a partially-known system modeled as a finite-state DES. The proposed technique first tabularly gathers information from a diagnosis tool, termed *diagnoser*, constructed from the available information of the known part of the system and then executes an active-learning technique to iteratively capture information regarding the system to construct the remaining portion of the diagnoser correlating with the unknown portion of the system. This proposed method is achieved by systematically completing a series of tables through inference of observable behavior to an oracle in order to complete the construction of the diagnoser which is ultimately able to detect and identify occurred faults through the examination of the observable

behavior of the system. It is proven that the developed technique terminates after a finite number of iterations and returns a correctly conjectured diagnoser. Furthermore, we derived a sufficient condition for diagnosability of the system under analysis, which guarantees the diagnosis of any occurred fault in a bounded number of observations. In addition, a real-world example is provided to detail the steps of the proposed method.

In summary here is a list of the main accomplishments for this task include:

- Introducing the concepts of asynchronous and semi-asynchronous diagnosis
- Developing algorithms for the construction of asynchronous and semi-asynchronous diagnosers
- Formal definitions for F_i-asynchronous diagnosability, and F_i-semi-asynchronous diagnosability
- Deriving the necessary and sufficient conditions for F_i-asynchronous diagnosability and F_i-semi-asynchronous diagnosability
- Quantifying the diagnosis delay by the number of observations required to diagnose an occurred fault for asynchronous and semi-asynchronous fault diagnosis and determining the upperbounds for diagnosis delays for asynchronous and semi-asynchronous diagnosable systems
- Investigating the relations between different diagnosis schemes including, synchronous, semi-asynchronous, and asynchronous diagnosabilities.
- Applying techniques to the diagnosis of a flight control system
- Developing an algorithm for the construction of a diagnoser when applied to an unknown system.
- Deriving a condition to verify the diagnosability of a constructed diagnoser for an unknown system.
- Obtaining an upper bound for diagnosing a fault.
- Identification of a partially-known DES system by developing an active-learning technique to construct a language-equivalent automaton.
- Developing an algorithm for the construction of a diagnoser when applied to a partially-known system.
- Deriving a condition to verify the diagnosability of a constructed diagnoser for a partially-known system.
- Applying the technique to construct a diagnoser for a partially-known system to an aircraft takeoff procedure with Boeing 737 Maneuvering Characteristics Augmentation System (MCAS) as the case study.

3.2.2 Project Progress for Task T2-2 (Fault Diagnostics and Prognostics through Data Analytic Approaches)

3.2.2.1 Period of Performance under

Task T2-2 Start Date: Q4 2015

Conclusion Date: Q2 2021

Faculty lead: Dr. Mo Jamshidi

3.2.2.2 General Description of Task T2-2

The three terms of diagnostics, health monitoring and prognostics are sometimes intertwined and often assumed to be interchangeable. They are related, but not the same. In the former, diagnostics identifies the nature or cause of some phenomenon, while in the latter, health monitoring keeps track of current status systematically with a view to collect information, while prognostics refers to the prediction about how something (such as sensor or actuator failure) will develop in the future. While in the previous task our focus was on the fault detection through general behavior of the system, in this task we are concerned with detection of fault in a system of vehicles through lower layers of the control hierarchy. For this purpose, we are essentially dealing with large amounts of data being exchanged among the vehicles both in simulation and on real-time robotic agents. We will utilize "Big Data" analytic techniques to detect failed sensors, actuators, eliminate noises, or cyber-attacks in data transmissions. Using our developed "Deep Learning", the raw data will be pre-processed and post-processed to create mined data and eventually predict failures of vehicles.

3.2.2.3 Objectives for Task T2-2 over the Course of Project

The following are the objectives for each quarter along with the impact of the objectives

- **Objective Name:** Conduct assessment of fault diagnosis technologies/methods/techniques for certain and uncertain systems.

  Objective Type: Research and Experimental

  Objective Description: Learn theory and application of data analytic approaches using existing tools, hardware, and software, common to the industry.

  Impact: Observe any popular and/or successful techniques for conducting fault diagnosis to identify patterns that may direct research toward innovation.

- **Objective Name:** Develop fault diagnosis techniques for systems of systems including autonomous UGV/UAV platforms.

  Objective Type: Research and Experimental

  Objective Description: Develop custom systems for prevention, detection, and correction of faults in systems of systems applications.

  Impact: Results in more robust and reliable systems of systems which are capable of correcting themselves by use of data analytics, logic processing, and classical or intelligent controllers.

- **Objective Name:** Create representational and extensible simulated environment for study of UGV systems behavior and for development of their fault detection/correction systems.

  Objective Type: Research and Experimental

  Description: Created simulated environment to conduct rapid prototyping and benefit from source control management systems.

  Impact: Allows for rapid and isolated prototyping of multiple research directions and configurations at once, as well as the ability to backup and rollback specific moments in the research process.

3.2.2.4 Progress Against Planned Objectives in Task T2-2

Advances were made in the following items:

- Created solution: "Diagnostics and Prognostics Framework".
- Created fault diagnosis system for Kobuki Turtlebot 2
- Implemented fault diagnosis system for cloud-connected UAVs in landing and take off conditions.
- Created comprehensive fault diagnostics system, "threat mitigation service", for Pioneer 2.
- Created simulated environment for study of UGV autonomous navigation and fault detection/correction systems.

3.2.2.5 Technical Accomplishments in Task T2-2

Fault diagnostics is the identification of the nature or cause of some phenomenon in a system. These phenomena are typically due to errors from subsystem components that are measurable by various sources of performance data. The goal is to develop a means to conduct fault diagnostics and prognostics through data analytic approaches. This research aims to explore the possibilities of such an architectures. During the first phases of development, the objective was to find existing algorithms of fault diagnosis and prognostics frameworks. A comprehensive survey was conducted, articles were selected for investigation. From the research, techniques were selected and reused or modified in search of innovation.

The repertoire of algorithms used during development include and are not limited to:

- Principal Component Analysis
- Multi-Basis Clustering
- Optimized Cluster Tracking
- Genetic Algorithm
- Reinforcement Learning
- Recurrent Neural Network

- Long Short-Term Memory

Fault diagnostics and prognostics framework prototypes were developed for Kobuki Turtlebot 2, DJI Phantom 3 drones, Pioneer 2, custom UAV, and custom UGV solutions. Hardware implementations involved designing the plant, selecting the specific objective to be solved, and modifying existing technologies to create a solution.

The simulated environment was created on the Unity engine since it allows for set up of 3D environments and the definition of behaviors through its scripting capabilities. Unity's ML-Agents package was selected as the source for the sim's intelligence due to it being PyTorch-based and possessing configurable implementations of algorithms such as reinforcement learning, imitation learning, and more. The sim is characterized by objects running "Agents" from the ML-Agents package. The environment itself consists of a track that has "way point" objects distributed throughout and a vehicle object which is being controlled by an Agent.

Summary of main accomplishments for this task:

- Created algorithm for fault diagnostic based on Multi-Basis Clustering, Optimized Cluster Tracking, and Principal Component Analysis.
- Created algorithm for fault diagnostic expanding upon DPF by employing an unsupervised clustering technique.
- Implemented test bed based on Kobuki Turtlebot 2 for testing fault diagnostics.
- Implemented DPF algorithm to Kobuki-based test platform.
- Reconstructed faulty signals acquired from DJI Phantom 3 drone on landing and take off conditions.
- Implemented Pioneer 2 based test bed and DAQ capability for testing fault diagnostics.
- Implemented LSTM-based model to Pioneer 2 test bed
- Created fuzzy logic based threat mitigation service on Pioneer 2 test bed
- Created Recurrent Neural Network implementation for detecting unknown faults
- Extended fault detection algorithm with cloud capability
- Created simulated environment for rapid prototyping

3.2.3 Project Progress for Task T2-3 (Developing a decentralized fault accommodation mechanism)

This task has been merged with Task 2-1. A close-out report is submitted under task T2-1.

3.2.4 Project Progress for Task T2-4 (Internet of Things (IoT) 5G Networks with Game Theory Applied to Resource Allocation and IoT Security)

3.2.4.1 Period of Performance under

Task T2-4 Start Date: May 2015

Conclusion Date: December 2022

Faculty lead: Abdollah Homaifar & Brian Kelley

3.2.4.2 Progress Against Planned Objectives in Task T2-4)

In the previous reports, we studied the problem of 3D deployment of a set of heterogenous Unmanned Aerial Vehicles (UAVs) acting as the flying base stations (BSs) to provide wireless coverage in a given geographical area. By allowing some overlap between the coverage area of different cells, we addressed an important UAV deployment challenge which is the inter-cell interference. First, we considered the case of orthogonal channels and found the maximal coverage of two UAVs in a rectangular area. Next, assuming that the UAVs operate in the same frequency band, i.e., non-orthogonal channels, we found the maximum coverage in the presence of interference. In this report, we consider the downlink scenario in UAV-supported wireless network where the goal is to minimize the total required transmit power of UAVs while satisfying the users' rate requirements. To this end, the optimal locations of UAVs as well as the cell boundaries of their coverage areas are determined. To find those optimal parameters, the problem is divided into two sub-problems that are solved iteratively. In the first sub-problem, given the cell boundaries corresponding to each UAV, the optimal locations of the UAVs are derived using the facility location framework. In the second sub-problem, the locations of UAVs are assumed to be fixed, and the optimal cell boundaries are obtained using tools from optimal transport theory.

3.2.4.3 Technical Accomplishments in Task T2-4

3.2.4.4 Introduction and Motivation

Recently, using aerial base stations to support ground cellular networks has received significant attention. Particularly, UAVs can act as aerialbase stations to support cellular networks in high demand and overloaded situations, or for the purpose of public safety and disaster management. The main advantage of using UAVs is that they do not need to have an actual pilot and hence they can be autonomously deployed in dangerous environments for the purpose of search, rescue and communication.

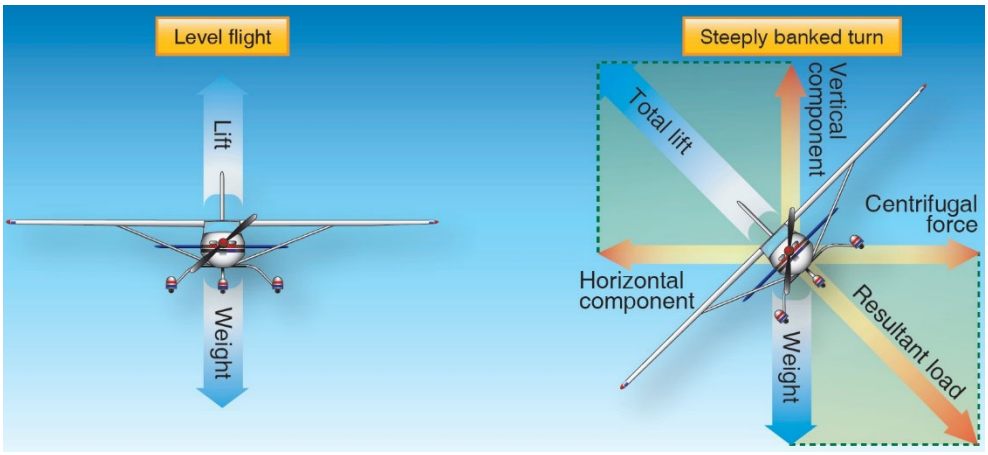


Figure 100: Level flight and banked turn.

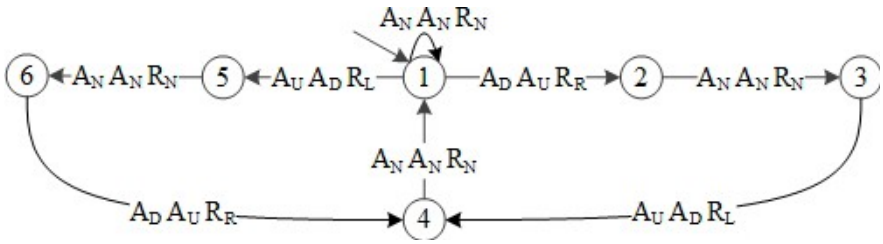


Figure 101: DES model of normal aircraft flight maneuvers.

purpose of public safety and disaster management. The main advantage of using UAVs is that they do not need to have an actual pilot and hence they can be autonomously deployed in dangerous environments for the purpose of search, rescue and communication. Furthermore, deploying UAVs acting as base stations is extremely useful in providing an improved quality-of-service (QoS) for ground users. The deployment of UAVs faces many challenges such as power consumption, coverage optimization and interference management.

To address the UAV deployment challenge, researchers provide an analytical approach to optimize the altitude of a single UAV for providing maximum coverage area on the ground. A UAV-enabled small cell placement optimization problem is investigated in the presence of a terrestrial wireless network to maximize the number of users that can be covered. The optimal flight altitude of a single UAV-BS operating under the Rician fading channel is derived. Researchers studied the use of multiple UAVs as wireless relays in order to facilitate the communication between ground sensor nodes. However, the work does not consider the use of UAVs as aerial BSs. Another research work, considered the use of multiple UAVs to compensate for the cell overload and outage in cellular networks. Beyond deployment, another important challenge for mobile UAV base stations is channel modeling. For instance, the probability of line of sight (LOS) for air to ground communication as a function of elevation angle and average height of buildings in a dense urban area was determined. The air to ground path loss model has been presented by some research works. As discussed in past researches, due to path l oss and shadowing effects of obstacles, the characteristics of the

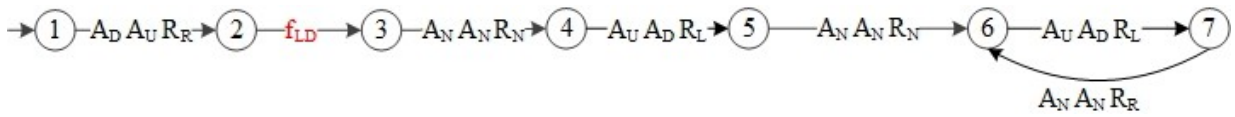


Figure 102: Flight commands for Left Aileron Stuck Down, f_{LD}.

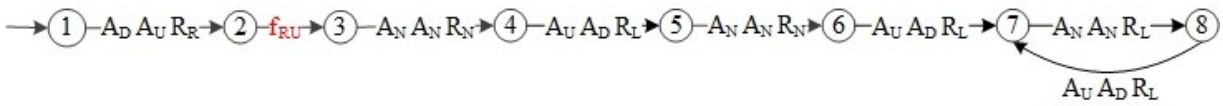


Figure 103: Flight Commands for Right Aileron Stuck Up, f_{RU}.

air to ground channel depend on the height of the aerial base stations. By increasing the altitude of a DSC, the path loss increases, however, shadowing effect decreases and the possibility of having line-of-sight (LoS) connections between ground users and UAVs increases. Therefore, an optimum altitude for the aerial base station which results in a maximum coverage exists. Assuming only one UAV operats with no inter-cell interference, the optimum altitude for the UAV which leads to a maximum coverage is derived. However, the authors did not consider the case of multiple UAVs where beyond altitude, the distance between UAVs also impacts the overall coverage performance. The problem of multiple UAV deployment is much more challenging as the the distance between the UAV-BSs and their relative positions affects the overall coverage performance. Moreover, due to the presence of interference between the received signal from different UAVs, additional interference management/avoidance protocols are necessary.

We develop a novel approach for optimally deploying UAVs to provide wireless to service ground users while minimizing the overall UAV transmit power needed to satisfy the users' data rate. We consider multiple UAVs in the downlink scenario and derive, jointly, the optimal cell boundaries (coverage area) and locations of UAVs that minimize the required transmit power. To this end, we first fix the cell boundaries, and solve the facility location problem to determine the optimal locations of UAVs based on users' distribution. Next, given the prospective locations of UAVs, using optimal transport theory, a powerful mathematical framework from probability theory, we find the optimal cell boundaries for the UAVs.

3.2.4.4.1 System Model

In our model, we considered a geographical area divided into K subareas in which N wireless users are distributed based on an arbitrary distribution $f(x, y)$. This area must be

serviced by multiple UAVs that will act as flying base stations. Each subarea will be served by a single UAV located at (x_i, y_i, h_i) in the Cartesian coordinate where index i corresponds

to UAV i. Initially, we consider the subarea i over $[x_{s,i}, x_{s,i+1}] \times [y_{s,i}, y_{s,i+1}] \subset \mathbb{R}^2$. We consider a downlink scenario in which UAVs adopt a frequency division multiple access (FDMA) technique to transmit data to the ground users at a desired data rate. FDMA assigns individual frequency bands to users and each user has its own dedicated channel for communications. Without loss of generality, we assume that the total transmit power of UAVs and the total available bandwidth is sufficient to meet the users' rate requirement.

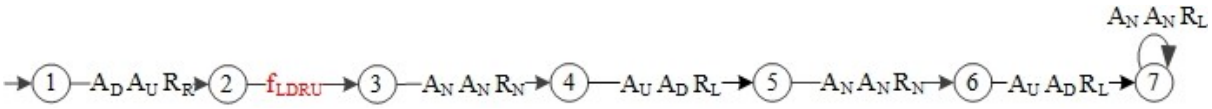


Figure 104: Flight commands for Left and Right Aileron simultaneously stuck down and stuck up, f_{LDRU}.

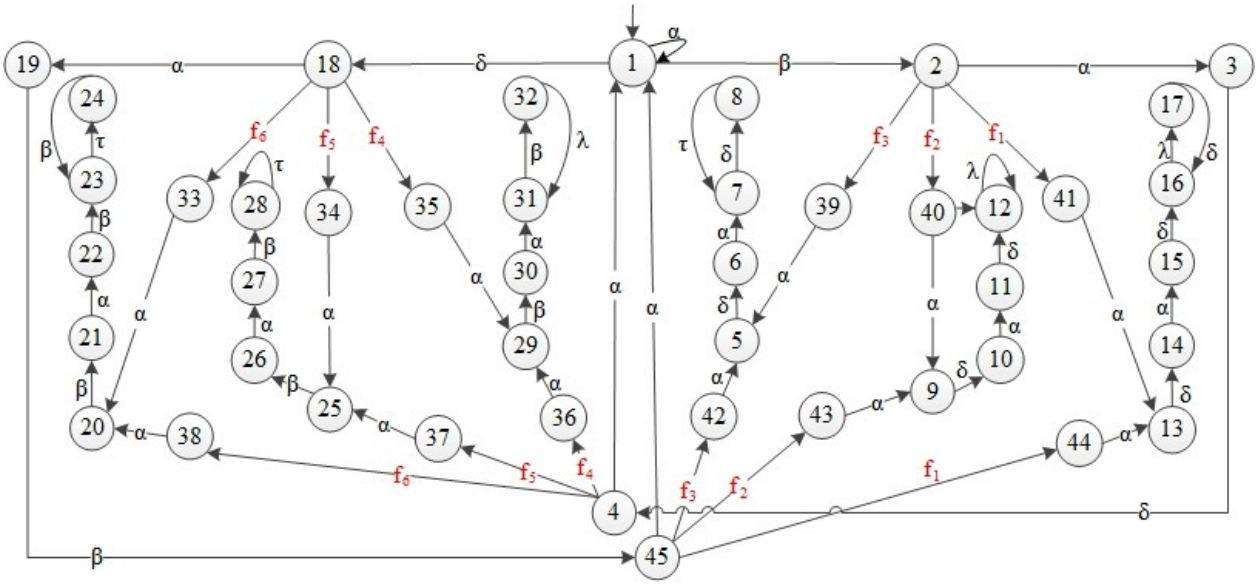


Figure 105: DES model of a Fixed Wing aircraft including faulty and normal behaviors during the bank turn.

In our model, the UAVs transmit over different frequency bands and hence, they do not interfere with one another. Moreover, hereinafter, we use the notion of a cell to indicate the coverage region of each UAV. In other words, each UAV is associated with a cell within which the ground users serviced by this UAV are located. Note that, at the initial setup, the cell boundary associated with each UAV is not optimal, and our goal is to optimize those such that the total transmit power is minimized. Next, we first provide the air-to-ground channel model and, then, present the problem formulation.

In order to analyze the wireless coverage of the UAVs, we first study the air-to-ground (ATG) channel propagation model. As discussed in previous works, the radio signal from a UAV base station reaches its destination in accordance to two main propagation groups. The first group corresponds to receiving a LoS signal while the second group corresponds to receiving a strong non-LoS (NLoS) signal due to reflections and diffractions. These groups can be considered separately with different probabilities of occurrence which depend on the environmental factors such as the density and height of buildings, and the elevation angle. Here, we adopt a model for characterizing the ATG channels for LAP systems. The free space pathloss (FSPL) is given by:

$$\text{FSPL} = 20\log\left(\frac{4\pi f_c d}{c}\right), \tag{14}$$

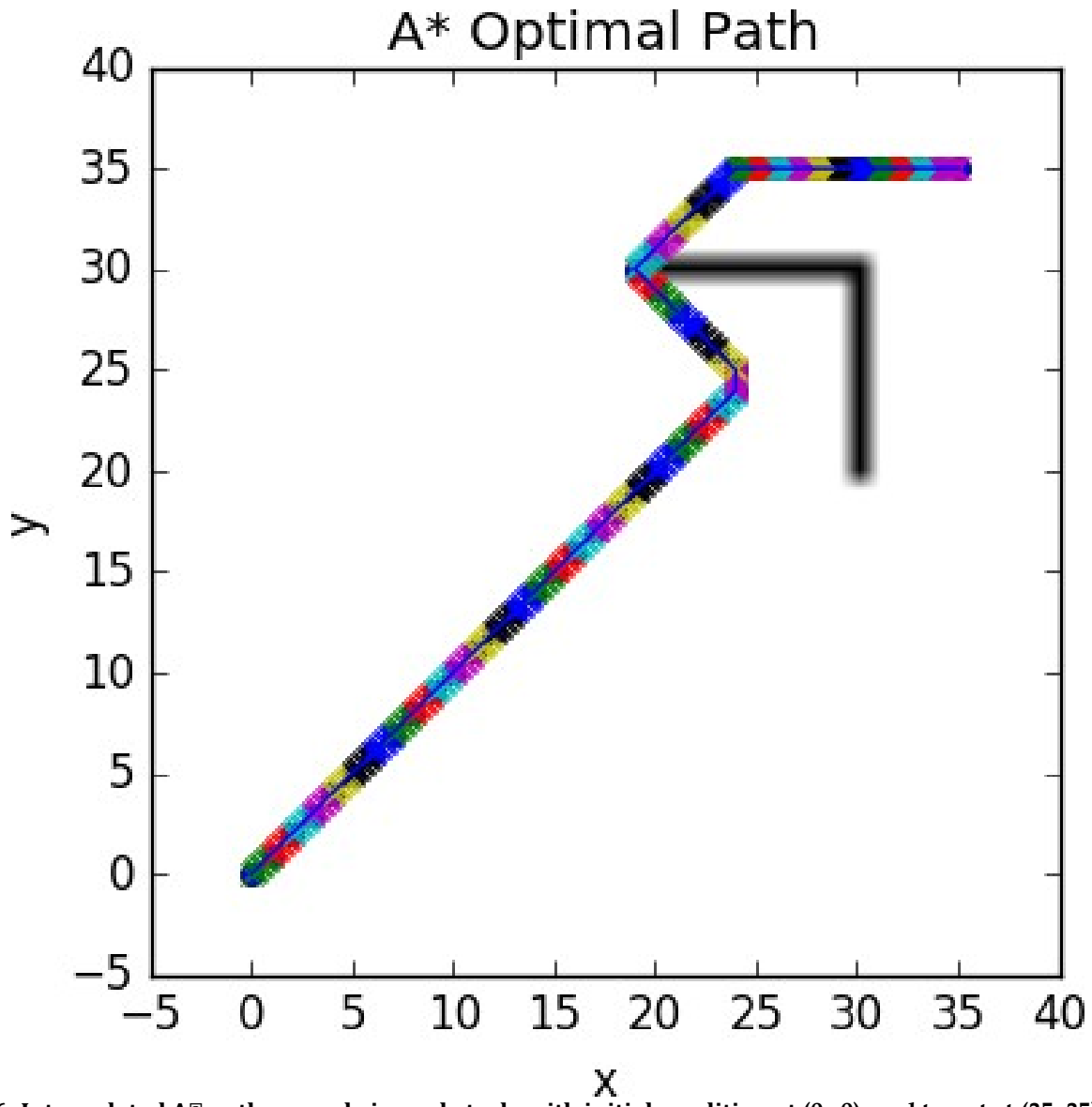


Figure 106: Interpolated A path around given obstacle with initial condition at (0, 0) and target at (35, 35). 2mx2m cost map that is segmented into a 40x40 grid. White: traversable, Black: obstacle.

in which f_c is the carrier frequency and c is the speed of light. In addition, $d = \sqrt{h^2 + r^2}$ is the distance between the UAV hovering at altitude h and the ground receiver located at radial distance r from the UAV in the 2D plane. Considering the mean value of LoS rather than its instantaneous

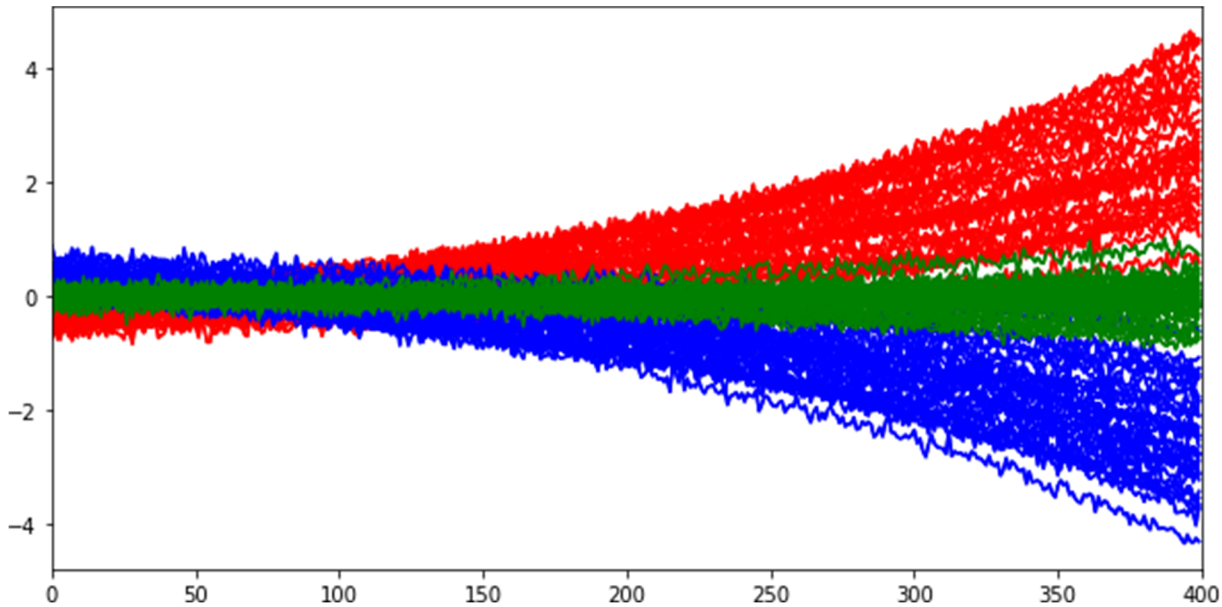


Figure 107: UGV trajectory subject to fault conditions, Green: No Fault, Red: Left Suspension Fault, Blue: Right Suspension Fault.

characteristics the mean pathloss model for ATG channel is therefore given by

$$PL(\text{dB}) = \text{FSPL}(\text{dB}) + \eta_\xi(\text{dB}), \tag{15}$$

where η_ξ represents the excessive pathloss due to shadowing and scattering in which the subscript ξ refers to the propagation group such that $\eta_\xi \in \{\eta_{LoS}, \eta_{NLoS}\}$. Each propagation group happens with a specific probability which depends on the environment. The values of η_{LoS} and η_{NLoS} should be found experimentally and η_{NLoS} is typically much larger than η_{LoS}. As the excessive pathloss, η_ξ, takes on two values, i.e., η_{LoS} and η_{NLoS} with probabilities $\mathbf{P}_{\text{LoS}}$ and $\mathbf{P}_{\text{NLoS}} = 1 - \mathbf{P}_{\text{LoS}}$, it can be modeled as a Bernoulli random variable with parameter $\mathbf{P}_{\text{LoS}}$,

$$\eta_\xi \sim \text{Bernoulli}\,(\mathbf{P}_{\text{LoS}}), \tag{16}$$

where $\mathbf{P}_{\text{LoS}}$ is the probability of having a LoS link and $\mathbf{P}_{\text{NLoS}} = 1 - \mathbf{P}_{\text{LoS}}$ is the probability of having a NLoS link. The probability of receiving LoS signal from UAV-BS U_i for a ground user located at (X, Y) depends on the altitude of the UAV-BS, h_i, and its horizontal distance to the user that is equal to $r_i = \sqrt{(X - x_i)^2 + (Y - y_i)^2}$, in which (x_i, y_i) is the location of the UAV at the 2D plane. The LoS probability is given by [**?**]:

$$\mathbf{P}_{\text{LoS}}(h_i, r_i) = \frac{1}{1 + \alpha \exp\left(-\beta\left(\arctan\left(\frac{h_i}{r_i}\right) - \alpha\right)\right)}, \tag{17}$$

in which α and β are constant values which depend on the environment. Since we cannot determine whether the link is LoS or NLoS as a priori, we consider the spatial expectation of the pathloss

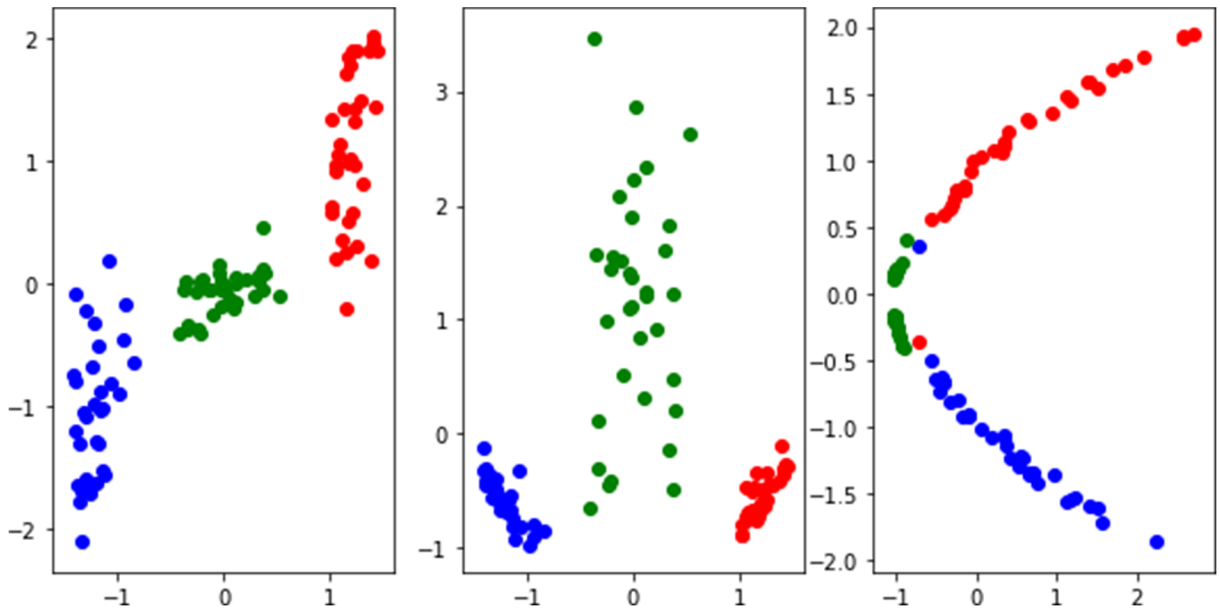


Figure 108: Plots of summary statistics, Left: (skewness, mean) Center: (skewness, kurtosis) Right:(variance, furthest value from 0).

over LoS and NLoS links,

$$\overline{PL}(\text{dB}) = \text{FSPL}(\text{dB}) + \eta_{\text{LoS}}(\text{dB})\mathbf{P}_{\text{LoS}} + \eta_{\text{NLoS}}(\text{dB})\mathbf{P}_{\text{NLoS}}. \tag{18}$$

By substituting (14) and (17) into (18), and letting $d_i = \sqrt{h_i^2 + r_i^2}$ to be the distance between the UAV U_i and the user, we have

$$\overline{PL}(\text{dB}) = 20\log(d_i) + \frac{A}{1 + \alpha\exp(-\beta(\theta - \alpha))} + B, \tag{19}$$

in which $A = \eta_{\text{LoS}}(\text{dB}) - \eta_{\text{NLoS}}(\text{dB})$ and $B = \eta_{\text{NLoS}}(\text{dB}) + 20\log(\frac{4\pi f c}{c})$.

Next, given a UAV base station with transmit power P_i^t, we find its optimal flight altitude h_i, which maximizes the size of the covered area. We define the service threshold in terms of the minimum allowable received signal power for a successful transmission. Having defined the expected pathloss in (19), the received signal power at a ground receiver located in radial distance r_i from the ground image of the UAV is given by

$$P^r(\text{dB}) = P_i^t(\text{dB}) - \overline{PL}(\text{dB}), \tag{20}$$

which requires to be greater than E, i.e., $P^r \geq E$. This is equivalent to having

$$\overline{PL}(\text{dB}) \leq P_i^t(\text{dB}) - E. \tag{21}$$

We define the coverage radius for a UAV-BS U_i with transmit power P_i^t as the radial distance in which the received signal power on a ground receiver reaches the threshold E, i.e.,

$$R_i < r_{i\ \overline{PL}(\text{dB}) = P_i^t(\text{dB}) - E}, \tag{22}$$

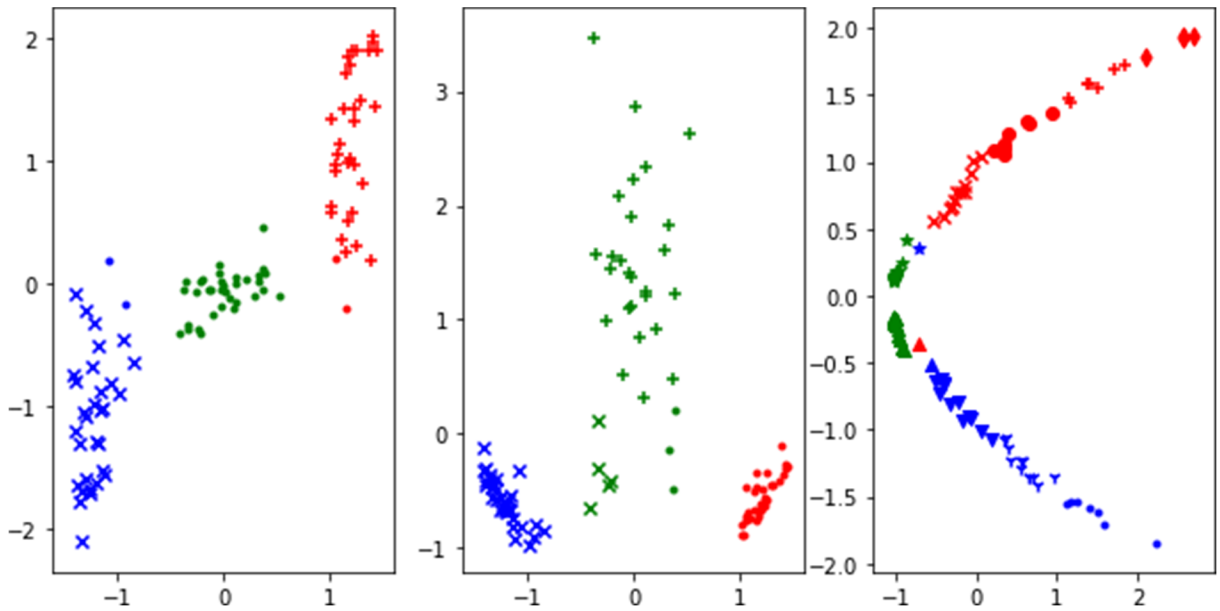


Figure 109: Plots of spectral clustering results on 1: (skewness, mean), 2: (skewness, kurtosis), 3:(variance, furthest value from 0).

in which R_i is the coverage radius of UAV-BS U_i. Using (19), the above condition can be re-written as:

$$20\log(d_i) + \frac{A}{1 + \alpha\exp(-\beta(\arctan\left(\frac{h_i}{R_i}\right) - \alpha))} + B + E - P_i^t = 0, \tag{23}$$

where $d_i = \sqrt{h_i^2 + R_i^2}$. The equation in (23) shows that R_i is an implicit function of h_i. However, as it is shown in (23) is a unimodal function and has only one stationary point which corresponds to the maximum coverage radius. Let h_i^* denote the optimal flight altitude which results in the maximum coverage radius. We find h_i^* by taking partial derivative from the expression in (23) as:

$$\frac{\partial R_i}{\partial h_i} = 0, \tag{24}$$

which yields the following equation:

$$\frac{h_i}{R_i} + \frac{9\ln(10)\alpha\beta A\exp\left(-\beta[\arctan\left(\frac{h_i}{R_i}\right) - \alpha]\right)}{\pi\left[\alpha\exp\left(-\beta[\arctan\left(\frac{h_i}{R_i}\right) - \alpha]\right) + 1\right]^2} = 0. \tag{25}$$

Numerically solving the equations in (23) and (25) gives us the optimal flight altitude h_i^* and the corresponding coverage radius R_i of the UAV-BS $U_i \in \mathcal{U}$ as a function of its transmit power P_i^t and environmental parameters.

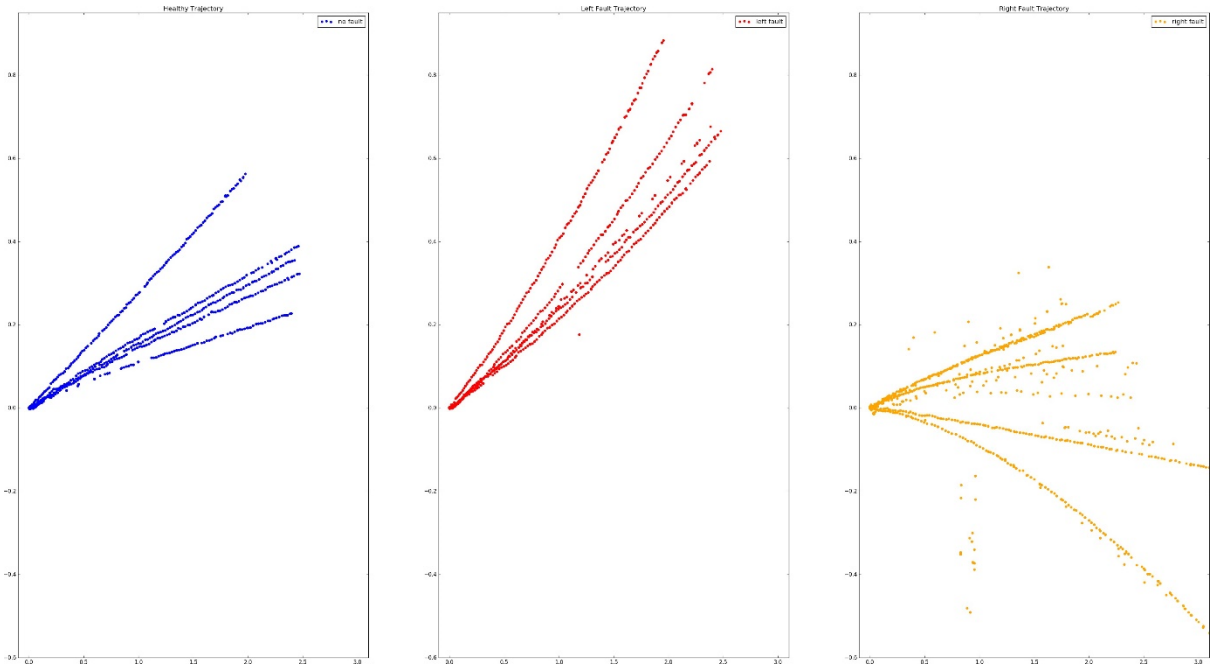


Figure 110: UGV trajectory subject to fault conditions, Left: No Fault, Center: Left Suspension Fault, Right: Right Suspension Fault

3.2.4.4.2 Problem Formulation Consider the transmission between UAV i and a ground user located at (x, y) coordinates. The achievable rate for the user is given by:

$$R_i(x,y) = W_i \log_2 \left(1 + \frac{P_i(x,y)/\bar{L}_i(x,y)}{N_0}\right), \tag{26}$$

where W_i is the transmission bandwidth of UAV i,$P_i(x,y)$ is the UAV transmit power to the user, $L_i(x,y)$ is the average path loss between UAV i and the user, and $N_0$is the noise power. Considering B_i as the total available bandwidth at UAV i, and M_i as the number of users serviced by UAV i, we have $W_i = \frac{B_i}{M_i}$ Note that, M_i is the number of users inside cell boundary of UAV i which is computed as $M_i = N \int\int_{C_i} f(x,y)\,dxdy$, with N being the total number of users, and C_i being the cell boundary corresponding to UAV i. Clearly, the number of users covered by the UAV depends on the distribution of users, cell boundary, and the location of the UAV. The minimum transmit power required to satisfy the rate requirement β of ground users is given by:

$$P_{i,\min}(x,y) = (2^{\beta W_i} - 1) N_0 \bar{L}_i(x, y), \tag{27}$$

.

Note that, as the number of users increases, the bandwidth per user decreases. Consequently, a higher transmit power is required to meet the users rate requirement. Given the location of UAVs the average total transmit power of the UAVs in the network is given by:

$$P_t = \sum_{i=1}^{K} \int\int_{C_i} P_{i,\min}(x,y) f(x,y)\,dxdy. \tag{28}$$

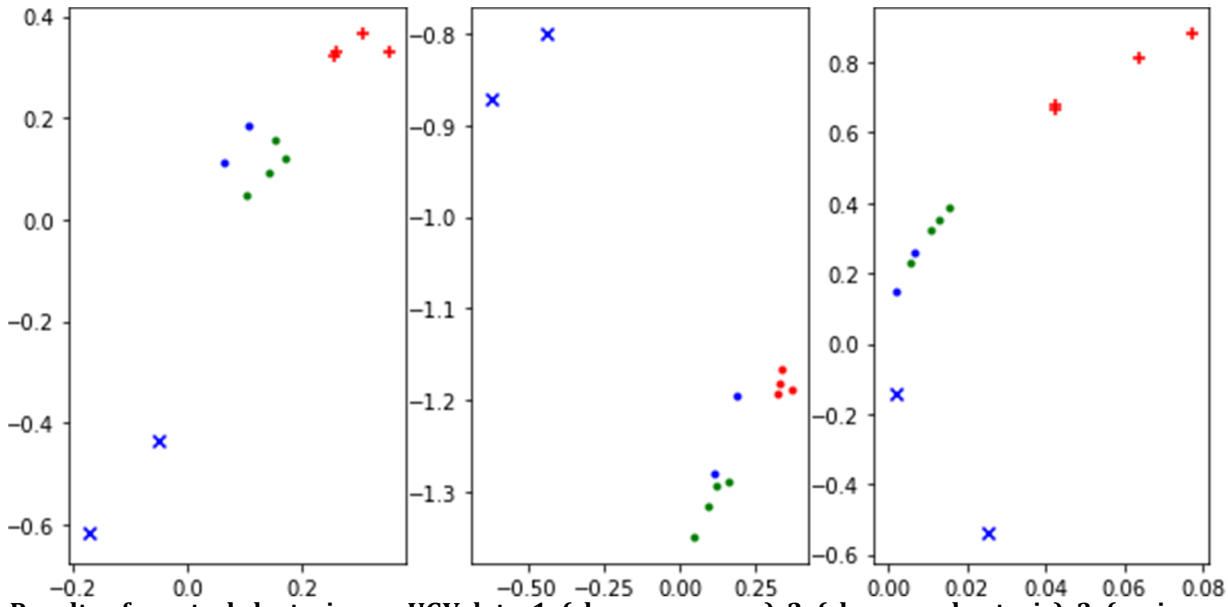


Figure 111: Results of spectral clustering on UGV data, 1: (skewness,mean), 2: (skewness,kurtosis), 3: (variance,furthest value from 0)

Our goal is to minimize the total required transmit power by finding jointly the optimal locations of the UAVs and their associated cell boundaries. Therefore, the power minimization problem can be formulated as follows:

$$\min_{C_i, x_i, y_i, h_i} P_t = \sum_{i=1}^{K} \int_{C_i} (2^{\beta M_i B_i} - 1) N_0 \bar{L}_i(x, y) \mathrm{d}x \mathrm{d}y, \tag{29}$$

where $i \in \{1, 2, \ldots, K\}$, C_i is the cell boundary that shows the coverage region of the UAV, and $x_i, y_i, h_i)$ is the location of UAV i.

The solution for this problem provides optimum cell boundaries and UAVs location such that the total average transmit power of UAVs is minimized while the rate requirement for all the users is maintained. However, solving this optimization is challenging due to the mutual dependency of optimization parameters. Furthermore, the problem must be solved over a continuous space while considering an infinite number of possible UAVs' locations and cell boundaries.

3.2.4.4.3 Optimal UAVs Location and Cell Boundaries In order to solve the opti-mization problem in 29, we separate the problem into two optimization problems and solve them sequentially. In the first problem, we assume that the cell boundaries for UAVs are given and the objective is to determine the optimal location of UAVs for which the transmit power is minimized. In the second problem, for the given locations of UAVs we derive the optimal cell association which lead to the minimum total required transmit power.

Consider a scenario in which the UAVs move and change their locations based on the users distribution. In order to minimize the total transmit power, given the cell boundaries, we find the

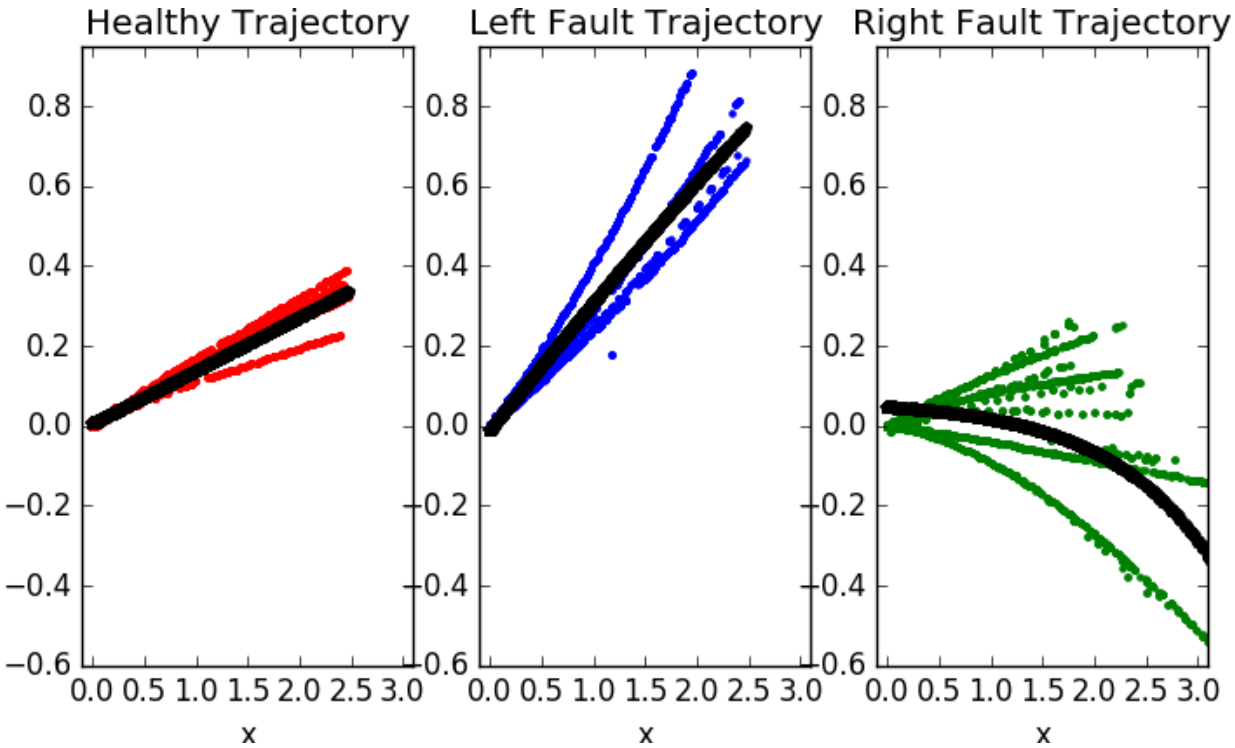


Figure 112: Optimized Fault Noise Functions, Left: No Fault, Center: Left Suspension Fault, Right: Right Suspension Fault

optimal location of each UAVs in its corresponding subarea. Given the distribution of users over the geographical area, the total transmit power of UAVs is given by $P_t = \sum_{i=1}^{K} P_i$, where P_i is the average transmit power of UAVs i over its corresponding subarea which is given by:

$$P_i = \int_{y_k}^{y_{k+1}} \int_{x_k}^{x_{k+1}} (2^{\beta W_i} - 1) N_0 \bar{L}_i(x, y) f(x, y) \mathrm{d}x \mathrm{d}y. \tag{30}$$

Note that, for a given cell boundary, the number of users inside the cell is fixed and optimizing P_t and P_i will not depend on M_i. Furthermore, in this case, minimizing P_t is equivalent to minimizing P_i for all $i \in \{1, 2, \ldots, K\}$. Therefore, the optimization problem in 29 can be written as:

$$\min_{x_i, y_i, h_i} P_i = \int_{y_{s,i}}^{y_{s,i+1}} \int_{x_{s,i}}^{x_{s,i+1}} ((x - x_i)^2 + (y - y_i)^2 + h_i^2) \tag{31}$$

$$\times (P_{\mathrm{LOS}} + (1 - P_{\mathrm{LOS}})) f(x, y) \mathrm{d}x \mathrm{d}y, \tag{32}$$

Now, we derive a closed-form expression for the optimal location of the UAVs when they are deployed at high or low altitudes relative to the size of the subareas.

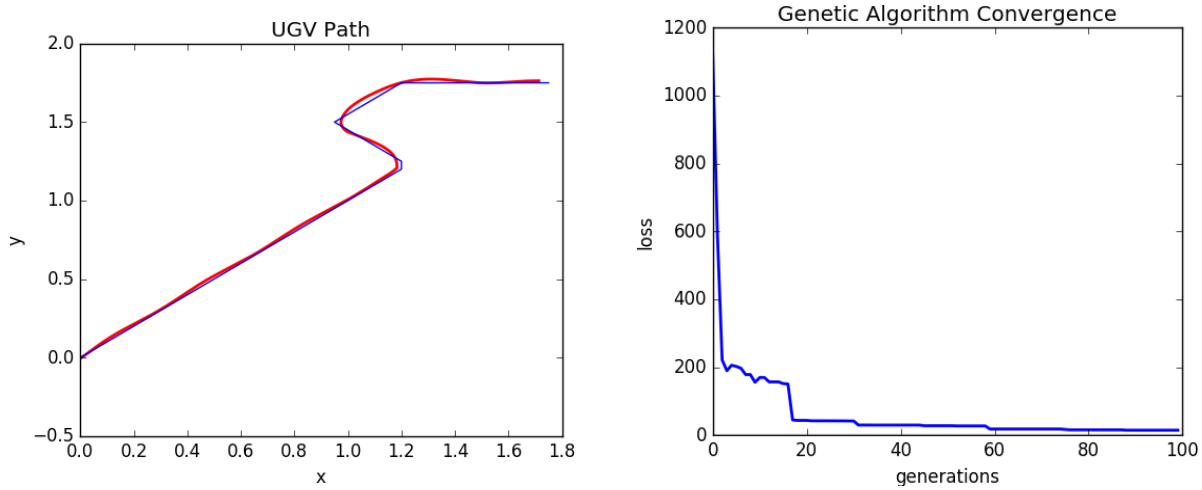


Figure 113: Left: Simulated path of UGV under healthy conditions using optimized PID controller. Blue: Desired Path, Red: Simulated Path, Right: Genetic algorithm convergence plot under healthy conditions. Converges to 14.39

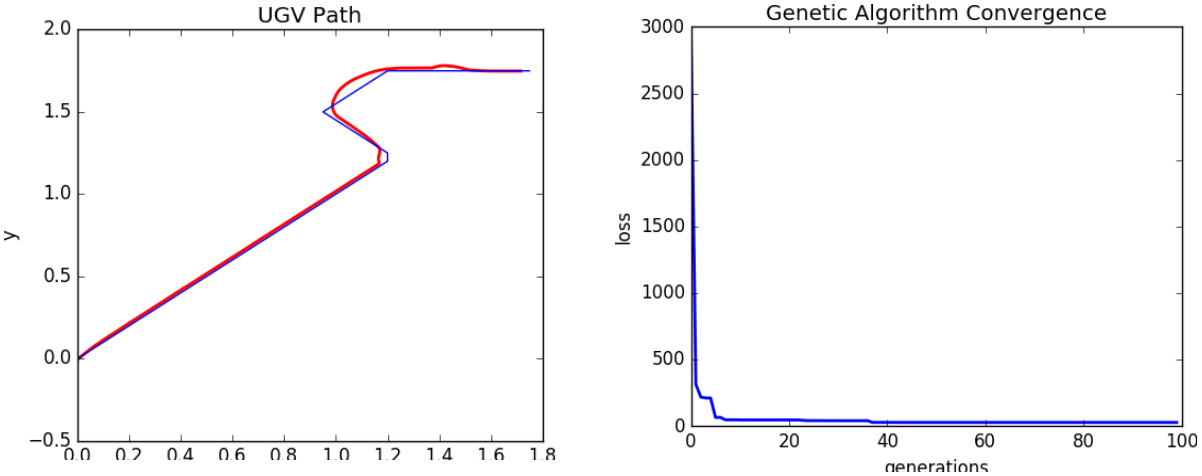


Figure 114: Left: Simulated path of UGV with the left suspension disabled using optimized PID controller. Blue: Desired Path, Red: Simulated Path, Right: Genetic algorithm convergence plot with the left suspension disabled. Converges to 33.56

Theorem 1: Seeking a minimum required transmit power, the optimal location of UAVs *i*

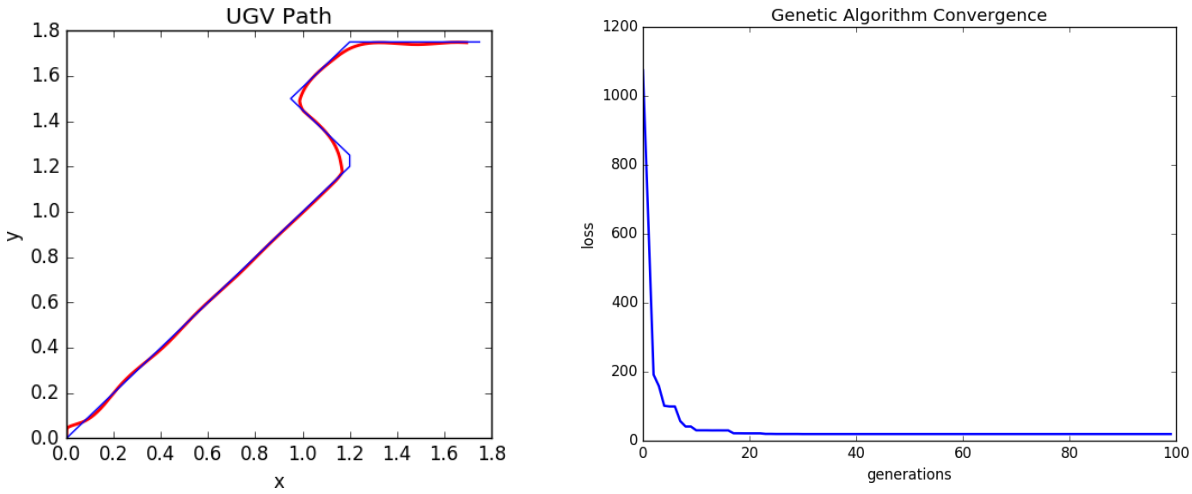


Figure 115: Left: Simulated path of UGV with the right suspension disabled using optimized PID controller. Blue: Desired Path, Red: Simulated Path, Right: Genetic algorithm convergence plot with the right suspension disabled. Converges to 21.37

positioned at high or low altitudes compared to the size of its corresponding subarea is given by:

$$x_i^* = \frac{\int_{y_k}^{y_{k+1}} \int_{x_k}^{x_{k+1}} x f(x,y)\,dx\,dy}{\int_{y_k}^{y_{k+1}} \int_{x_k}^{x_{k+1}} f(x,y)\,dx\,dy}, \tag{33}$$

$$y_i^* = \frac{\int_{y_k}^{y_{k+1}} \int_{x_k}^{x_{k+1}} y f(x,y)\,dx\,dy}{\int_{y_k}^{y_{k+1}} \int_{x_k}^{x_{k+1}} f(x,y)\,dx\,dy}, \tag{34}$$

where $h_i^2 \gg (x - x_i)^2 + (y - y_i)^2$ or $h_i^2 \ll (x - x_i)^2 + (y - y_i)^2$. Note that, considering $f(x, y)$ as the distribution of users over the geographical area, (x_i^*, y_i^*) corresponds to the centroid of the area.

P roof: At very high altitudes, i.e., $h^2 \gg (x - x_i)^2 + (y - y_i)^2$, we have $d_i(x, y) \approx h_i$ and $\theta = \sin^{-1}\left(\frac{h}{d_i(x,y)}\right) \approx \frac{\pi}{2}$ and therefore, $P_{\text{LOS}} \approx 1$. Then we have $L_i(x, y) \approx \left((x - x_i)^2 + (y - y_i)^2 + h_i^2\right)$. Similarly, for very low altitudes we calculate $L_i(x, y) \approx \eta((x - x_i)^2 + (y - y_i)^2 + h_i^2)$. In both cases, we have:

$$\frac{\partial P_i}{\partial x_i} = \frac{\partial \int_{y_k}^{y_{k+1}} \int_{x_k}^{x_{k+1}} \bar{L}(x, y) f(x, y) \mathrm{d}x \mathrm{d}y}{\partial x_i} \tag{35}$$

$$\int_{y_k}^{y_{k+1}} \int_{x_k}^{x_{k+1}} \frac{\partial \bar{L}(x, y)}{\partial x_i} f(x, y) \mathrm{d}x \mathrm{d}y = 0. \tag{36}$$

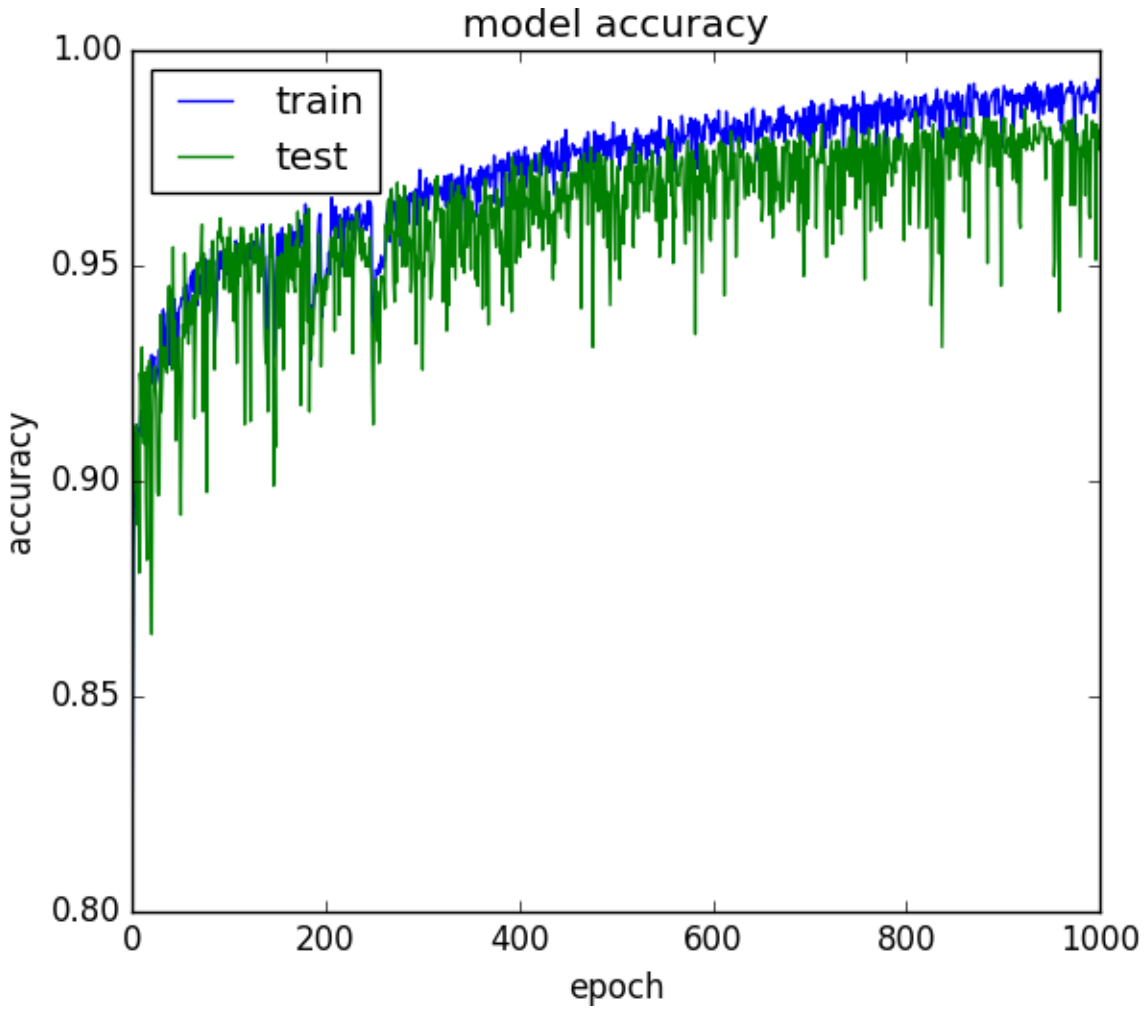


Figure 116: Accuracy convergence plot for LSTM given 1900 epochs, training accuracy: 99.08%, testing accuracy: 98.13%

and,

$$\frac{\partial P_i}{\partial y_i} = \int_{y_k}^{y_{k+1}} \int_{x_k}^{x_{k+1}} \frac{\partial \bar{L}(x,y)}{\partial y_i} f(x,y)\,dx\,dy = 0. \tag{37}$$

Finally, solving these two equations, we find x_i^* and y_i^*.

3.2.4.4.4 Numerical Results Assuming that UAVs are operating in urban environments, numerical results are presented. The operating frequency is $f_c = 2GHz$ and the size of the area is $1000m \times 500m$. The area is divided into two equal subareas and contains two UAVs. We also consider a hotspot area in which users are distributed according to a truncated Gaussian

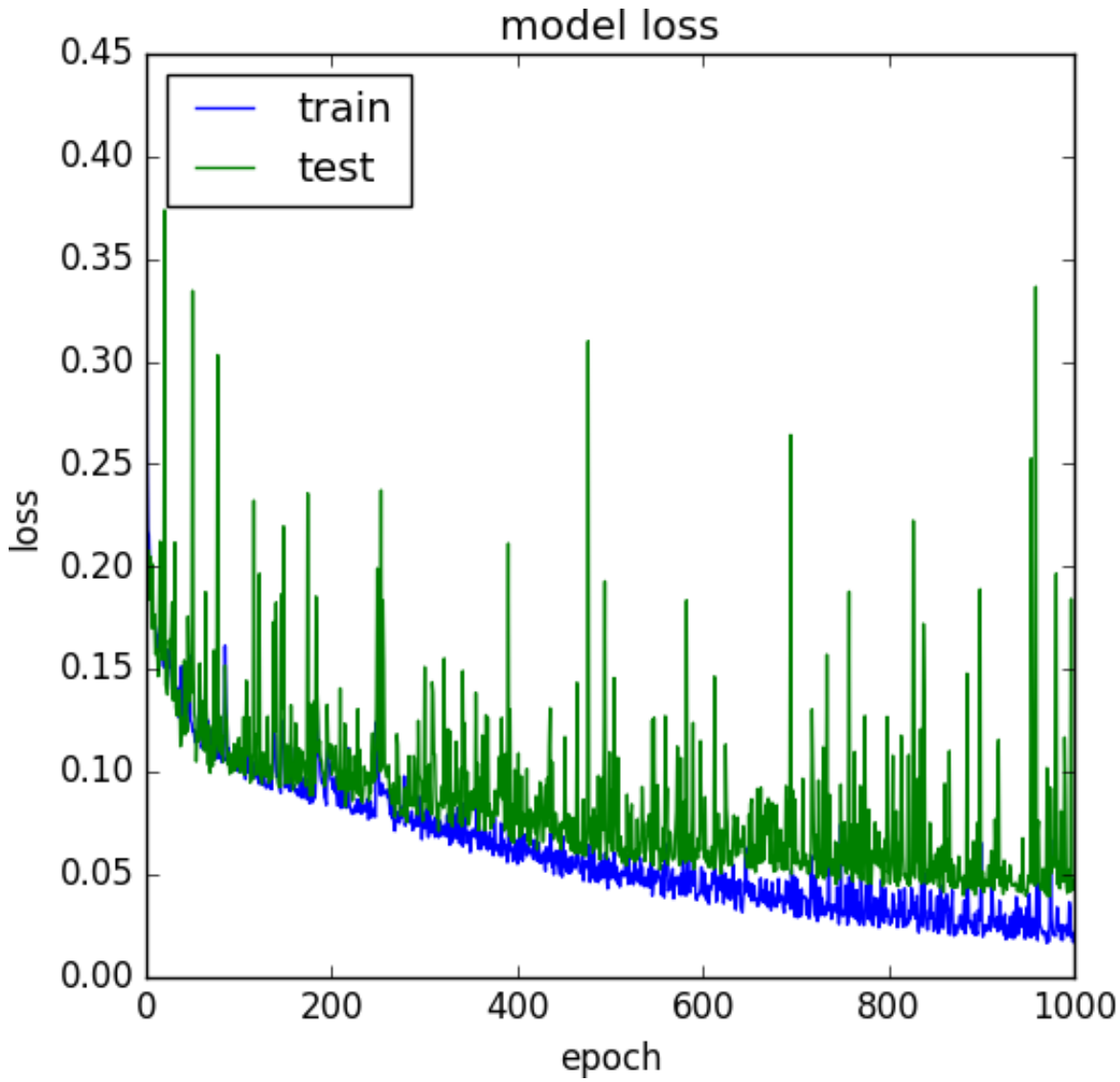


Figure 117: Loss convergence plot for LSTM given 1000 epochs, training loss: 0.0197, testing accu-racy: 0.0474

distributions. Figure 119 shows the average transmit power of UAVs versus the density of the users for the optimal cell boundaries and the Voronoi cell boundaries. Note that, in Figure 119, we assume that the UAVs are located at the center of the subareas, and their altitude is 200 m. As we can see, the average transmit power for optimal cell boundaries is significantly lower than the Voronoi case. According to Figure 119, the average transmit power is around 0.3 W and 0.12 W, respectively, for the Voronoi and the proposed optimal cell boundary cases. Furthermore, the Voronoi case is more

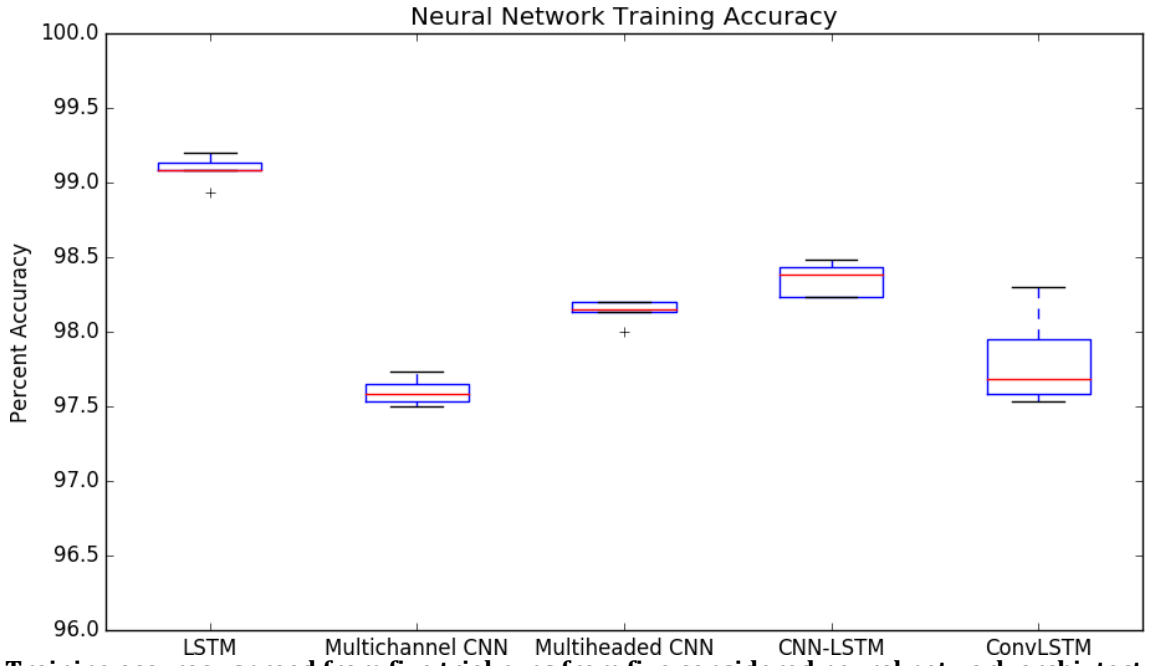


Figure 118: Training accuracy spread from five trial runs from five considered neural network archi- tectures.

sensitive to the users' density compared to the optimal cell boundaries. This is due to the fact that the optimal cell boundaries are determined based on the users' density such that the transmit power is minimized. However, in the Voronoi case, the cell boundaries are set without considering the users density. As observed in Figure 119, for the low user density case in which the users are more spread over the area, the performance of Voronoi and optimal cell boundaries are close. However, as the density increases, the proposed optimal case becomes better but then they get close again. The reason is that, for a very highly dense scenario, most users are located around the hotspot center and they are served by the closest UAV. As a result, the average channel gain is high for the users and thus, power efficiency for the Voronoi case is improved. As we see from Figure 119, for $\rho \approx 0.02$, the proposed approach yields a maximum power improvement over the Voronoi case.

Figure 120 shows the impact of UAV altitude on the average transmit power for optimal cell boundaries with $\rho = 0.01$. In our setup, UAV 1 is closer to the hotspot center than UAV 2. Figure 120 shows that, the total average transmit power is minimum at an altitude of 400 m. In fact, the UAVs should not be positioned at very low altitudes, due to high shadowing and a low probability of LOS connections towards the users. On the other hand, at very high altitudes, LOS links exist with a high probability but the large distance between UAV and users results in a high path loss. As shown in Figure 120, the optimal individual altitude for UAV 1 and UAV 2 are around 320m and 500m, respectively. However, the total transmit power of both UAVs is minimized for $h_1 = h_2 = 400m$.

Figure 121 illustrates the inverse of the total average transmit power as a function of altitude for the optimal cell association case. Note that, in this figure, we used inverse of power solely for a better illustration of the results and for clarity of the figure. Here, we consider all the possible

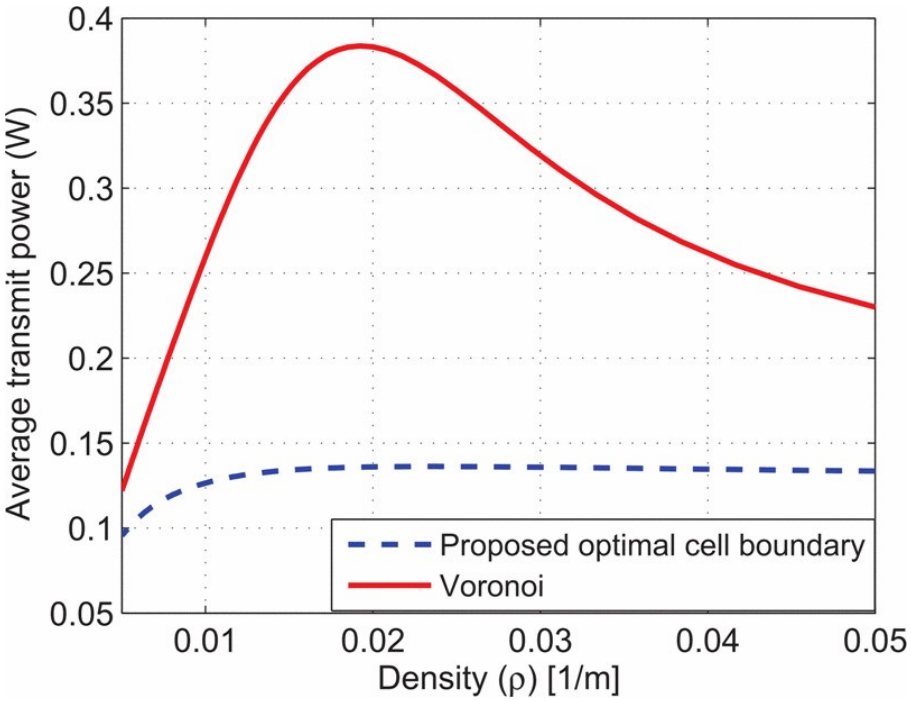


Figure 119: Average required transmit power versus users' density.

combinations of and from 200 m to 1200 m. As seen from Figure 121, the minimum total average transmit power (maximum inverse of power) is about 0.12 W and it is achieved for h1=310 m and h2=530 m. Note that, since the hotspot center is closer to UAV 1, on the average, this UAV has a higher chance of LOS links to users compared to UAV 2. Hence, UAV 2 should be at a higher altitude in order to improve its channel condition (more LOS links) to the users.

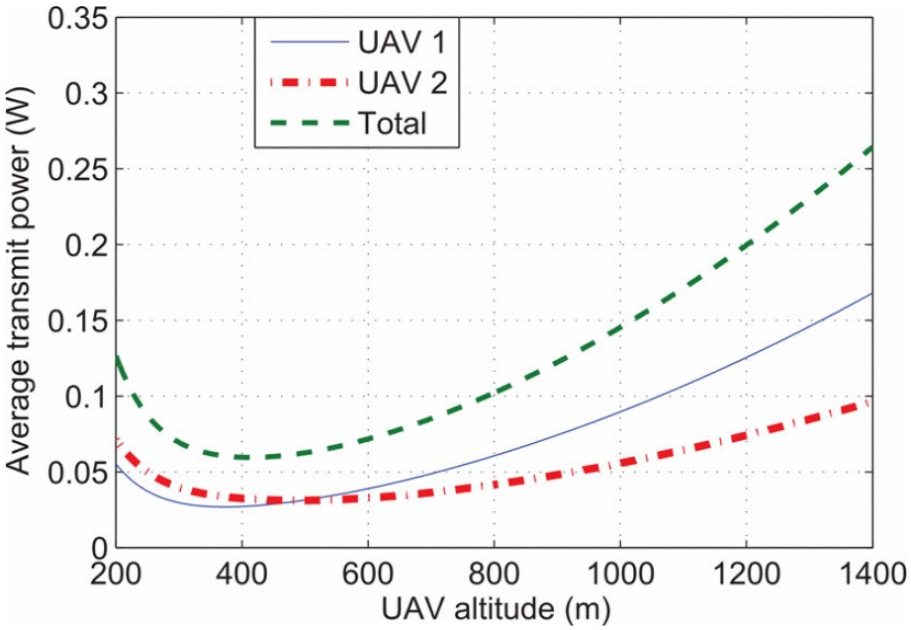


Figure 120: Average required transmit power versus UAVs altitude.

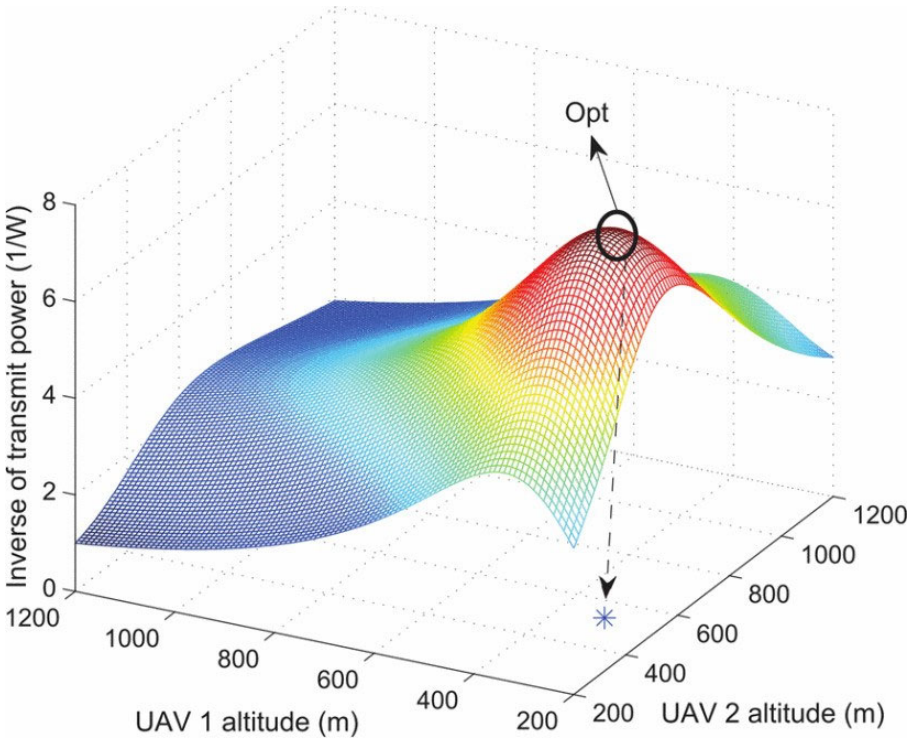


Figure 121: Inverse of average transmit power versus altitude.

3.2.5 Project Progress for Task T2-5 (Enhancing network performance utilizing the cognitive features of the agents)

3.2.5.1 Period of Performance under

Task T2-5 Start Date: Fall 2016

Conclusion Date: Spring 2021

Faculty lead: Dr. Abdollah Homaifar, Dr. Brian Kelley

3.2.5.2 General Description of Task T2-5

Due to recent advances in electronics and communications, the agents are currently equipped with cognitive radio devices that significantly improve the performance of the communication systems. This cognitive capability allows the users to sense the environment and monitor the operation of other agents to take proper responses for the observed information. The intelligence capability of the agents provides the possibility of independent decision-making regarding their communications parameters including the transmission power, transmission frequency, bandwidth, and utilized coding and modulation technique. We will take advantage of the agents' cognition toward planning an efficient and reliable user-centric communication system, where most communication design parameters are determined in a decentralized fashion. This involves several design paradigms including power allocation, utilized frequency and also channel access with the goal of interference reduction in the network, taking into account the communication strategies of other agents. We will adopt a game-theoretic framework in which the rational users set their communication parameters in a way to improve their individual benefits while taking into account the social welfare. Game theory is a powerful mathematical tool to analyze the interactions among the intelligent users when they have conflicting interests or they coordinate in performing a common task.

3.2.5.3 Objectives for Task T2-5 over the Course of Project

- **Objective Name:** Enabling D2D Communication in mmWave Band

  Objective Type: Literature review- network design- performance evaluation

  Objective Description: In order to enable D2D communication in the mmWave band we were required to perform a thorough literature review to identify the-state-of-art and gaps. Then, we proposed solution for the identified gaps. Finally, we evaluated the performance of the proposed solutions analytically and by extensive simulations.

  Impact: D2D and mmWave are two key technologies in 5G wireless communication that facilitate infrastructure-less high-rate communication for natural disaster situations and battlefields.

- **Objective Name:** Optimizing Communication Network Performance Using Machine Learning (ML) approaches

  Objective Type: Literature review- algorithm design- performance analysis

  Objective Description: In order to apply ML approaches to optimize network performance, we survey the literature to identify the possible solutions. We proposed an algorithm

based on deep reinforcement learning. Finally, we evaluated the performance of the proposed algorithm through extensive simulations.

Impact: ML approaches play an important role in optimizing wireless networks with low overhead and complexity.

3.2.5.4 Progress Against Planned Objectives in Task T2-5

D2D communication is envisioned to improve the 5G networks' capacity and spectral efficiencies. Exploiting the mmWave band is an attractive solution to accommodate the bandwidth-intensive applications in D2D communications. In this dissertation, we investigated the benefits of D2D communication and enabling D2D communication in the mmWave band. Furthermore, we identified challenges involved in establishing a reliable D2D communication framework in the mmWave band. We address some of these main issues in the mmWave and μWave band such as resource allocation, blockage susceptibility, erroneous antenna beam alignment, peer association and antenna beamwidth optimization. In order to solve these problems, we utilized tools from stochastic geometry, game theory and machine learning. In the following subsection, we summarize the problems investigated in Task 2-5 and review the solutions proposed to address those problems.

3.2.5.5 Technical Accomplishments in Task T2-5

Describe the technical accomplishments made during the course of the project.

We proposed a novel distributed mechanism which enables D2D devices to select between the mmWave band and the μW band for data transmission. In order to devise a distributed mmWave band communication protocol for D2D communications, the D2D users are in charge of detecting LOS links along with their corresponding direction to perform proper beam alignment. Our proposed algorithm enables the D2D devices to perform such a task by using peer-discovery beacons and comparing the AoA spectrum of their intended peer over subsequent time intervals. We have used stochastic geometry to provide a complete framework to analyze the performance of the proposed mechanism in terms of the received SINR coverage probability of D2D users for which closed-form analytical formulas are derived. Our simulation results demonstrate that the proposed mechanism achieves considerable performance gain over the single band (i.e., mmWave/μW) D2D communications. Moreover, our simulations validate the analytical results discussed in the chapter.

We proposed a mathematical framework to analyze the impact of AoA estimation on the performance of a mmWave D2D network. Based on the prior information we have about the error, the AoA estimation error is modeled using normal and uniform distributions. We have used stochastic geometry to provide a complete framework to analyze the D2D network performance in the presence of error in terms of the received SINR coverage probability, for which analytical formulas are derived. Simulation results show that the coverage of the network with erroneous beam alignment can be degraded by about 35% compared to the one with perfect beam alignment. Moreover, our simulations validate the analytical results discussed in the chapter. Considering the significant impact of beam alignment error on the network performance, proposing a mechanism that corrects and compensate the beam alignment error using a feedback loop is a promising future direction.

We proposed a novel D2D peer selection approach which exploits the context information about the users' velocity and size of their demanded data to assign a link to the D2D pairs. We derived a closed-form formula for the probability of link-consistency which has a significant impact on the

QoS of D2D links. We formulated the problem as a matching game and proposed a novel distributed algorithm to reach a stable matching between the set of D2D users and available links. Simulation results show that the proposed algorithm yields considerable gains in terms of traffic offloading from the cellular network and the average user throughput compared to that of the random partner assignment scenario for the D2D users.

We proposed a CCN-based D2D framework in the mmWave band along with a decentralized peer association algorithm to enhance data throughput and offload the cellular network. The proposed algorithm enables D2D requesters to find D2D transmitters that are cached with their desired data and select the DT that provides the most stable link for data transmission. The proposed algorithm uses context information, including D2D users trajectory and size of the requested data, as well as mmWave characteristics such as susceptibility to blockage and directional communication. Extensive simulations are performed to evaluate the performance of the algorithm. Simulation results show that the proposed algorithm, compared to other baseline algorithms, improves network data throughput significantly while offloading the cellular network.

A novel decentralized scheme is proposed to enable D2D users to perform the initialization process in a CCN-based mmWave D2D network. The proposed scheme comprises of two phases, namely, heuristic peer association algorithm and synchronous beamwidth selection algorithm. The context-aware peer association algorithm is low-overhead with a low computational load and enables peer association in a decentralized manner. Following the peer association, antenna beamwidth optimization is performed considering the trade-off between antenna beamwidth and data throughput in directional communication. A synchronous LLL-based algorithm is proposed to obtain the joint beamwidth selection strategy of all users to maximize the network data throughput. the performance of the proposed scheme is evaluated through extensive Monte Carlo simulations. Simulation results show that the proposed initialization scheme significantly improves the network performance compared to other methods in the literature.

Finally, we proposed a novel multi-agent DRL-based algorithm to optimize D2D UEs' antenna beamwidth in a directional D2D network in the mmWave band. The proposed algorithm considers D2D UEs' mobility, payload size, QoS requirements, beam alignment cost and non-stationarity of the environment. The proposed algorithm enables D2D links to learn an optimized antenna beamwidth policy to increase the network sum-throughput while maintaining the D2D link reliability. D2D links are trained offline using a shared reward function while the implementation is distributed and does not require any online coordination. The training algorithm is based on the multi-agent DRL, and the non-stationarity of the environment is addressed by augmenting users' observation with a low dimensional fingerprint. Finally, the performance of the proposed antenna beamwidth optimization algorithm is evaluated through extensive simulations. Also, a performance comparison is performed with existing approaches, such as IQL and random beamwidth selection. Results show that our proposed algorithm improves network performance significantly and outperforms other approaches.

3.2.6 Project Progress for Task T2-6 (Delay-tolerant and loss-tolerant consensus in networks of agents)

3.2.6.1 Period of Performance under

Task T2-6 Start Date: May 2015

Conclusion Date: April 2021

Faculty lead: Dr. Sun Yi

3.2.6.2 General Description of Task T2-6

Communications among agents often experience substantial delays, packet dropouts, and intermittent connectivity and link breakage. The effects of these impairments on system performance are not trivial, but also not predictable. Development of an effective solution to this is still an open-ended problem. The dynamical behavior of each agent is dependent on received information. Thus, the control design taking into account communication disruptions and delays is important. Controllers in such networks should work robustly against changes in the network topology. In order to address these issues, we will perform stability analysis for networked multi-agent systems. The research aims at developing a theoretical framework to establish stability criteria for multi-agent systems considering network topology. Randomly changing network topology as well as fixed topology will be considered. For example, connection between the robustness margin to time delays and eigenvalues of network topology will be used to investigate convergence properties of consensus protocol. Also, stability in case of communication channel collapse will be analyzed, for example, through Lyapunov equations. In our recent work, we have designed controllers to improve system performance even in the presence of uncertainty by breaking the barriers of infinite-dimensionality using the Lambert W function. We will further study this problem, by considering the effect of both time delay and uncertainty on networked multi-agent systems to yield better understanding and advanced control strategies.

3.2.6.3 Technical Accomplishments in Task T2-6

First, the consensus problem and the system's stability in the presence of communication delay was studied. The analytical solutions are derived in terms of the Lambert W function. Then, the stability of the systems is determined from the characteristic roots (or eigenvalues). Also, using the derivative of the function sensitivity with respect to changes in the delay are quantified and analyzed. By simulations alone, one could conclude some systems consisted of agents with response speeds much faster than those in systems in other communication topologies. Fast system responses are ideal but not practical in reality for most systems. The time delay from .1 seconds to 1 second caused each system to become unstable. Fast systems became unstable for shorter time delays compared to Systems in other communication topologies. It was mentioned by Saber and Murray that there is a tradeoff between performance and robustness. Applying the Lambert W function to these simulated systems confirmed this approach is able determine the eigenvalues of the system. Although the Lambert W function-based approach was applied to only four topologies, it can be applied to general topologies for quantitative study of stability and sensitivity. This is the unique and novel contribution of the presented work. Compared to existing methods, the presented approach make it possible to analyze and quantify the stability, sensitivity, and convergence speed of delayed systems. These simulated systems were implemented in Simulink for real-time experiments. The Lambert W function assisted in the stability analysis of real systems. The testbed was created for the analysis of delay in real-time systems. Future research can implement the work presented here into the development of a control system for cooperative and coordinated control of MAS in the consensus problem. The rightmost eigenvalues could be used as a key point for eigenvalue assignment or pole placement within a control loop to ensure the system remains stable especially

while switching topology. This should guarantee an optimal performance/topology. This work can also be implemented on real network systems, such as drones, DC motors, and ground robots [23].

The focus of next problem was the consensus of MAS and the stability of the system in the presence of time delay and communication trust variations. Graph Theory and the Laplacian matrix are used to simulate convergence speeds of the system. Analytically, the solutions using the Lambert W function-based approach for solving the DDE. The eigenvalues obtained are used to determine stability of the system. The convergence speed and stability of the system is analyzed with respect to time delay and variations in trust. Through the analysis of the simulations ran from Graph Theory, one could conclude that the networked system reached consensus at faster speed when an agent is trusted the most. Time delay, however, resulted in each topology of the networked system to become unstable at some point. Varying the trust became important in the event of sensor damage or signal loss. When noise is introduced into a system, one can help tune out the effects of noise on the remaining agents by varying trust in a manner that reduces convergence speed. As the convergence speed was slower, noise was tuned out of the undamaged sensors within the system. Thus, proving that a faster rate of convergence is not always ideal. Variations in trust allow for the system to perform without failure in the event of disturbances such as signal loss and sensor damage. The simulations validate these findings and confirm the analytical methods used. The presented research can be applied towards practical scenarios by aiding in the design of MAS communication to achieve the most desirable performances. Some example scenarios include UGV and UAV cooperation, flying and vehicular ad hoc networks, distributed computing, and many more.

It is also desired to know how long does it take for the developed diagnosers to diagnose system fault occurrences. The diagnosis delay can be characterized by the number of system transitions required to diagnose an occurred fault, as DES systems are time-abstract event-driven. An accurate number of transitions for detecting any particular type of fault depends on the state of system understudy, which is not available to the diagnoser. Therefore, it is not meaningful to find the diagnosis delay at all states of the system. However, it is possible to find an upperbound for the number of transitions required to diagnose an occurred fault, as will be addressed in this dissertation under both asynchronous and semi-asynchronous conditions.

Then, trust issues have been studied further. Trust not only impacts decision making, but it also protects the MAS. Many issues within the MAS community still currently exist such as natural agent deterioration and adversarial attacks. This research demonstrated the novelty of vision-based interactions and how it offers many defenses to current robotic MAS issues as an alternative. Theoretically, this creates a range of acceptable actions an agent can perform at a given time without intruding on its freedom. This assumes an agent is around witness agents that can enforce these rules when broken. The big takeaway of this research is that trust evaluation is not meant to be complex and resource demanding because an MAS has a simultaneous task at hand. Sharing data with suspicious agents is not advisable after it experiences physical trauma or unruly decision-making. This research offers some natural defenses that were implied throughout this paper that fend off against potential loopholes. First of all, the collaboration threshold forces agents to begin without sharing information by default, which reduces risk of oversharing. Adversaries that can modify trust values themselves could lock trust to 1 in order to eavesdrop on all agent data. Alternatively, an adversary can do the same with setting trust to 0 to remove each other and cause chaos. Countering the eavesdropping attack can be done by implementing mutual trust agreements for sharing data. The chaos problem can be solved with a majority vote for removing an agent from

the network. These counters are simple fixes that increase the time and effort the adversary needs to win. This study is not perfect and does not have an answer to some aspects. First of all, some MAS applications involve robots that never enter each other's sensory range. Another abuser to this system is the MITM attack. If the adversary catches the TCP message and modifies it before it reaches the recipient, the system would be in ruin. This can happen without a single difference in the physical behavior or performance of the subtasks. Studies such as SecuredTrust have defenses against this in the literature. Defects that do not show obvious abnormalities can also pose a huge threat to this work.

After those research above, a new method was proposed that combined an analytical delay estimation technique based on the Lambert W function based approach and delay differential equations with the robustness of Smith Predictors. Therefore, an Adaptive Smith Predictor was developed capable of handling delay and its effects analytically which lead to more stable controlled responses. The Adaptive Smith Predictor system performed well in adjusting itself to handle unknown delay in simulation and experiments through the Internet. The method proposed in this research was simple to construct, and easy to apply to real world applications. More work is underway to handle unknown varying delay and to improve other factors (e.g., cases with increasing, decreasing, multiple delays in networks, network-based stochastic control systems). The successful completion of reliable operation through the Internet could easily be transferred to smaller networks. This research focused on constant delay in a system. The upper bounds of the delay allowed is dependent on the applicants preference on responses. This is credited to the Smith Predictor's ability to handle long dead times. One of the challenges so far with varying delay is that once the delay is adapted to, the system may already be affected by the new current delay. The method proposed can be equipped to handle variations in delay. Also, the type of data being transferred through a network needs to be considered in the future. Videos can suffer as a sensor (e.g., dark areas, low bandwidth, etc.). Other sensor feedback, or combinations, should be studied as well. In conclusion, robust remote teleoperation in dynamic networked environments is possible [24].

The last one was control of multiple agents systems of aerial and ground robots using AI/ML-based algorithms. The measured pose real-time tracking response for the drone is processed in such a way to follow the response of the desired pose real-time that correlates to the ground robot specified waypoint. For the drone, the neural network (NN) PIV position real-time tracking performance (inertial frame) displays some oscillation in the response due to the disturbance parameter within the 80 seconds while tracking the desired position real-time tracking response as illustrated in the 3D plot and the Inertial Frame. The response of the drone (NN-PIV) did have sufficient time to reach the desired set point with less overshoot with higher accuracy. Moreover, the networked UAVs show higher accuracy of the NN-based control based on the results. As an example, the large percent errors mean that the drone measured response values are far from the desired response values. The small error means that the drone measured response values were very close to the desired response values. For future works, this research will continue as a work in progress to analyze the development of a NN controller to perform and produce a flexible, and robust control system with additional stability, which reduces the effects of disturbances. Therefore, further study, testing, and research will continue to explore the application of a robust controller for development and the introduction to new sources of disturbances. For instance, the use of a smooth signal generator. This investigation will determine the performance by observing the system's behavior to identify and reduce the unpredictable effects of disturbance in a UAV control system.

In summary here is a list of the main accomplishments for this task include:

- Quantitative analysis of effects communication delays and design of controllers were conducted via infinite eigenspectrum analysis.
- Developing of control strategies considering variant delays of networked systems by incorporating estimation of delays into the control-loop, and control for synchronization of networked systems.
- A predication-based delay control method is implemented on systems connected through internet for robust synchronization.
- A vulnerability of delay predication for real-time control was addressed and compensated by adding online delay estimation method.
- Also, tt was implemented on two systems connected through Internet. Analytical theories for delay differential equations are used and controllers are designed based on the finding.
- Investigating the relations between different diagnosis schemes including, synchronous, semi-asynchronous, and asynchronous diagnosabilities.
- System's behaviors were analyzed using the Neural Network controller to perform plant identification and to apply the Feedforward model (PIDVAff) approach to stabilize the system's response.
- The neural controller is designed in such a way that makes the plant output to follow the output of the reference model
- development of a method to find a way to optimize networks for multiple agents systems (MAS).

3.2.7 Project Progress for Task T2-7 (Wireless Scale Free Topologies for Resiliency and Jamming Immunity)

3.2.7.1 Period of Performance under

Task T2-7 Start Date: 2016

Conclusion Date: May 2021

Faculty lead: Dr. Brian Kelley

3.2.7.2 General Description of Task T2-7

The main objective of this task is to support the ubiquitous deployment of densely connected networks. Privacy and the prevention of eavesdropping are significant concerns in IoT and military applications. Furthermore, densely connected networks require improvements to throughput, latency, and capacity. These concerns warrant motivation for new approaches, ideally approaches that improve upon current techniques in every way. Task 2-7 created new technologies that meet these conditions by providing low-latency privatization of signals and high-throughput high-efficiency communications that exceed the abilities of the current generation of communications technology. For example, we demonstrate low-latency alternatives to encryption that allow for quick and secure

conversations between two users. Also, we demonstrate the ability to increase the spectral efficiency of the best-performing antenna configuration on the market by a factor of 10.

3.2.7.3 Objectives for Task T2-7 over the Course of Project

Robust and secure connections are becoming increasingly necessary as heterogeneous and ubiquitous connections saturate our environment. Therefore, achieving bleeding-edge data rates and privacy are critical components of Beyond5G and 6G standard formulations. The research outlined in this document achieves an unprecedented increase in throughput through the use of a multi-input multi-output (MIMO) technology called K-User MIMO. Furthermore, we show that a 10-fold increase in spectral efficiency is achievable with K-User MIMO, compared to the traditional Massive MIMO techniques. In addition, for the security aspect of Task 2-7, we invented new secure waveform configurations that privatize communication links based on random channel effects between transmitters and receivers. Our Physical Layer Security method achieves lower computational overhead and latency for communications than traditional data encryption techniques.

3.2.7.4 Progress Against Planned Objectives in Task T2-7

3.2.7.4.1 Research 1: On the Application of K-User MIMO for 6G Enhanced Mobile Broadband The goal of 5G is to enable a fully connected society such that instant informa- tion is available just a touch away. 5G achieves this through three key paradigms viz., enhanced Mobile Broadband (eMBB) for gigabit data rates, Ultra Reliable and Low Latency Communica- tions (URLLC) for latency less than 1ms and massive Machine-Type Communications (mMTC) for 1 million connected devices/sq. km [25, 26]. These paradigms are supported by a multi-layer technology strategy including small cell architectures [27, 28], millimeter wave communication and massive MIMO. Millimeter wave systems facilitate communication in the high Radio Frequency (RF) bands using analog, digital and hybrid breamforming [29, 30]. Massive MIMO deploys large antenna arrays at base stations and operates in the low to mid RF bands [31–34].

3.2.7.4.2 Background and prior research on K-User MIMO K-User MIMO is an architecture in which there are K access points and K mobile devices, each equipped with multiple antennas, i.e., spatial dimensions. In the simplest form of K-User MIMO, each Access Point connects to one of the K mobile devices. We consider a different form of K-User MIMO, known as K-User MIMO X, in which each of the K mobile devices receives signals from K access points. This is shown in Figure 100. The all-to-all architecture achieves very high throughput whilst supporting flexibility in achieving diversity. Each of the K access points could send redundant information streams to maximize reliability. Alternatively, they could send unique information streams in order to maximize capacity. K-User MIMO X can switch between these two modes without any change in the mathematics of the algorithm. Further scenarios can be envisioned in which K-User MIMO X allows for adaptive allocation of power to users with more favourable Signal to Noise Ratios. Additional encoding of data across time and frequency could be applied so the signals could adapt to malicious behaviour such as jamming and eavesdropping.

In any form of K-User MIMO, each mobile device receives both its desired signals and interference signals (signals meant for other mobile devices). To manage interference, K-User MIMO systems are often studied in the context of Interference Alignment (IA). IA is a technique that aligns interfering signal vectors in order to maximize interference-free space at each mobile device [35].

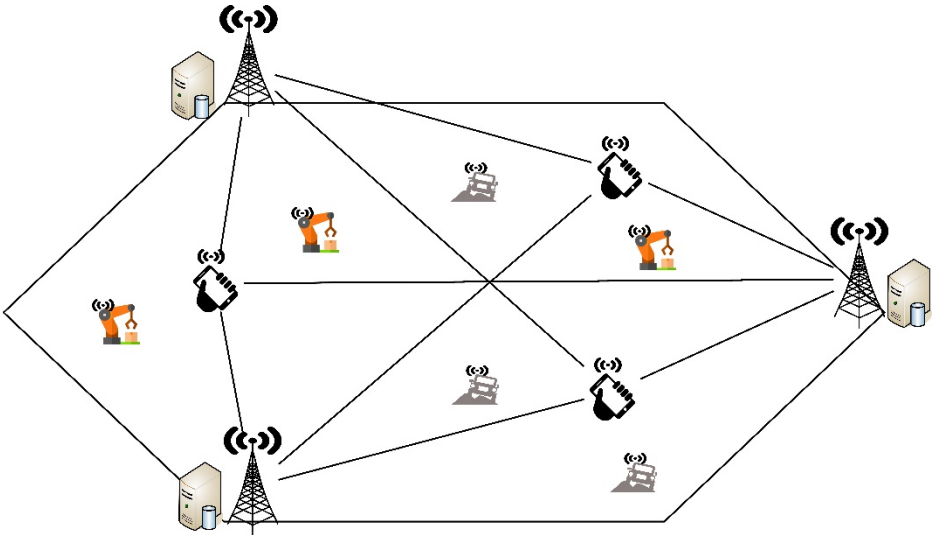


Figure 122: K-User MIMO X network Example for K = 3

By applying suitable channel dependent precoders to the transmit signals, and beamformers to the receive signals, several interfering users can communicate simultaneously. Alignment helps confine the interference at each mobile device to a smaller dimensional subspace while projecting the desired signals into the null space of the interference.

Several works have analyzed IA on a theoretical level. A typical metric used to characterize IA is known as Degrees of Freedom (DoF). DoF is defined as the number of spatial dimensions that are free from interference [36]. The authors in [37] have provided examples showing acheivability of IA and various DoF in K-User interference networks with different antenna configurations. In [38], an iterative algorithm for obtaining the precoders and beamformers is presented for a Time Divi- sion Duplex (TDD) mode of operation. The precoders in this method are a function of the dual relationship between the MIMO forward and reverse channels. Another IA framework involving TDD channels is presented in [39]. In [40], interference alignment in MIMO downlink networks is investigated where precoders are derived by eigen decomposition of the MIMO channels.

Another IA scheme for a K-User MIMO X network, is proposed in [41]. By appropriately precoding the transmit signals, this scheme maximizes the interference free space by limiting the interference at every mobile device to half of the received signal space. Further, by applying a zero forcing beamformer which is a function of interfering channels and precoders, interference cancellation has been achieved for K 3. The algorithms in [41] are purely theoretical and we improve upon them.

Discussion on how to demodulate symbols is not provided in [41]. Further, in order to maximize capacity, we wish to operate the K-User MIMO X system such that each access point is transmitting different symbols to each mobile device, whilst all being on the same frequency subcarrier. We have investigated the case where each access point is on a different subcarrier in [42] and found that the

bandwidth and therefore the capacity of the K-User system is reduced by a factor of K which is significant. We present a new signal separation beamformer to regain the lost factor of K.

Interference Alignment uses precoders and beamformers that are channel dependent. Naturally, channel estimation is critical to IA [43, 44]. Both [41] and our previous works [42, 45, 46] either consider perfect channel state information or do not consider exact channel estimation error models. Neither considers the overheads arising from transmitting pilot symbols for estimation. [41] assumes perfect channel state information. [42] does not explicitly estimate the channel but assumes a Cramér-Rao variance for the estimation error, which is only a lower bound on the error. In [45] the channel is not estimated but the effects of imperfect estimation are simulated. Residual interference due to imperfect precoders and beamformers is modelled as a random variable and an expression for its distribution is provided. [46] is the first paper that introduces signal separation concepts for K-user MIMO but it too assumes perfect channel estimation.

We extend our previous work to provide realistic and practical capacity results for K-User MIMO X systems, that account for channel estimation overheads.

Table 14: System model parameters for key derivations

Parameter	Description	Size	For K 3 = ≥ 6
M	Minimum number of antennas at each access point & mobile device	$\geq K(K-1)$	
N_D	Number of desired signals at each mobile device	K	3
N_I	Number of interference terms at each mobile device	$K(K-1)$	6
H_{ij}	Channel matrix between access point j and mobile device i	$M \times M$	6 × 6
v_{ij}	Precoder vector for signal between access point j and mobile device i	$M \times 1$	6 × 1
s_{ij}	Symbol to be transmitted between access point j and mobile device i	1×1	1 × 1
w_i	AWGN at mobile device i	$M \times 1$	6 × 1
P_i^I	Matrix of aligned interference column vectors at the mobile device i	$M \times \frac{N_I}{2}$	6 × 3
U_i	Zero forcing beamformer matrix at mobile device i	$M \times M - \frac{N_I}{2}$	6 × 3
P_{ij}^D	Matrix of desired column vectors at mobile device i to isolate signal from access point j	$M - \frac{N_I}{2} \times N_D - 1$	3 × 2
U_{ij}	Signal Separation beamformer matrix at mobile device i to isolate signal from access point j	$M - \frac{N_I}{2} \times M - (\frac{N_I}{2} - N_D - 1)$	3 × 1
N_s	Number of recovered copies of each symbol s_{ij}	$M - \frac{N_I}{2} - (N_D - 1)$	1

3.2.7.4.3 System model for cellular based K-User MIMO This section describes the system model for K-User MIMO X that shows, transmit precoding, receive interference cancellation and signal separation. Table 14 shows the parameters used in the key equations and derivations. Figure 101 shows the overall K-User MIMO protocol steps. Lastly, an Orthogonal Frequency Division Multiplexing (OFDM) multiple access protocol is shown. This protocol illustrates the pilot overhead resulting from serving several (> K) mobile devices.

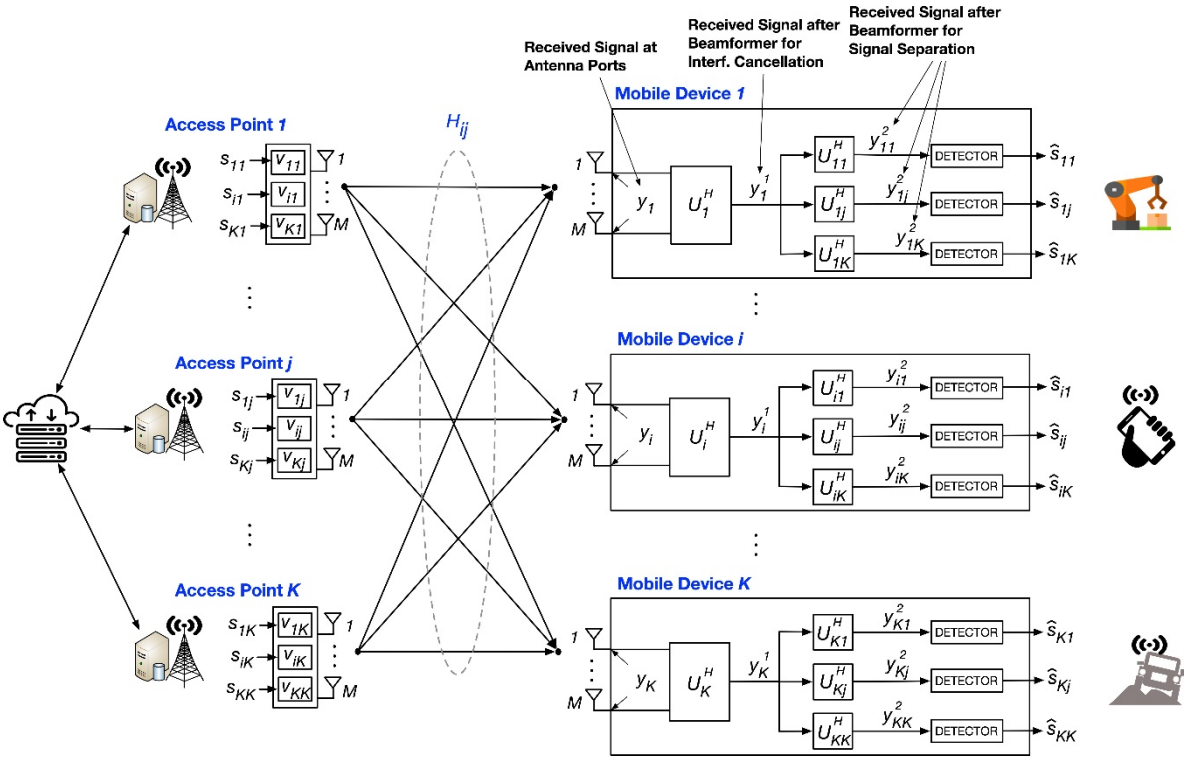


Figure 123: K-User MIMO X network showing the application of precoders at the access points as well as stage 1 and stage 2 beamformers (Equations (50)- (52))

3.2.7.4.4 Received Signal at antenna Each of the K access points and mobile devices is equipped with M antennas. The all-to-all connectivity results in each mobile device receiving K desired signals and $K(K-1)$ interfering signals. This is shown in Figure 101 and(50). Transmitted symbols s_{ij} between the j^{th} access point and the i^{th} mobile device are precoded by length M precoder vectors v_{ij} and transmitted over Rayleigh fading channels H_{ij}. Without loss of generality, we consider downlink transmissions. The received signal at the i^{th} mobile device is given by,

$$y_i = \underbrace{H_{ij}v_{ij}s_{ij}}_{\text{Desired signal}} + \underbrace{\sum_{j=1}^{K}\sum_{k=1,k\neq i}^{K} H_{ij}v_{kj}s_{kj}}_{\text{Total interference}} + \underbrace{w_i}_{\text{Noise}} \quad (50)$$

We focus on the maximum capacity scenario. Therefore it is assumed that each s_{ij} is unique. The precoders v_{ij} are channel dependent and are obtained by solving a system of alignment equations. The procedure for obtaining the precoder vectors is shown in Appendix A. The noise at the i^{th} mobile device, w_i is assumed to be 0 mean Additive White Gaussian Noise (AWGN) with variance $\sigma_{w_i}^2 = E[w_i w_i^*]$ where $E[.]$ represents the expected value.

3.2.7.4.5 Received signal after beamformer for interference cancellation A beamformer matrix U_i is applied to the received signal y_i to cancel the interference as shown below.

$$y_i^1 = U_i^H \cdot y_i = U_i^H \sum_{j=1}^{K} H_{ij} v_{ij} s_{ij} + U_i^H \cdot \sum_{j=1}^{K} \sum_{k=1, \neq i} H_{ij} v_{kj} s_{kj} + U_i^H w_i \tag{51}$$

$$I_\epsilon \approx 0$$

where, I_ϵ is the residual interference after cancellation. In the case of perfect channel state information, it is exactly equal to zero. The interference cancellation beamformer U_i is obtained by first generating a matrix, whose columns contain aligned interference vectors. The left hand side of the Singular Value Decomposition (SVD) of this matrix contains the beamformer. The derivation of this beamformer is shown in Appendix A.

3.2.7.4.6 Received signal after beamformer for desired signals separation Symbols transmitted from access points on the same frequency subcarrier, add coherently at the mobile device. Under this model, symbol recovery is straight forward when each access point sends the same symbol to the i^{th} mobile device. However, for maximizing capacity, each access point must be able to send different symbols to the i^{th} mobile device. In such cases, a second beamformer operator is applied after interference cancellation, as shown in (52). This second operator is applied K times at each mobile device (shown in Figure 100) and helps separate the signals sent from each of the K access points.

$$y_{ij}^2 = U_{ij}^H U_i^H y_i^1 = U_{ij}^H U_i^H \sum_{j=1}^{K} H_{ij} v_{ij} s_{ij} + U_{ij}^H I_\epsilon + U_{ij}^H U_i^H w_i \tag{52}$$

$$I_\epsilon \approx 0$$

$$= U_{ij}^H U_i^H H_{ij} v_{ij} s_{ij} + U_{ij}^H U_i^H w_i \xrightarrow{\text{To detector for symbol recovery}} \hat{s}_{ij}$$

The signal separation beamformer U_{ij} is obtained by first generating a matrix whose columns contain desired signal vectors from other access points that act as interference when recovering the symbol transmitted from a specific access point. The left hand side of the SVD of this matrix contains the beamformer. The derivation of this beamformer is shown in Appendix A.

After separation, the desired signals from each of the K access points can be decoded by correcting for the effects of U_{ij}, U_i, H_{ij} and v_{ij}. The decoding and detection process that recovers the symbols s_{ij} is shown in Appendix A.

3.2.7.4.7 K-User MIMO X Multiple Access Protocol We now apply K-User MIMO X to a typical OFDM cellular scenario. Let us assume that there are nK mobile devices ($n \geq 1$) associating with K access points forming n K-User Groups (UG). In a single UG, there are K^2M^2 Channel Impulse Responses (CIR) that need to be estimated. We propose that each of the MK transmit antennas sends pilot symbols on non-overlapping OFDM symbol times. We assume the use of pilot signals that are configured for both synchronization as well as channel estimation. An example of such a sequence is the Zadoff Chu sequence, commonly used in LTE. Akin to LTE pilot signals, when one transmit antenna is sending pilots, all other transmit antennas are off. This

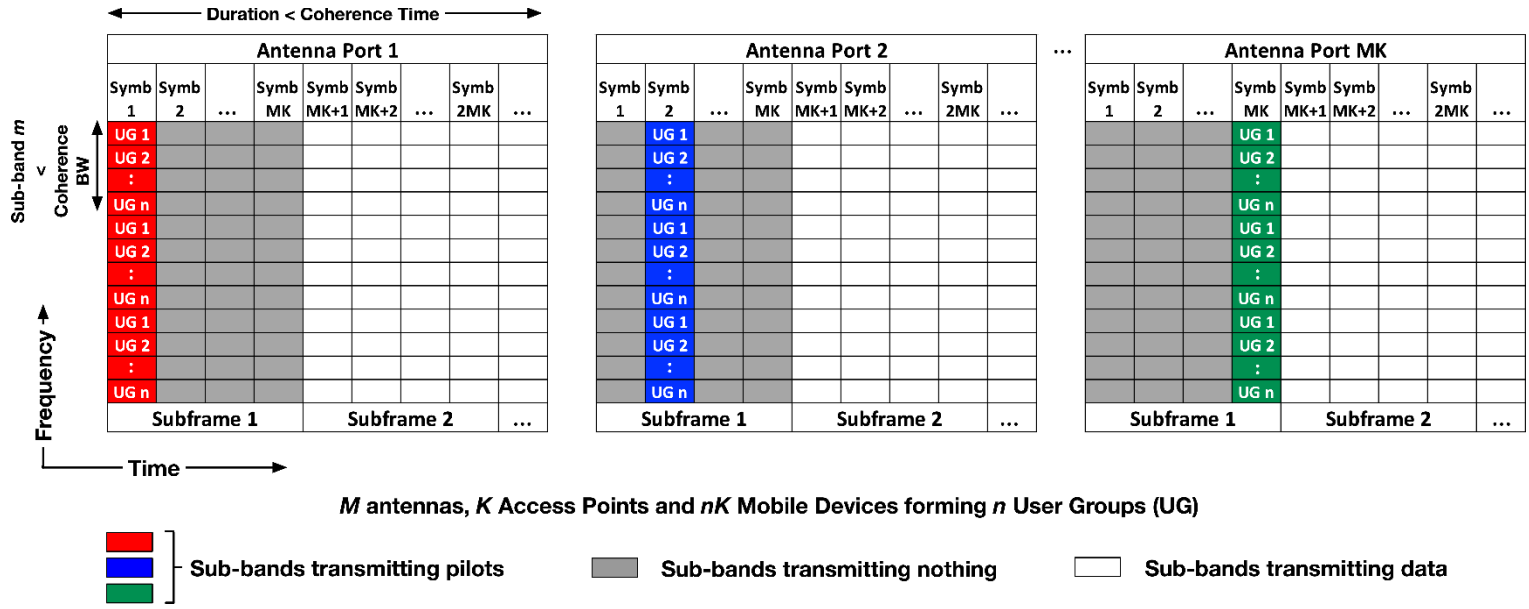


Figure 124: Example showing the channel estimation overhead for n K-User Groups (UG). The bandwidth allocated for estimation is divided among the n K-User Groups. Each group transmits pilot signals in its allocated band.

is shown in Figure 102. In the time domain, the channel needs to remain constant for at least MK symbol times. We leverage frequency domain resources to support the n User Groups. In the frequency domain, the available bandwidth B is divided into m sub-bands where $m = \frac{B}{B_c}$ and B_c is the channel coherence bandwidth. Each of the m sub-bands is divided equally among the n K-User Groups. As shown in Figure 102, each UG gets a chunk of bandwidth in each of the m sub-bands in which to transmit pilot signals for synchronization and channel estimation. The channels in the sub-bands not available for a certain UG, can easily be obtained by interpolation. The channel estimates are conveyed by the mobile devices to a global network entity in the backhaul which makes all channels available to all access points and mobile devices through the appropriate interfaces. The overhead from this step is not considered and will make up future work.

3.2.7.4.8 Derivation of Shannon Capacity for K-User MIMO X And Small Cell Geometric Capacity In Rayleigh Fading This section presents an analysis of Shannon Capacity as a function of K, with and without pilot overheads. Further simulation results show the statistical distributions of capacity for $K = 3$ in Rayleigh fading and small cell geometries.

3.2.7.4.9 Ideal K-User MIMO X capacity versus K excluding pilot overhead The theorem for the upper bound capacity for K-User MIMO X incorporates channel and beamformer gains that scale with K and antenna array size M. It is defined below.

Theorem 1. *The upper bound multi-user capacity of a K-User MIMO X system is bounded by*

$$C_{bits/sec} \leq BK^2 log_2(1 + [M - \tfrac{N_I}{2} - (N_D - 1)][M - \tfrac{N_I}{2}]\ M^2 \times SINR)$$

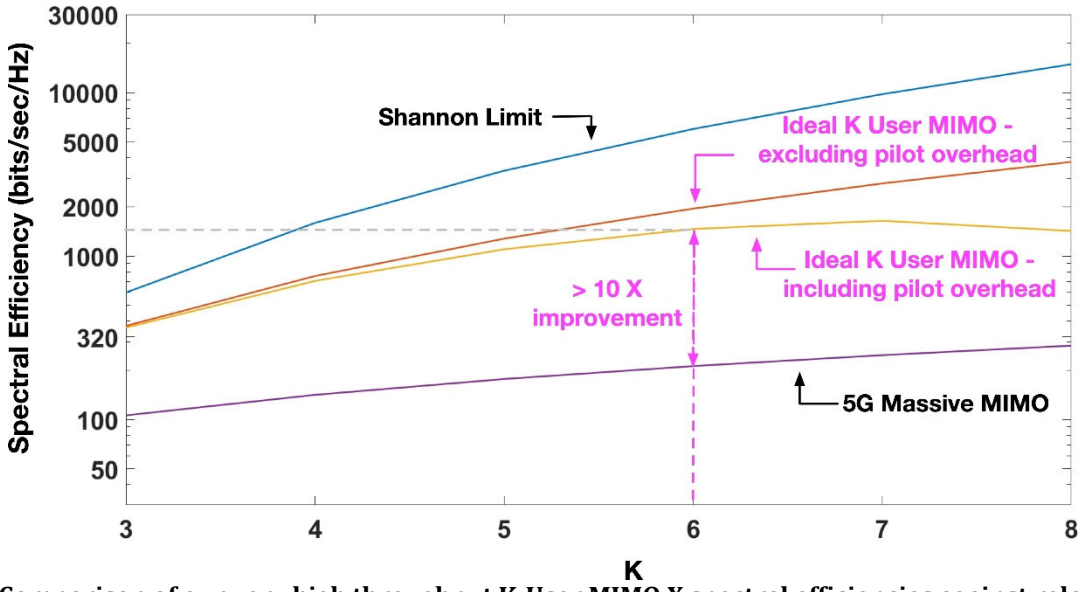


Figure 125: : Comparison of our very high throughput K-User MIMO X spectral efficiencies against related technologies such as 5G-NR Massive MIMO. Spectral Efficiencies are for a 100m cell in an Indoor A channel scenario. The curve for Massive MIMO is based on the formula shown in [47]. The number of antennas is $M = K(K-1)$.

where B is the bandwidth, and the $SINR = \frac{P_t d^{-a}}{\sigma_w^2}$ includes the transmit power P_t and the distance dependence based on a path loss exponent α and target distance d. The proof of the theorem and verification by simulation are shown in Appendices B and C respectively. The result of Theorem 1 gives the upper-bound multi-user capacity of the combined K^2 streams. Note that the theorem represents an unconstrained case which assumes that the entire available time-frequency resources are available only for data transmission.

3.2.7.4.10 Ideal K-User MIMO X capacity versus K including pilot overhead Real systems are impacted by various overheads for synchronization signals, channel estimation pilot signals and exchange of other control information. We consider two of the most important overheads, that of synchronization and channel estimation. We propose the use of pilot signals configured for both time synchronization and channel estimation. Consequently, we define the following theorem that refines Theorem 1 to include pilot overheads.

Theorem 2. *If MK channel estimation symbols can be transmitted in less than the channel coherence time, i.e., $MKT_{symb} < T_c$, the remaining time can be reserved exclusively for data transmission. The capacity equation in Theorem 1 can be modified as follows,* $C_{bits/sec} \leq \frac{T_c - MKT_{symb}}{T_c} BK^2 log_2(1 + [M - \frac{N_I}{2} - (N_D - 1)][M - \frac{N_I}{2}]^2 M^2 \times SINR)$

where T_c is the channel coherence time. T_{symb} is the duration of 1 OFDM symbol given by

$(N + CP)T_s$ where N is the size of the FFT, CP is the size of the cyclic prefix in samples and T_s is the sampling period.

Figure 103 shows the Shannon Capacity curves for K-User MIMO X with and without pilot overheads. The key observation from the curves is that while the unconstrained capacity from Theorem 1, continues to grow with K, that is not the case when pilot overheads are taken into account. After a certain point, it can be seen that the pilot overheads overwhelm the gains from K-User MIMO and the capacity begins to drop. In highly varying channel scenarios, it is more beneficial to operate at a lower value of K and schedule several User Groups in a multiple access framework similar to that shown in Figure 102. Figure 103 also compares the performance of our K-User MIMO X system with 5G-NR Massive MIMO [47]. It should be noted that the Massive MIMO model assumes one access point equipped with M antennas and K single antenna mobile devices. The Shannon limit is also plotted. This limit is based on the analysis provided in [48] and assumes a system with a single access point and single mobile device each with MK antennas.

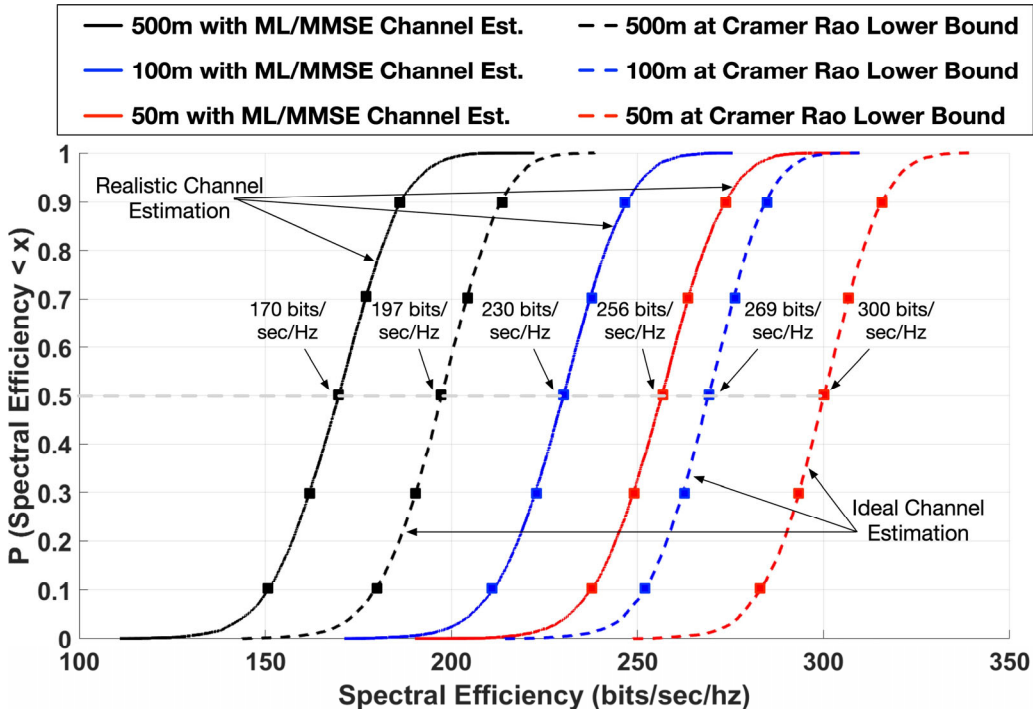


Figure 126: Cumulative distribution functions of spectral efficiencies in bits/sec/Hz for K 3 in single hexagonal cells of radius 50m, 100m and 500m for Indoor A channel scenarios [49]. Results are shown both with realistic channel estimation using ML/MMSE and also at the Cramér-Rao Lower Bound.

3.2.7.4.11 Capacity results for Small Cell K-User MIMO at K = 3 This section describes the simulation model, choosing K= 3 as an example. Simulations are done in MATLAB and the key parameters are listed in Table 15. Cell spectral efficiency performance in 500m, 100m and 50m hexagonal cells is obtained and shown in Figure 104.

Table 15: Simulation Parameters

Channel Model	Rayleigh Fading
Channel Scenario	Indoor A [49]
Cell Radius	50 m, 100 m, 500 m
Transmit Power	16 dBW
Total Bandwidth	20 MHz
FFT Size (N)	2048
Cyclic Prefix (CP)	512 samples
Sampling frequency	30.72 MHz
Sub-carrier spacing	15 kHz
Number of used sub-carriers	1320
Noise Figure	4 dB
Thermal Noise Density	203.9 dBW/Hz
Path Loss Exponent	3

Without loss of generality, the hexagonal geometry is chosen, for simplicity of analysis. The system can easily be translated to stochastic geometries, commonly associated with 5G systems. 3 access points are placed on alternate corners of the cell. 3 mobile devices are placed uniformly within the cell. The simulations assume an exponential path loss $L_{ij} = d_{ij}^{-\alpha}$.

While K-User MIMO X is not limited by any particular channel scenario, we note that one example of a use case is a high throughput IoT robotic factory model. Hence without loss of generality, a Rayleigh fading channel model with the Indoor A power delay profile [49] is used. Considering an Indoor A channel scenario with parameters such as, velocity v = 3kmph, carrier frequency f_c = 1.9GHz, speed of light $c = 3 \times 10^8$m/s, the channel coherence time T_c is calculated as, $T_c = \sqrt{\frac{9}{16\pi f_d^2}} \approx 80$ms where f_d is the Doppler shift given by $f_d = \frac{vf_c}{c}$. Let us assume that we only have a fraction of the channel coherence time, say 60ms.

Multiple channel and location trials are run. In each trial, the channel is estimated in the first 18 symbol times (1.5ms). We assume the use of well known estimation methods such as the Maximum Likelihood (ML) or Minimum Mean Square Error (MMSE).

For statistical analysis, it suffices to assume the presence of one K-User group. In such a case, the entire bandwidth is available for pilot signals. We use the Zadoff Chu sequence for both time synchronization and channel estimation. After channel estimation, the remaining 58.5ms is available for data transmission. The Cumulative Distribution Function (CDF) of the spectral efficiency is calculated from the multiple trials and plotted in Figure 104. These spectral efficiencies account for pilot signal overhead as well as estimation error.

In the case of $n > 1$ K-User Groups, in the channel estimation symbols, the bandwidth can be divided as shown in Figure 102. In the data transmission symbols, the bandwidth could also be divided into sub-bands in which the different User Groups could be network scheduled using greedy or proportional fair algorithms.

3.2.7.4.12 Incorporation of channel estimation errors Figure 105 shows the variance of the estimation error as a function of signal to noise ratio for both the ML and MMSE estimates.

The Cramér-Rao Lower Bound (CRLB) is also shown. As expected, the ML and MMSE variances are higher than the CRLB. The equations for the channel estimation methods and the CRLB are well researched in other works and are hence shown in Appendix D. Figure 104 also shows the spectral efficiency if the estimation error variance is at the Cramér-Rao Lower Bound (CRLB). To simulate this, we calculate the CRLB as shown in (68) in Appendix D. The CRLB is calculated for each transmit-receive antenna link and error terms are drawn from CN $(0, \sigma^2_{CRLB})$. These errors are added to the actual channels to simulate estimation error at the CRLB.

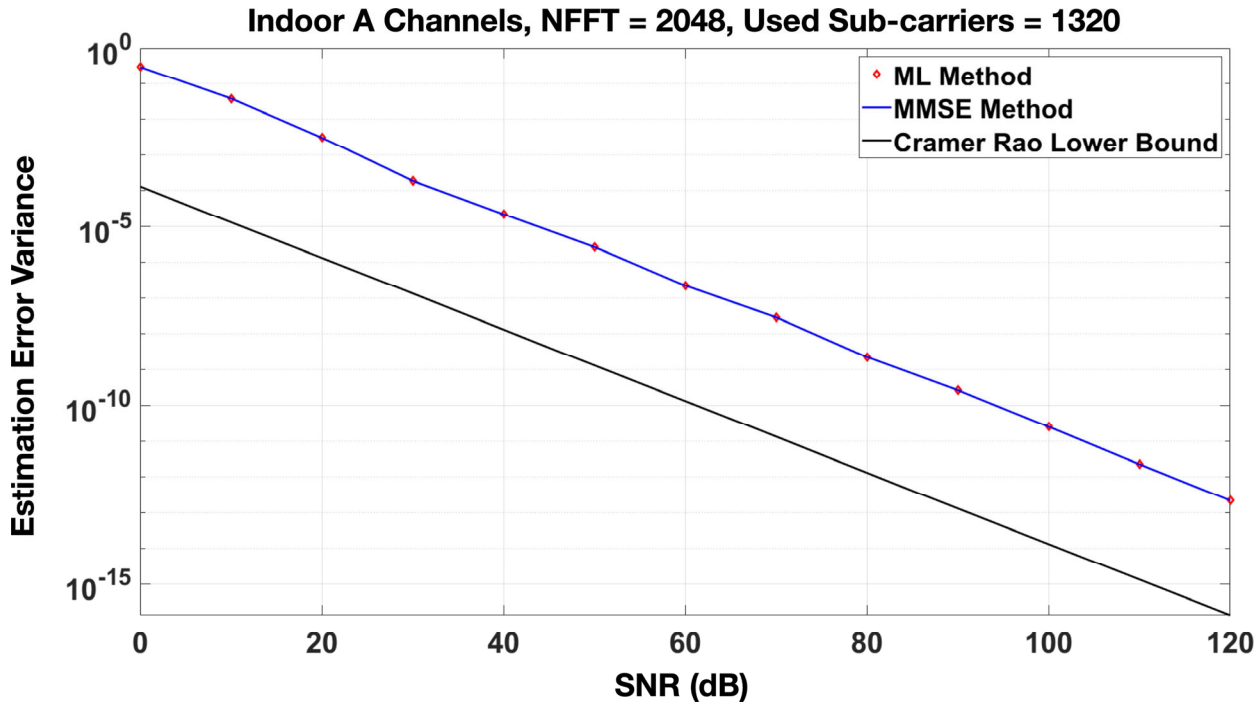


Figure 127: Comparison of channel estimation error variance between ML/MMSE and the Cramér-Rao Lower Bound (CRLB)

It can be seen from Figure 104 that the median best case spectral efficiency in 500m, 100m and 50m cells are 170, 230 and 256 bits/sec/Hz respectively. These results are under Rayleigh fading conditions. If the channel follows a Rician distribution, it means that there is one dominant line of sight path in the Channel Impulse Response. The stronger the line of sight component, the rarer the occurrences of deep fades. Though this scenario is not simulated explicitly, we have investigated another scenario, which is maximum capacity scheduling with Rayleigh fading. This approach schedules *K*-User groups in a particular sub-band with the best channel conditions, thus weeding out deep fade channel instances. We have found very little improvement in spectral efficiency with this type of scheduling. We believe that the same will be the case with Rician channels. The reason for this is that the non-co-located nature of K-User MIMO transmitters already provides enough channel diversity to overcome the effects of deep fades.

The spectral efficiencies in Figure 104 are multi-user values and take into account the fact that channel estimation through pilot signals takes up 18 symbol times, where no data is transmitted. In a 20MHz band this amounts to a best case scenario of 3.4Gbps, 4.6Gbps and 5.1Gbps respectively and

in the case of 5 aggregated bands, data rates in excess of 17Gbps, 23Gbps and 25Gbps respectively can be achieved. This underlines the wide range of exciting possibilities that can be achieved in beyond-5G and 6G networks with K-User MIMO X.

The 6G extension of eMBB, defined in [50] as eMBB Plus will serve mobile and IoT communications with data rate requirements far greater than 5G. The high throughput and spectral efficiency of K-User MIMO X lend themselves well to help support eMBB Plus. 6G will also be more machine learning and security driven [51, 52]. The all-to-all nature of K-User MIMO X, with its ability to switch between maximum capacity and maximum reliability modes, makes it particularly suitable for future machine learning integration. Machine learning algorithms could be used to come up with new encoding schemes across the K transmitters that can help adapt to spatially and temporally varying channel conditions as well as eavesdroppers and jammers. This flexibility leads perfectly into the 6G version of URLLC, known as event-defined-URLLC [52] which provides context-aware communications, not thought of in 5G.

3.2.7.4.13 Conclusion We have reviewed and identified the massive scope for increased throughput for beyond-5G or 6G networks. Under realistic channel estimation constraints, we have provided a K-User MIMO X framework that can cancel interference, demodulate and maximize capacity through signal separation. Further practical aspects such as OFDM multiple access for channel estimation and data transmission have been described. Lastly, cell capacity performance has been simulated and compared with related technologies.

3.2.7.4.14 Appendix: K-User MIMO Interference Alignment Mathematics for K=3

3.2.7.4.15 System of Interference Alignment Equations for precoder generation Interference cancellation can be achieved subject to the following constraints.

Postulate 1 [41]: At each mobile device, interfering signals from the same access point, cannot be aligned in the same direction such that, $H_{ij}v_{kj} \neq H_{ij}v_{lj}$ where $i \neq k \neq l$.

Postulate 2 [41]: In a user system, since each mobile device receives $K(K-1)$ interference components, in order to align $(K-1)$ interference signals along K dimensions, the condition to be met is $span(H_{im}v_{km}) = span(H_{in}v_{ln})$ where $k, l \neq i$

Since there are $K(K-1)$ interference terms at each mobile device, the minimum number of antennas at each access point and mobile device has to be $M = K(K-1)$.

Table 16 show one possible set of Interference Alignment equations for $K = 3$. These equations are formed by dividing the $K(K-1)$ interference terms at each mobile device into pairs, subject to postulates 1 and 2. Each pair represents the left and right hand sides of a single IA equation. Since there are $K(K-1)$ interference terms at each mobile device, there will be $\frac{K(K-1)}{2}$ IA equations. These equations will be used to solve for the precoders.

Each access point will apply K precoders. From the IA equations, the precoders v_{ij} are obtained

Table 16: K-User MIMO X Interference Alignment conditions, K = 3

	IA conditions [41]
Rx 1	span($H_{11}v_{21}$) = span($H_{12}v_{22}$) span($H_{11}v_{31}$) = span($H_{13}v_{33}$) span($H_{12}v_{32}$) = span($H_{13}v_{23}$)
Rx 2	span($H_{21}v_{11}$) = span($H_{22}v_{12}$) span($H_{21}v_{31}$) = span($H_{23}v_{13}$) span($H_{22}v_{32}$) = span($H_{23}v_{33}$)
Rx 3	span($H_{31}v_{11}$) = span($H_{33}v_{23}$) span($H_{31}v_{21}$) = span($H_{32}v_{12}$) span($H_{32}v_{22}$) = span($H_{33}v_{13}$)

as follows,

$$\begin{aligned} v_{12} &= H_{22}^{-1}H_{21}v_{11} & v_{33} &= H_{13}^{-1}H_{11}v_{31} & v_{21} &= H_{31}^{-1}H_{32}v_{12} \\ v_{32} &= H_{22}^{-1}H_{23}v_{33} & v_{22} &= H_{12}^{-1}H_{11}v_{21} & v_{23} &= H_{13}^{-1}H_{12}v_{32} \\ v_{13} &= H_{33}^{-1}H_{32}v_{22} & v_{11} &= H_{31}^{-1}H_{33}v_{23} & v_{31} &= (H_{21})^{-1}H_{23}v_{13} \end{aligned} \quad (53)$$

The initial value of v_{11} is obtained by first defining a matrix E [41] as follows,

$$\begin{aligned} E = {} & (H_{31})^{-1}H_{33}(H_{13})^{-1}H_{12}(H_{22})^{-1}H_{23}(H_{13})^{-1}H_{11} \\ & \times (H_{21})^{-1}H_{23}(H_{33})^{-1}H_{32}(H_{12})^{-1}H_{11}(H_{31})^{-1}H_{32} \\ & \times (H_{22})^{-1}H_{21} \end{aligned} \quad (54)$$

It is to be noted that E is obtained from (53). Then, v_{11} is arbitrarily chosen to be one of the eigen vectors of E and subsequently all the other precoders can be obtained in the order, v_{12}, v_{21}, v_{22}, v_{13}, v_{31}, v_{33}, v_{32} and v_{23}.

3.2.7.4.16 Obtaining the beamformer for interference cancellation The zero forcing beamformer matrix U_i is obtained by first defining a matrix P_i^I for each mobile device and taking the Singular Value Decomposition (SVD) as follows,

$$\begin{aligned} P_1^I &= [H_{11}v_{21} \quad H_{11}v_{31} \quad H_{12}v_{32}] \\ &= [\bar{U}_1^{(1)} \quad \bar{U}_1^{(0)}] \begin{bmatrix} \bar{\Lambda}_1 \\ 0 \end{bmatrix} [\bar{V}_1^{(1)} \quad \bar{V}_1^{(0)}]^H \end{aligned} \quad (55)$$

Where P_1^I is the set of aligned interfering column vectors at the first mobile device. From (55) we can set $\bar{U}^{(0)} = U_1$, the zero forcing beamformer at the first mobile device. The number of columns in U_1 is equal to the number of non zero singular values in $\bar{\Lambda}_1$. Similarly we determine the beamformers at the second and third mobile devices by defining $P_2^I = [H_{21}v_{11} \quad H_{21}v_{31} \quad H_{22}v_{32}]$ and $P_3^I = [H_{31}v_{11} \quad H_{31}v_{21} \quad H_{32}v_{22}]$ where $P^I{}_2$ and $P^I{}_3$ are the sets of aligned interfering column vectors at the second and third mobile devices respectively.

3.2.7.4.17 Obtaining the beamformer for signal separation At the i^{th} mobile device in order to separate the different signals sent from the K access points we apply a second beamformer matrix. This operator is applied K times at each mobile device. This matrix is obtained by defining a matrix P_{ij}^D for each access point-mobile device pair and taking the SVD as follows,

$$\begin{aligned} P_{11}^D &= [U_1^H H_{12} v_{12} \quad U_1^H H_{13} v_{13}] \\ &= [\bar{U}_{11}^{(1)} \quad \bar{U}_{11}^{(0)}] \begin{bmatrix} \bar{\Lambda}_{11} \\ 0 \end{bmatrix} [\bar{V}_{11}^{(1)} \quad \bar{V}_{11}^{(0)}]^H \end{aligned} \tag{56}$$

Where P_{11}^D is the set of desired column vectors at mobile device 1 corresponding to access points 2 and 3. From (56) we can set $\bar{U}_{11}^{(0)} = U_{11}$. The number of columns in U_{11} is equal to the number of non zero singular values in $\bar{\Lambda}_{11}$. The matrix U_{11} when multiplied to the received signal after zero forcing, isolates the desired signal at mobile device 1 from access point 1. Similarly the desired signals at mobile device 1 from access points 2 and 3 respectively can be isolated by defining $P_{12}^D = [U_1^H H_{11} v_{11} \quad U_1^H H_{13} v_{13}]$ and $P^D{}_{13} = [U_1^H H_{11} v_{11} \quad U_1^H H_{12} v_{12}]$ where $P^D{}_{12}$ and $P^D{}_{13}$ are the sets of desired column vectors at mobile device 1 corresponding to access points 1, 3 and 1, 2 respectively. The same process can be repeated at the other mobile devices.

3.2.7.4.18 K-User MIMO X demodulation and symbol detection The following de-modulation process helps recover symbols arriving from each access point. Note that this process is applied after zero forcing and signal separation. Defining $\Gamma_{ij} = U_{ij}^H U_i^H H_{ij}$ and taking the SVD, we have,

$$\Gamma_{ij} = \Phi_{ij} \cdot \Lambda_{ij} \cdot \Psi_{ij}^H \tag{57}$$

Where Λ_{ij} is a diagonal matrix containing the singular values of Γ_{ij} such that,

$$\Lambda_{ij} = \begin{bmatrix} \lambda_{ij}(1) & \lambda_{ij}(2) & \cdots & 0 & 0 \\ \vdots & \vdots & & & \\ 0 & 0 & \lambda_{ij}(N_s) & 0 & \cdots \; 0 \end{bmatrix} \tag{58}$$

The information symbol from access point j to mobile device i is estimated by the element-wise division,

$$[\hat{s_{ij}}(1), \ldots, \hat{s_{ij}}(\gamma), \ldots, \hat{s_{ij}}(N_s)]^T = \frac{\phi_{ij}^H \tilde{y}_{ij}}{\psi_{ij}^H v_{ij}} \tag{59}$$

The demodulation process recovers N_s copies (refer Table 14) of each symbol s_{ij} corresponding to the number of non-zero singular values in Λ_{ij}. Due to interference cancellation, the demodulated signal obtained from (59) takes the form,

$$\hat{s_{ij}} = \sum_{\gamma=1}^{N_s} \lambda_{ij}(\gamma) s_{ij}(\gamma) + \hat{w}_i \tag{60}$$

where $\lambda_{ij}(\gamma)$ is the γ^{th} diagonal element in Λ_{ij} and $\hat{w}_{ij}$ is the noise after demodulation.

The spectral efficiency after interference cancellation, signal separation and demodulation can be computed as follows,

$$C_{bits/sec/Hz} = log_2[1 + \frac{|\sum_{\gamma=1}^{N_s} \lambda_{ij}(\gamma) s_{ij}(\gamma)|^2}{|\hat{w}_i|^2}] \tag{61}$$

3.2.7.4.19 Proof of Theorem 1

Proof. To complete the proof, we leverage two well known results from Matrix Theory [53]. First, for any matrix A, the sum of the squares of the singular values is equal to the square of its Frobenius Norm such that $\sum_i \lambda_i^2(A) = \|A\|_F^2$ where $\lambda_i(A)$ is the i^{th} singular value of A and $\|A\|_F$ is the Frobenius Norm of A. This follows from the decomposition of matrix A as a singular value decomposition, $A = U\Lambda V^H$. The Frobenius norm is invariant under orthogonal transformation of the left and right orthogonal matrices. Therefore, based on the fact that $\lambda(A)$ is the component of A along the diagonal of Λ, the sum of squares of the components of Λ equals the square of the Frobenius norm.

Second, the Frobenius Norm of the product of matrices is upper bounded by the product of the Frobenius Norms of the individual matrices such that $\|AB\|_F \leq \|A\|_F \|B\|_F$. The proof of this is based on the well-known Cauchy Schwarz Inequality.

From Equation (57), $\Gamma_{ij} = U_{ij}^H U_i^H H_{ij}$ and $\lambda_{ij}(1) \cdots \lambda_{ij}(N_s)$ are the singular values of Γ_{ij}. Combining the above two statements from [53],

$$\begin{aligned}
\sum_{\gamma=1}^{N_s} |\lambda_{ij}(\gamma)|^2 &= \|\Gamma_{ij}\|_F^2 \\
&\leq \|U_{ij}^H\|_F^2 \|U_i^H\|_F^2 \|H_{ij}\|_F^2 \\
&\leq \sum |\lambda(U_{ij}^H)|^2 \sum |\lambda(U_i^H)|^2 \sum |\lambda(H_{ij})|^2 \\
&\leq [M - \frac{N_I}{2} - (N_D - 1)][M - \frac{N_I}{2}] \cdot [M - \frac{N_I}{2}] M \cdot M
\end{aligned} \tag{62}$$

(63)

□

where $\lambda(U_{ij}^H)$, $\lambda(U_i^H)$ and $\lambda(H_{ij})$ are the singular values of U_{ij}^H, U_i^H and H_{ij} respectively. It should be noted that $\sum_{\gamma=1}^{N_s} |\lambda_{ij}(\gamma)|^2$ is the gain for one stream between the j^{th} access point and the i^{th} mobile device. There are K^2 such streams. Therefore the multi-user capacity is given by,

$$C_{bits/sec} \leq BK^2 log_2(1 + [M - \frac{N_I}{2} - (N_D - 1)][M - \frac{N_I}{2}]^2 M^2 \times SINR) \tag{64}$$

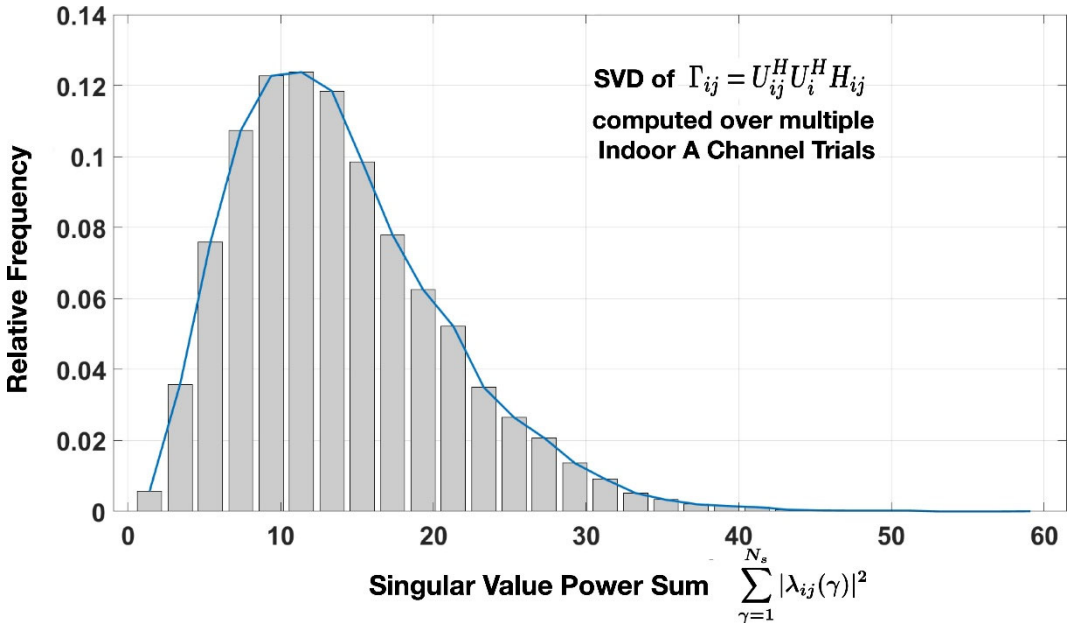


Figure 128: Relative frequency distribution of the singular value power sum for K 3. The upper bound for this sum is given by equation (63).

3.2.7.4.20 Verification of capacity upperbound To verify the capacity upperbound for K = 3, several simulation trials were run in which Rayleigh Fading Channels with the Indoor A power delay profile were generated. It should be noted that U_{ij} and U_i are normalized to have a power of 1 whereas H_{ij} has a power of 1 only in an expected sense. The values of $\sum|\lambda(U_{ij}^H)|^2$, $\sum|\lambda(U_i^H)|^2$ and $\sum|\lambda(H_{ij})|^2$ were found to be 3, 18 and 6 respectively which correspond to the dimensions of U_{ij}, U_i and H_{ij} respectively (Table 14).

The SVD in (57) is computed and a distribution of the sum of the singular value powers $\sum_{\gamma=1}^{N_s}|\lambda_{ij}(\gamma)|^2$ is plotted. This is shown in Figure 106. It can be clearly seen from the distribution that the values of the singular value power sum fall below the upper bound for K = 3 thus satisfying the theorem.

3.2.7.4.21 Channel estimation theory and Cramér-Rao Lower Bound In OFDM systems, for proper detection of symbols, channel estimation is performed by transmitting either known pilot symbols at certain frequency sub-carriers or across the entire OFDM symbol. In the K- User MIMO X framework described above, the pilot signals are set up such that only one antenna is transmitting in any given symbol time, during the estimation phase. So the MIMO channel estimation can be broken down into several single antenna channel estimations. To that end, in this section we summarize some of the well known theory on estimating a channel between a single transmit-receive antenna pair. We also present the mathematics for calculating the Cramér-Rao Lower Bound (CRLB).

Let $h = [h(0), h(1), \ldots, h(L-1)]^T$ represent the length L Channel Impulse Response (CIR). Let us denote the Channel Frequency Response (CFR) as $H = Fh$ where F is the $N \times N$ Fourier

Transform twiddle factor matrix where N is the Fourier Transform size.

Let us assume that an OFDM symbol X containing N_p pilots is transmitted where $N_p \leq N$. The received signal Y between a single transmit-receive antenna pair in the frequency domain is given by [54],

$$Y = diag(X)Fh + W \tag{65}$$

where W is the noise with variance σ_w^2.

Let F_p denote the $N_p \times L$ truncated Fourier matrix. The Maximum Likelihood estimate of the channel is given by [54],

$$h_{ML} = (F_p^H F_p)^{-1} F_p^H diag(X)^H Y \tag{66}$$

It is to be noted that h_{ML} is in the time domain. The corresponding frequency domain estimate H_{ML} can be obtained by taking the Fourier Transform.

The Minimum Mean Square Error estimate of the channel is given by [54],

$$H_{MMSE} = R_{HH_p}(R_{HH_p} + \sigma^2(diag(X)diag(X)^H)^{-1})^{-1} H_{ML} \tag{67}$$

where H_p is the channel frequency response at the pilot sub-carriers, R_{HH_p} is the cross-correlation between all the sub-carriers and the pilot-subcarriers and $R_{H_pH_p}$ is the auto-correlation between the pilot sub-carriers.

The CRLB is referenced from [55] in which it is defined as follows,

$$\sigma_{CRLB}^2 = \sigma^2 Tr\{D^{-1}\} \tag{68}$$

where $\frac{1}{\sigma^2}$ is the effective signal to noise ratio after the signal power has been normalized to 1. The operator $Tr\{.\}$ is the matrix trace. The matrix D, a function of the pilot locations is defined as $D_{n,k} = \sum_{m=0}^{N_p-1} e^{-j2\pi(n-k)i_m/N}$ $0 \leq n, k \leq L-1$ where $\{i_m : 0 < m < N_p - 1\}$ represents the indices of the pilot locations.

3.2.7.4.22 Research 2: On the Application of Key-Based Physical Layer Security in 5G Heterogeneous Networks The 5G wireless standard is set to unlock several new applications such as smart cities, autonomous vehicle control, industrial automation, drone coordination and control all of which require seamless connectivity [56] between a large number of devices. This diversity of devices and services has highlighted the need for improved capacity and reliability while at the same time providing robust and low latency security solutions for protection against malicious agents such as eavesdroppers. Several security applications have been proposed in the past [57, 58] most of which involve Layer 4 (Transport) and Layer 3 (Network) ciphering, access control, and other associated security policies applied along with tunneling or port forwarding [59]. Incorporating security at the higher layers requires that received data be sampled, demodulated and propagated through the layers of the Open Systems Interconnect (OSI) stack before applying the decryption process, leading to more latency. We integrate 5G heterogeneous networks, a key enabler for increased capacity, with enhanced security protocols at the Physical Layer to provide privacy at low delay.

Unlike traditional cipher-based privacy algorithms, Physical Layer Security applies privacy protocols at Layer-1 of the OSI stack as shown in 107. It is an alternate paradigm of security which uses imperfections in the wireless medium such as fading or noise to hide decryption information [60, 61] and provides security against eavesdroppers. There are two broad forms of Physical Layer Security techniques: dissemination and key extraction. Dissemination techniques aim to create better reception conditions at the intended receiver than the eavesdropper, either naturally or by using artificial noise [62, 63]. The gap in SNR can be used to convey secret information. The performance of these techniques is usually expressed in terms of the secrecy rate which is the rate difference between the intended channel and the eavesdropper's channel. On the other hand, key extraction based systems use information from the unique transmitter-receiver channel as the secret key for a higher-level encryption algorithm. These techniques primarily rely upon differences in the channel rather than SNR and cannot be overcome simply by placing the eavesdropper closer to the transmitter. key based techniques include Channel Quantization and Precoder Matrix Indexing (PMI), among others [64]. We provide 5G extensions to key based Physical Layer Security by leveraging Precoder Matrix Indexing.

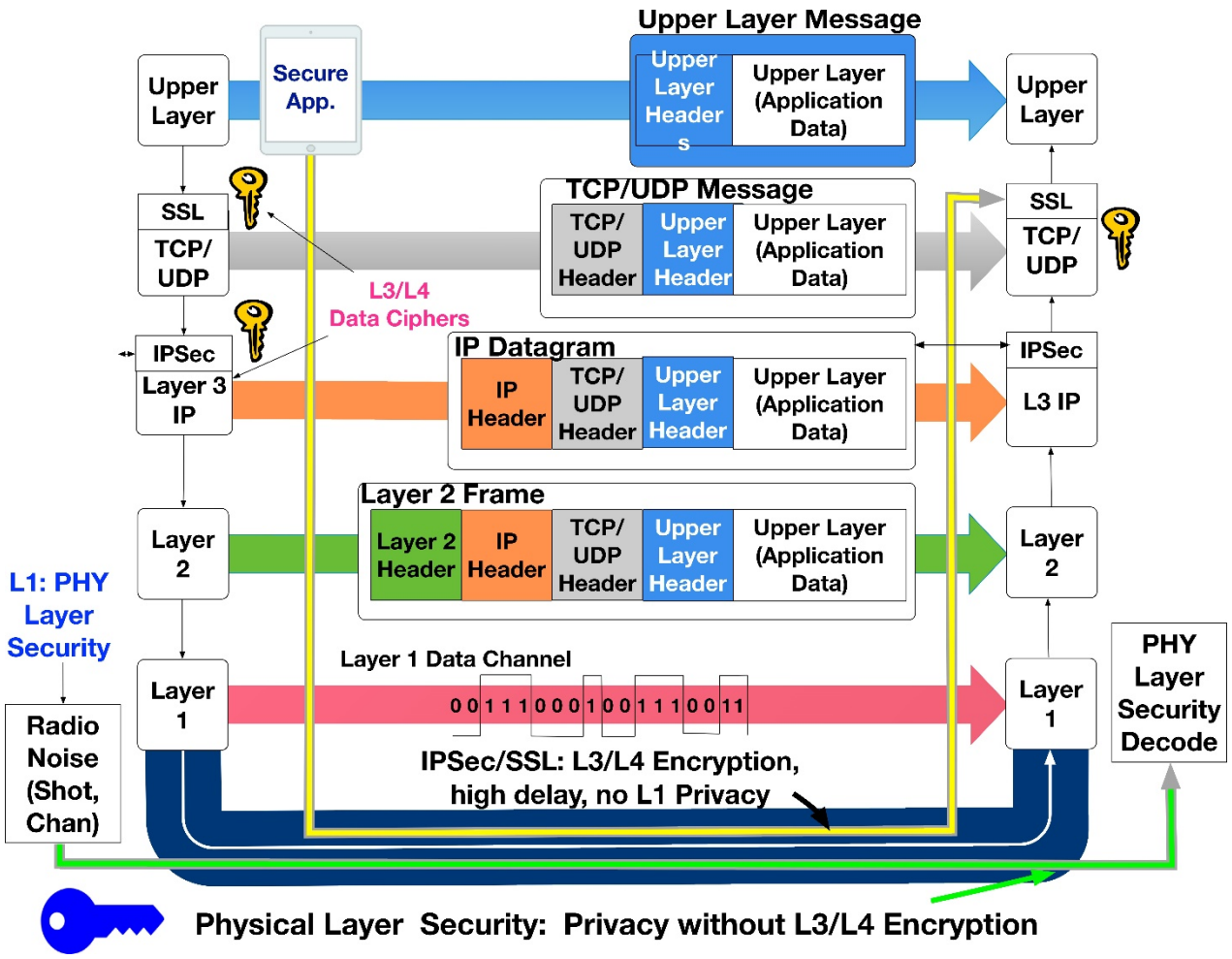


Figure 129: OSI 5-Layer model showing Physical Layer Security for privacy without higher layer ciphers.

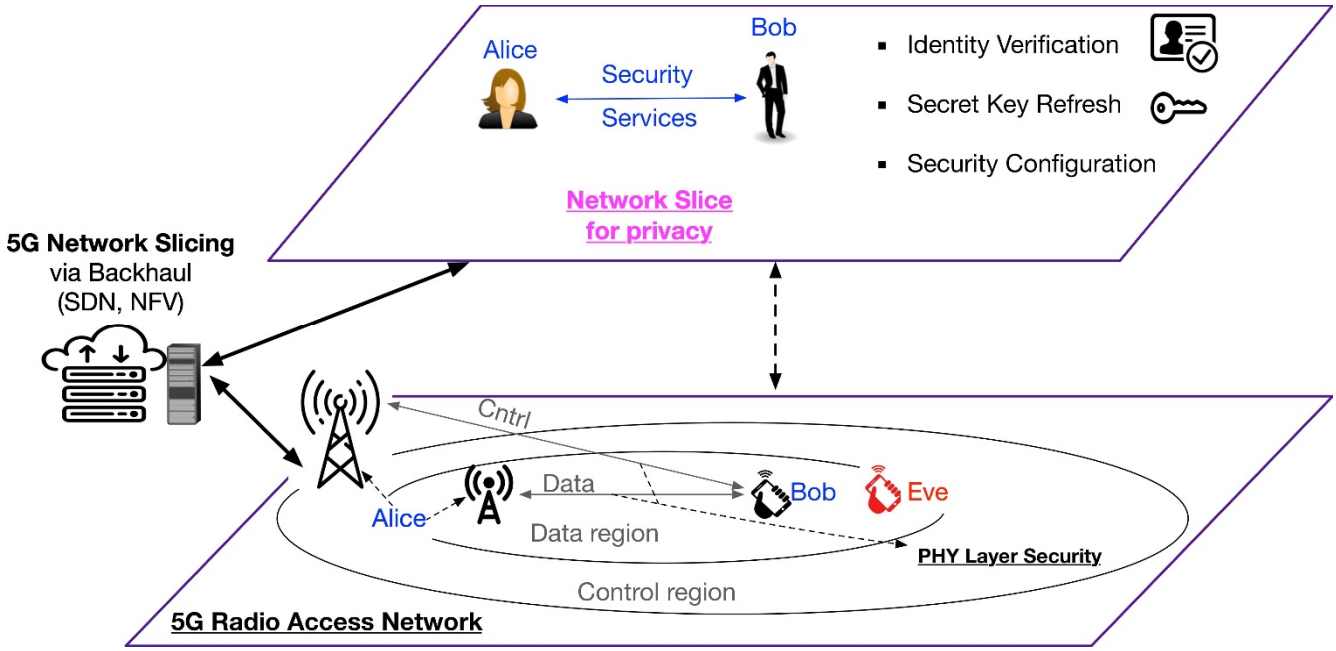


Figure 130: System model showing a new mode for securing 5G communication through a secrecy plane. [68]

3.2.7.4.23 Review of PMI schemes Precoder Matrix Indexing (PMI) schemes convey secret information by mapping secret keys to different transmit precoders which can then be de- tected at the receiver if the channel is known. The secret key is used as an index to a precoder in a universally known codebook. The secret bits themselves never leave the transmitter but the mapped precoders can be decoded by the receiver to recover the codebook indices. Secret trans- mission requires that the channel between the legitimate transmitter and receiver be unknown to the eavesdropper. A scheme presented in [65] involves shuffling of codebook indices based on the channel capacity so that even if Eve can detect the transmit precoder, no information is extracted. This method however would require large codebook sizes to prevent brute force iteration of the codebook indices at Eve. In [66] a random rotational matrix and Singular Value Decomposition (SVD) is used to send arbitrary precoders between Bob and Alice. In [67], a scheme that effectively creates a fast-fading transmitter-eavesdropper channel and protects against blind estimation based decryption is described.

3.2.7.4.24 Application of Physical Layer Security to Cellular Networks Several efforts to integrate dissemination type Physical Layer Security into cellular networks have been studied. A probabilistic secrecy rate analysis is provided in [69] for cellular networks with Poisson Point Process models for the location of nodes. The work in [70] deals with artificial noise based physical layer security in millimeter wave cellular networks. Another approach which combines stochastic geometry with the analysis of average secrecy rate in relation to eavesdropper densities is shown in [71]. Lastly, [72] presents closed form expressions for secrecy outage probability as a function of fading, location and number of interferers. However, there have been few attempts to apply key

extraction techniques to cellular networks apart from our previous works [73, 74] which have investigated key error rate based network planning for cellular environments with hexagonal geometries. We extend our previous work on Key based Physical Layer Security to more realistic stochastic network geometries.

3.2.7.4.25 System Model

3.2.7.4.26 Secrecy Plane through 5G Network Slicing 5G is expected to support bil- lions of connected devices in the upcoming future. To support this, a single network infrastructure must be capable of meeting a diverse set of service specifications. A key feature proposed for 5G is the separation of information on two independent logical planes- the control plane and the data plane [27]. In typical 5G heterogeneous networks it is envisioned that the control plane will facilitate the broadcast of system information (Radio Resource Management) at the macro cell level. The small cells can handle lower footprints so they will be mainly tasked with data offload.

108 shows this separation which is referred to as Network Slicing. It is defined in [75] as the ability to create multiple logical slice networks on one physical network, where each slice can individually be managed and configured depending on different needs related to latency, privacy etc. In 3GPP it is defined as "A logical network that provides specific network capabilities and network characteristics" [76] such as transmission of data or control or the provision of privacy. To support a large number of devices, at the network infrastructure level, Software Defined Networking (SDN) and Network Function Virtualization (NFV) need to be applied to achieve flexibility and programmability. Both these techniques will allow the creation of multiple virtual networks sharing a single physical resource. In the backhaul network, SDN and NFV enable the logical separation of functions onto different planes or slices [75].

The current security methods for 5G involves the programming of a secret key during installation which is then used as a cipher for a higher level encryption algorithm. However, to incorporate physical layer security methods we propose a new mode for communication networks that adds a third plane - the secrecy plane - which can be used for secure exchange involving authentication, transmission and refresh of secret keys and other sensitive network information. The Secrecy plane represents a logical network infrastructure for physical layer security protocols such as the one investigated here. The secrecy plane forms a network slice just like the data and control planes. This plane, shown in 108 would encompass the security of both the control plane and the data plane at the physical layer.

3.2.7.4.27 Geometry Model The proposed system model considers a stochastic geometry framework in which the base stations are spatially distributed according to a two-dimensional homogeneous Poisson Point Processes (PPP). This framework is more realistic and more in line with 5G Heterogeneous Networks than the hexagonal geometry. This is because the locations of nodes are more likely to be random in nature due to the size and unpredictability. The base station PPP is denoted by ϕ_{BS} and has density λ_{BS}. Given an area A, the number of base stations are drawn according to a Poisson Process with parameter $A\lambda_{BS}$. The nodes themselves are then uniformly placed in the two-dimensional space. We analyze the performance at a target user located at the origin. In the context of stochastic geometry, a cell is defined as the Voronoi Tessellation formed by the PPP ϕ_{BS}. This means that users in a Voronoi Cell are closest to the base station in that cell and are therefore assumed to be served by it. Consequently, we make the assumption that the

target user placed at the origin connects to the base station closest to it. Once the target base station is found, every other base station is a source of co-channel interference at the target.

The target base station is assumed to be Alice and the target user is assumed to be Bob. Slivnyak's theorem states that performance metrics obtained at the target placed at the origin hold good for any generic receiver [77]. Communication between Alice and Bob is required to be protected against eavesdroppers by applying Physical Layer Security. The eavesdropper Eve, is assumed to be passive and can be located anywhere within the target Voronoi cell. A sample realization of the geometry is shown in 109.

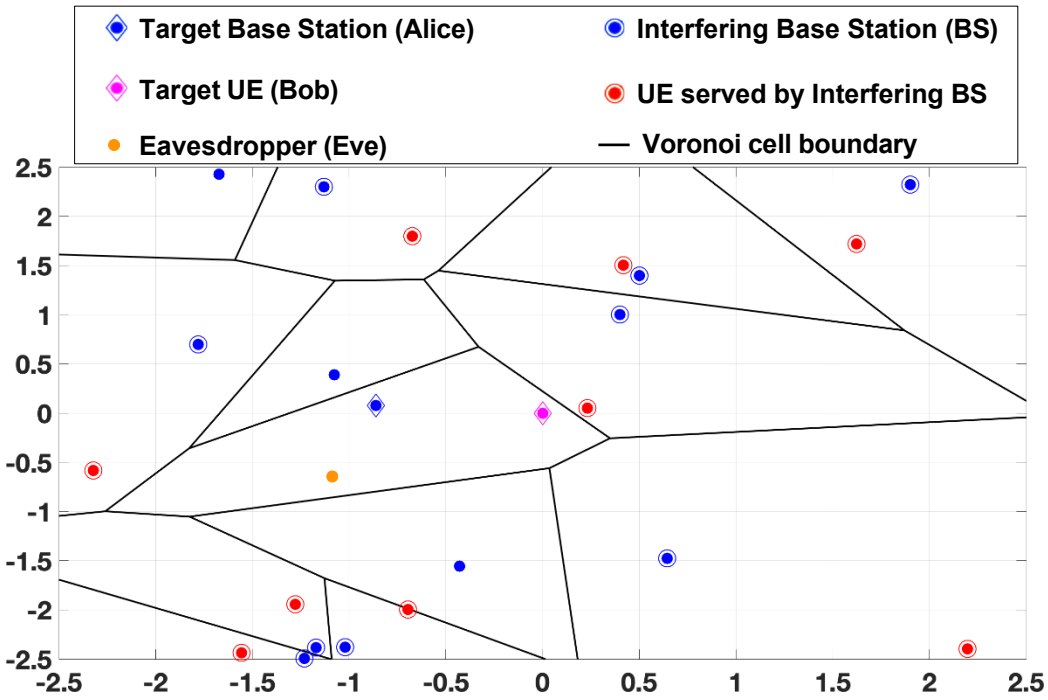


Figure 131: An example realization of the system geometry based on a Poisson Point Process in a 5 x 5 sq. km area

3.2.7.4.28 Signal Model Both Alice and Bob have MIMO capability with N_A and N_B antennas respectively. The signal model used consists of the transmission of precoded reference signals between Alice and Bob and vice versa. The channels between the two are assumed to be Time Division Duplex (TDD) channels, denoted by H_{AB} and H_{BA}. Perfect channel reciprocity is assumed, so that $H_{BA} = H_{AB}^{T}$.) The system uses a codebook which is assumed to be known universally. The codebook contains the precoders each of which has an index called the Precoder Matrix Index (PMI). The binary equivalent of the PMI is used as the secret key. It can be seen that a k bit codebook contains 2^k precoder matrices. We used the DFT codebook presented in [78].

3.2.7.5 Review of the 5G Extended MOPRO Scheme

In this section, a PMI based physical layer security scheme called MIMO-OFDM Precoding matrix index based scheme with ROtation matrix (MOPRO [66]) is described. We add 5G extensions to the work in [66] including 5G reference signals and heterogeneous networks. The steps of the scheme [66], are shown below and also in 110 and 111. The scheme assumes an OFDM system in which the secret key is exchanged between Alice and Bob through precoded reference signals in each sub-band. 5G NR does not require cell specific reference signals but configuration of cell parameters to match LTE reference signals is possible [79]. With respect to MOPRO, it should be noted that the reference signals r_1 and r_2 shown in 111 are on different sub-bands. The scheme also relies upon the channel being correlated for at least 3 symbol times.

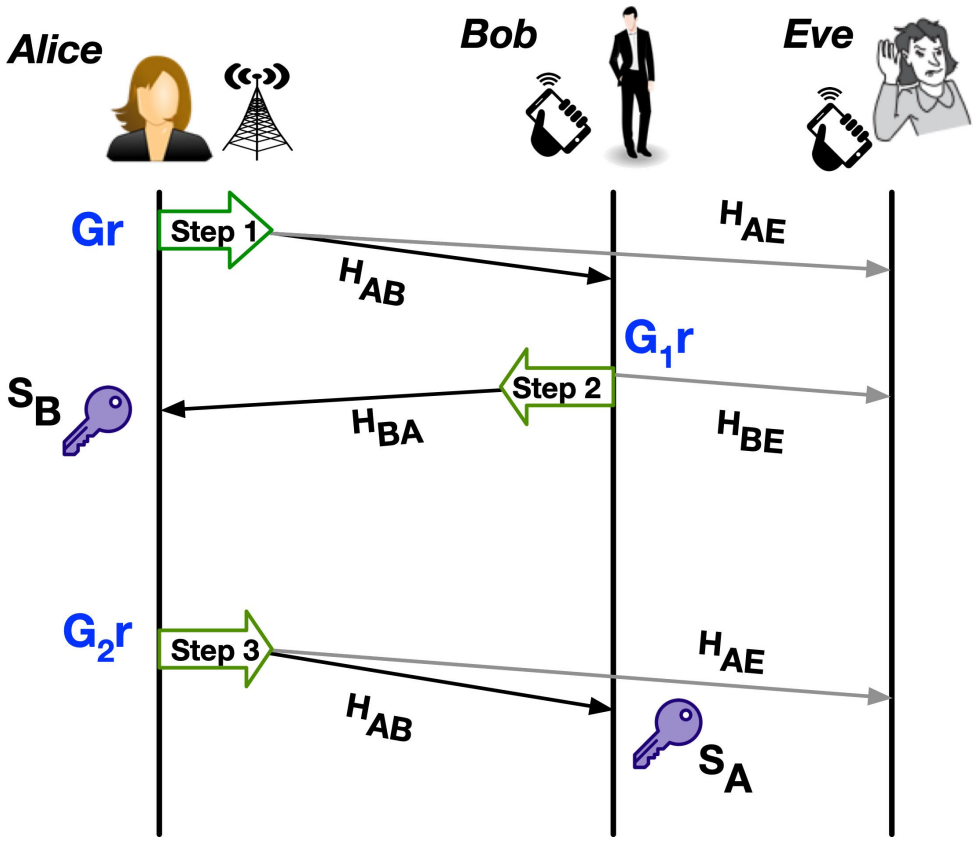


Figure 132: 5G Extended MOPRO signalling time schedule showing transmission of rotated refer- ence signals.

Step 1: Alice to Bob

1. Alice first transmits a reference signal $\mathbf{r}$, rotated by a random unitary matrix $\mathbf{G}_i$ in sub-band i where $\mathbf{r} \in C^{N_A \times 1}$ and $\mathbf{G}_i \in C^{N_A \times N_A}$. Bob then estimates the channel and obtains $\mathbf{H}_{AB,i}\mathbf{G}_i$ for sub-band i and performs the following SVD, $\mathbf{H}_{AB,i}\mathbf{G}_i = \mathbf{U}_{B,i}\mathbf{\Sigma}_{B,i}(\mathbf{V}^{\dagger}_{B,i}\mathbf{G}_i)$

Step 2: Bob to Alice

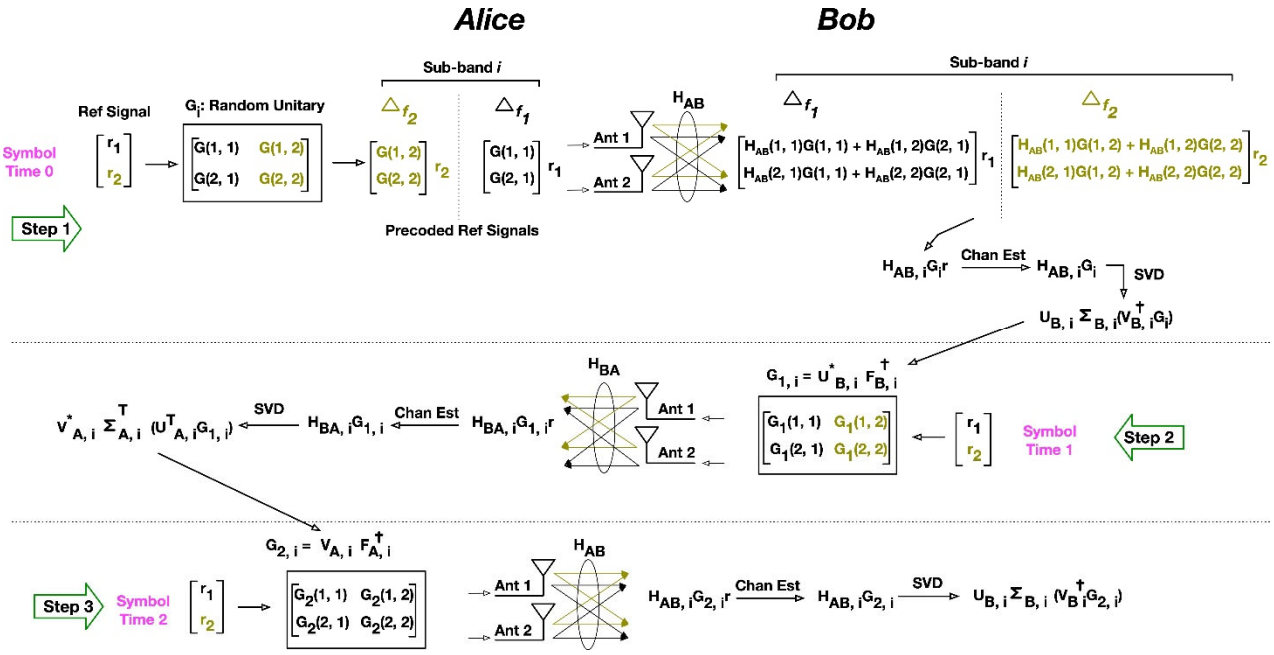


Figure 133: Signalling Procedure of the 5G Extended MOPRO compatible with LTE like OFDM- reference signals. In each step, the reference signals r1 and r2 are on different frequency sub-bands.

1. Bob generates a p bit random secret key, $\mathbf{S}_B$ and optionally applies channel coding to it to generate $\mathbf{C}_B$.

2. If a k bit codebook is used, $\mathbf{C}_B$ is divided into $N_s = \lceil *^{p}_{k} \rceil$ groups of k bit sequences. The total number of sub-bands, N_s must be equal to $\lceil *^{p}_{k} \rceil$ i is denoted by $\mathbf{C}_{B,i}$. . The k bit sequence in sub-band

3. For each sub-band i, Bob looks up $\mathbf{C}_{B,i}$ in the codebook indices and finds the corresponding precoder $\mathbf{F}_{B,i}$.

4. Bob then transmits a rotated reference signal $\mathbf{G}_{1,i}\mathbf{r}$ to Alice, where $\mathbf{G}_{1,i} = \mathbf{U}^{*}_{B,i}\mathbf{F}^{\dagger}_{B,I}$

5. Alice estimates $\mathbf{H}_{BA,i}\mathbf{G}_{1,i}$ and performs an SVD, such that $\mathbf{H}_{BA,i}\mathbf{G}_{1,i} = \mathbf{V}^{*}_{A,i}\Sigma^{T}_{i}(\mathbf{U}^{T}_{A,i}\mathbf{G}_{1,i})$

   Then she looks up the right singular matrix in the codebook and estimates the PMI. Alice repeats this for all sub-bands and reconstructs $\mathbf{C}_B$. If channel coding is applied, Alice decodes it to obtain $\mathbf{S}_B$.

Step 3: Alice to Bob

1. Next, Alice generates her own p bit random secret key, $\mathbf{S}_A$ and optionally applies channel coding to generate $\mathbf{C}_A$. The k bit sequence for each sub-band i is denoted by $\mathbf{C}_{A,i}$.

2. For each sub-band i, Alice looks up $\mathbf{C}_{A,i}$ in the codebook and finds the corresponding precoder $\mathbf{F}_{A,i}$.

3. Alice then transmits a rotated reference signal $\mathbf{G}_{2,i}\mathbf{r}$ to Bob, where $\mathbf{G}_{2,i} = \mathbf{V}_{A,i}\mathbf{F}^{\dagger}_{A,I}$

4. Bob estimates $\mathbf{H}_{AB,i}\mathbf{G}_{2,i}$ and performs an SVD. Then he looks up the right singular matrix in the codebook and estimates the PMI. Bob repeats this for all sub-bands and reconstructs $\mathbf{C}_A$. If channel coding is applied, Bob decodes it to obtain $\mathbf{S}_A$.

The secret key of the entire system is the concatenation of $\mathbf{S}_A$ and $\mathbf{S}_B$. Both Alice and Bob have half of the key which they generated themselves and the other half which they estimated in the form of the PMIs.

3.2.7.5.1 Security Properties of 5G Extended MOPRO It is worth analyzing exactly how the 5G Extended MOPRO scheme prevents the eavesdropper from intercepting the secret key. Firstly, the use of the random unitary matrix in step 1, effectively generates uniformly distributed secret keys. As stated in [66], the security guarantee comes from the fact that the secret key is split into two parts. For Eve to obtain the secret key, she needs to obtain both $\mathbf{S}_A$ and $\mathbf{S}_B$. Eve can estimate $\mathbf{H}_{AE}\mathbf{G}$, $\mathbf{H}_{BE}\mathbf{G}_1$ and $\mathbf{H}_{AE}\mathbf{G}_2$. If Eve is far away from both Alice and Bob, both $\mathbf{H}_{BE}\mathbf{G}_1$ and $\mathbf{H}_{AE}\mathbf{G}_2$ will be completely different from $\mathbf{H}_{BA}\mathbf{G}_1$ and $\mathbf{H}_{AB}\mathbf{G}_2$. Therefore Eve will not be able to ascertain any information about the secret keys.

In the case where Eve's location is close to that of Alice, her estimated $\mathbf{H}_{BE}\mathbf{G}_1$ may be similar to the actual $\mathbf{H}_{BA}\mathbf{G}_1$. In this case Eve perhaps might be able to acquire $\mathbf{S}_B$. But this is insufficient since she is still missing the other half of the key $\mathbf{S}_A$, which has been generated by Alice and never explicitly transmitted. A similar argument can be made for the case where Eve is near Bob and $\mathbf{H}_{AE}\mathbf{G}_2$ may be similar to the actual $\mathbf{H}_{AB}\mathbf{G}_2$. In this case Eve can possibly obtain $\mathbf{S}_A$ but not $\mathbf{S}_B$ since this part of the key has never left Bob.

3.2.7.5.2 Precoder detection for key recovery Let us consider a DFT codebook containing N_m precoder matrices of size $N_A \times N_A (= N_B \times N_B)$. Further, let us assume that a certain transmission uses a precoder matrix F_t ($t \in \{1, 2, \cdots, N_m\}$) from the codebook. The detected pre- coder at the receiver, F_r is a perturbed version of F_t as shown in (69). The perturbation is a result of both Additive White Gaussian noise and interference which is treated as noise.

$$F_r = F_t + \Theta \tag{69}$$

The precoder detection process uses the minimum distance operator χ_{opt} as shown below,

$$\chi_{opt} = min(\chi_1, \chi_2, \ldots, \chi_{N_m}) \tag{70}$$

where χ_n ($n = 1, 2, \cdots, N_m$) is the distance between the received precoder matrix and each element of the codebook. The detection procedure is based on the unitary property of the precoder matrices in the DFT codebook as shown in (71).

$$\chi_n = \|F_r F_n^{\dagger} - I\| \tag{71}$$

where $r \in \{1, 2, \cdots, N_m\}$ and $n = 1, 2, \cdots, N_m$ and the operator $\|\ \|$ is the Frobenius Norm. By combining, (69) and (71) we have,

$$\chi_n = \|(F_t + \Theta)F_n^{\dagger} - I\| = \|F_t F_n^{\dagger} - I + \Theta F_n^{\dagger}\| \tag{72}$$

where $t \in \{1, 2, \cdots, N_m\}$ and $n = 1, 2, \cdots, N_m$. The precoder index n that satisfies (70) is the estimated PMI. Once the PMI has been obtained for all the sub-bands, the estimated bits of the secret key

can be obtained. In general the Key bit Error Rate is defined as the probability that any bit in the transmitted key is not the same as in the estimated key. However there is a modification that is specific to MOPRO, which adds a scale factor of 0.5 to the Key bit Error Rate calculated as defined above. This is due to the fact that the secret key is the concatenation of both S_A and S_B but the errors are only in either S_A or S_B depending on the MOPRO stage. Recall that Alice generates S_A herself and estimates S_B. So the errors are only in S_B. Similar is the case for Bob.

3.2.7.6 Simulation procedure and results

In this section we present simulation results for the Key Bit Error Rate of the 5G Extended MOPRO scheme under both interference and noise limited scenarios in a stochastic geometry environment. We consider an area A = 5km × 5km. For a given base station density λ_{BS}, multiple trials are

Table 17: Simulation Parameters

Channel Model	Rayleigh Fading
MIMO System	2 × 2
Subchannel bandwidth	15 kHz (Δ_f)
Channel bandwidth	20 MHz
Number of subcarriers	1320
Codebook	DFT Codebook [78]
Transmit Power	46 dBm
Rx Noise Figure	4 dB
Thermal Noise Density	−173.8 dBm/Hz
Path Loss Exponent	3.5

performed to reduce the effect of noisy data. The number of base stations in a trial is drawn from the corresponding Poisson Process and the nodes are placed uniformly.

The target user at the origin connects to its closest base station. The target user is assumed to be Bob and the closest base station is Alice. Each of the base stations apart from the serving base station is counted as a co channel interferer. The aim of the simulation is to obtain the Key Bit Error Rate performance at the target Alice Bob pair in the presence of other interfering base stations using the same time-frequency resources.

Assuming that each base station (Alice) has the same transmit power of 46 dBm, all three steps of the security scheme are performed over 1320 subcarriers in the presence of both noise and interference. A free space exponential path loss model, $d^{-\alpha}$ is used, where d is the distance and α is the path loss exponent. The Key Bit Error Rate across trials is calculated and plotted for various values of base station densities. The simulation parameters are shown in Table 17.

In a small, finite area such as 25 sq.km, lower densities exhibit better Key Bit Error Rate performance simply on account of the fact that lower densities result in less interfering base stations on average. For λ_{BS} between $0.01km^{-2}$ and $0.1km^{-2}$, the average number of interferers is rather low so this leads to the best performance in Key Bit Error rate as indicated by 112. Further it can be seen that using a higher codebook size while improving security leads to higher key error rates. 113 shows the cumulative distribution functions of the Key Bit Error Rates for both 1 and 2 bit

codebooks at various base station densities. A reasonable threshold of 1% is met by both 1 and 2 bit codebooks for λ_{BS} = 0.01. λ_{BS} = 0.1 achieves close to 3%.

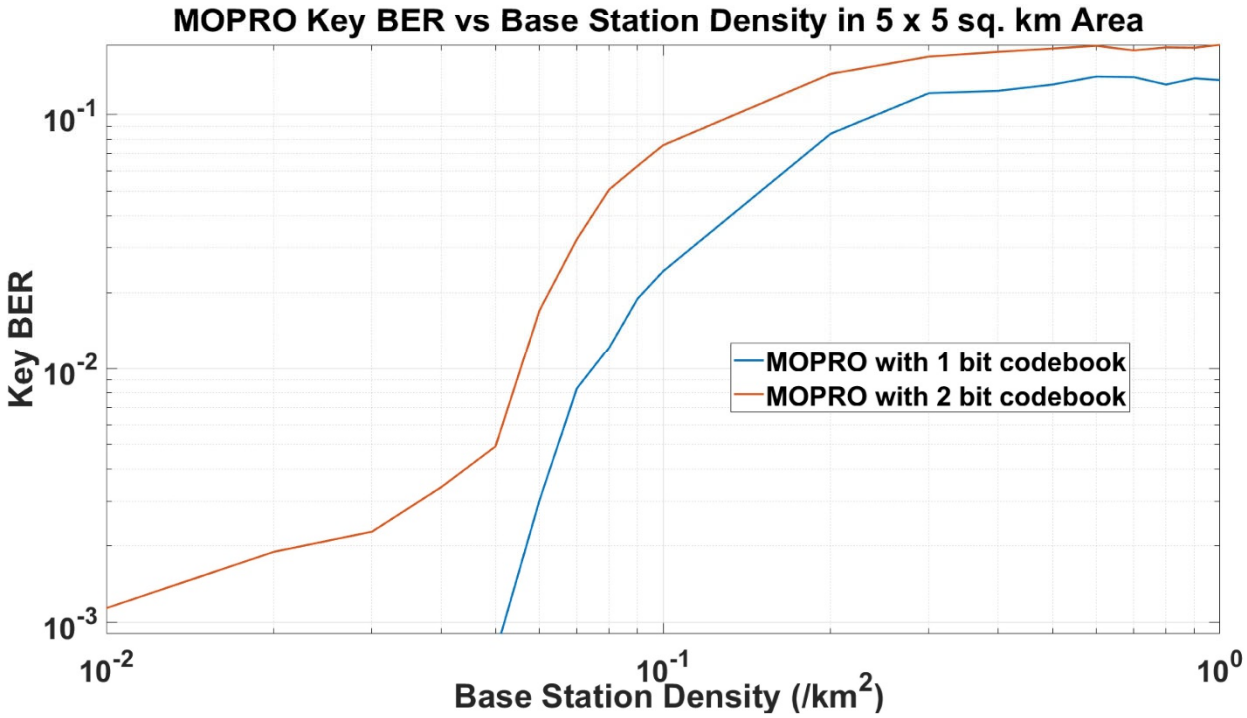


Figure 134: Key Bit Error Rate vs Base Station density

There are some additional considerations which could form a part of future work. The first is the inclusion of multiple eavesdroppers. The second is the case where there are multiple receivers with which secret keys need to be exchanged. In this case, network scheduling algorithms based on channel conditions can be applied. These include Proportional Fair algorithms or greedy algorithms which minimize Key Bit Error Rate.

3.2.7.6.1 Conclusion We proposed the use of network slicing to incorporate a new secrecy plane in 5G networks. We have investigated Key Based Physical Layer Security as a component of the Secrecy plane. Under stochastic geometry frameworks, the Key Bit Error rate performance was analyzed to obtain optimum base station densities for 5G Heterogeneous Network design. The results indicate a way forward for 5G dynamic key refresh and 5G network slicing for secret information exchange.

3.2.7.7 Technical Accomplishments in Task T2-7

Describe the technical accomplishments made during the course of the project.

T2-7 accomplishments were the innovative use of relaxed constraints to enable high throughput through K-User MIMO, seen in [42], [80], and used for IoT ecosystems in [81]. K-User MIMO facilitates all-to-all communication between K access points and K mobile devices. For such a

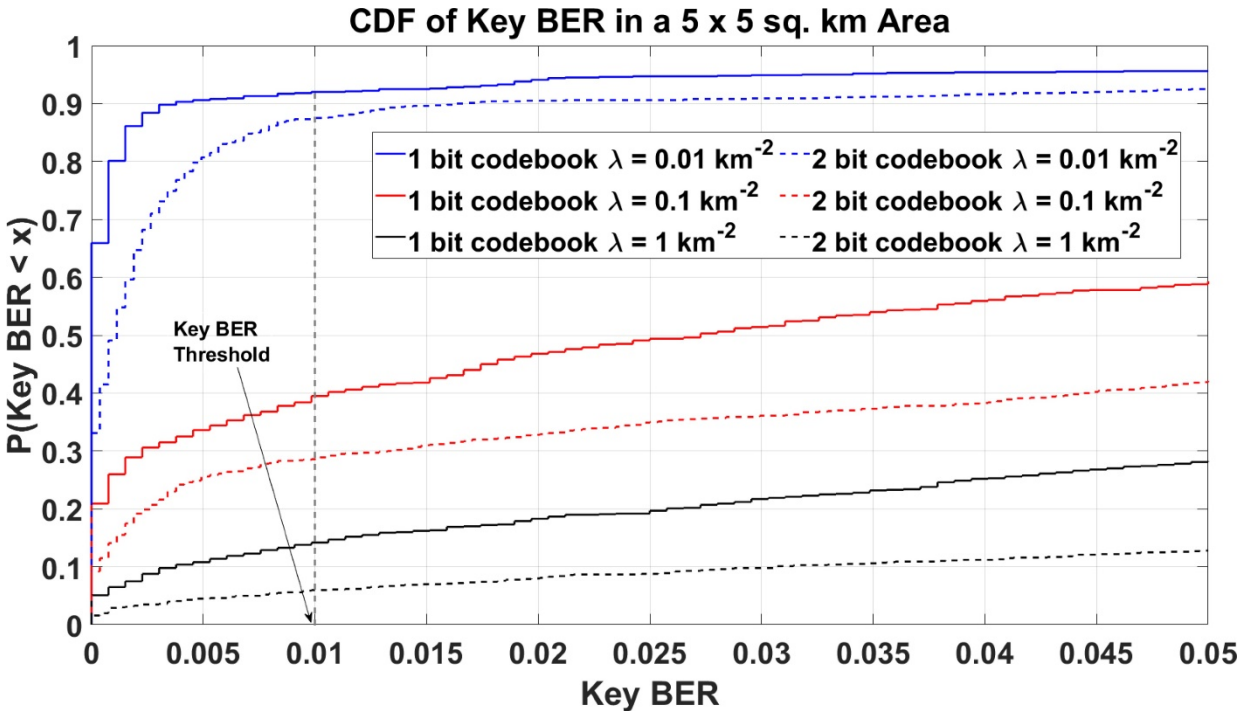


Figure 135: CDF of Key Bit Error Rate for various Base Station densities (λBS) and codebook sizes.

network, T2-7 illustrated the demodulation of K^2 independent data streams through a new interference cancellation beamforming algorithm that improves spectral efficiency compared to massive MIMO. Research shown in [80] derives a multi-user Shannon Capacity formula for K-User MIMO for $K \geq 3$. [80] defined an Orthogonal Frequency Division Multiplexing (OFDM) frame structure that demonstrates the allocation of time-frequency resources to pilot signals for channel estimation. T2-7 determined a practical upper bound for MIMO array sizes that balance estimation overhead and throughput. The simulation results show the practical capacity in small cell geometries under Rayleigh Fading conditions, with both perfect and realistic channel estimation, [80].

T2-7 also enabled the use of an entirely novel version of Physical Layer Security by creating protocols that map random secret key bits to precoders in a codebook through a Precoder Matrix Index, [82], [83]. This research introduced a scheme for extending 5G with Physical Layer Security, an emerging Open Systems Interconnect Layer-1 security area that achieves perfect secrecy data transmission. This technical achievement sets the foundations for a Stochastic Geometry approach to Key-based Physical Layer Security establishes a paradigm for 5G Heterogeneous Networks. T2-7 utilized Time Division Duplex algorithms with reference signals in a 5G-OFDM framework to enable shared, private information. The Physical Layer Security algorithm maps random secret key bits to a precoder in the Precoder Matrix Index (PMI) codebook. These new Layer-1 protocols, which are separate from higher-layer network and transport layer security, can decrease latency of the 5G-OFDM system and improve computational overhead [83].

Over the last few months, T2-7 has augmented the Physical Layer Security scheme in [82] and [83] to use machine learning to process received precoders in noisy and imperfect channel

environments at 99.49% accuracy in 10dB SNR environments. Results show that as we increase the antenna diversity, T2-7 shows that machine learning augmentation can lead to exemplary results for low-SNR environments.

In summary, task 2-7 has accomplished the following:

- Developed a more feasible implementation of K-User MIMO for $K \geq 3$.
- Innovated a new form of Physical Layer Security that provides perfect privacy that allows for low-latency, low-computation privacy.
- Augmented the new Physical Layer Security algorithm to make use of machine learning methods to improve detection accuracy in low SNR environments.

3.3 Project Progress and Project Plans for Thrust 3: Testing, Evaluations and Verification of Large-scale Autonomous Vehicles (TEVLAV)

3.3.1 Project Progress for Task T3-1 (Motion planning and control framework for coordinated collective motion of robotic swarms)

3.3.1.1 Period of Performance under Task T3-1

Start Date: January 2018

Conclusion Date: December 2021

Faculty lead: Dr. Ali Karimoddini

3.3.1.2 General Description of Task T3-1

A robotic swarm is a group of many robots in which a desired collective behavior emerges from local interactions among robots and between the robots and the environment. In robotic swarms, the individual member robots are simple in design and small in size. Relatively small robots can navigate narrow spaces, are disposable and economical, and are safe in human-robot collaborations. For example, a swarm of small quad-rotors can collect aerial imagery quickly and cheaply. Moreover, whereas individual members may be inherently limited in their sensing, communication, and performance capabilities, as a group, robotic swarms may solve complex problems collectively even when member robots do not have access to centralized control or global knowledge. Therefore, in recent years, robotic swarms have been suggested as a convenient means to perform tasks that are too risky for humans, or where fast response is crucial, and tasks that are beyond the capabilities of a single or few individual robots. For example, a swarm of drones could be instructed to spray pesticides, pollinate a field, and monitor crops. Inspired by termites, heterogeneous robotic swarms could be programmed to search and rescue people buried under snow avalanches or trapped in wildfires, pile sandbags along coastlines vulnerable to flooding, or layout a fence around fatal radio-active spills. There have been impressive light shows with a swarm of drones, such as Intel's Shooting Star drones, contracted in the 2017 Super Bowl halftime and the 2018 Winter Olympics. Robotic swarms can also be employed in search and rescue operations, mining, area explorations, planetary explorations, industrial machinery inspections , environmental monitoring, mapping, distributed manipulation, collective transportation, surveillance, and reconnaissance.

The ability of robotic swarms to execute complex tasks with enhanced adaptability and robustness stems from the core principle of the swarm of robotics. The core idea behind swarm robotics is that individual robots run local control algorithms, merely based on local communication and local sensing inputs such that a desired collective (global) behavior emerges from local interactions among the neighboring robots and between the robots and the environment. Moreover, this principle empowers individual robots with limited actuation and limited computational power to solve complex problems cooperatively without relying on any centralized observer. Not relying on a centralized observer promises robustness of the robotic swarm to member robot drop-outs, as a single(some) failing robot(s) will not peril task execution because its(their) role(s) will be fulfilled by one(some) of the many other robots in the swarm.

Many robotic swarm applications often involve a coordinated collective motion of a robotic swarm from the current location to other locations in space. For example, consider the problem of transporting large objects from location A to location B or the problem of carefully spraying pesticides and monitoring crops. Here, we drive or fly several robots from location A to location B while keeping the group together, following the desired path, and keeping the distance between each pair to not exceed a certain threshold. The process of collectively navigating a robotic swarm from the current location to other locations in space may require the swarm to rotate, translate, shrink, expand, and track the desired target (path) with enhanced adaptability, flexibility, and robustness to collaboratively complete tasks.

This project studies the coordinated collective motion of robotic swarms via abstraction. We propose a distributed swarm path planner, swarm pattern formation controller, and swarm time-varying formation controller to plan and control the motions of a robotic swarm through a relatively fewer set of meaningful parameters. Abstraction is made to capture the motion of the swarm using a set of few swarm descriptors or channeling the motion of the robotic swarm as the motion of a lower-dimensional system. More specifically, consider a scenario with hundreds of robots where a human operator would like to command the motions of these hundreds of robots to execute a given task. This can be very demanding and cumbersome if motions are to be specified individually for each robot. Moreover, since the original problem exists in the higher dimensional space, methods to reduce the dimension of the problem to a lower meaningful ones need to be devised. Hence, using abstraction techniques, we transform the motion of the robotic swarm into the motion of a lower-dimensional system. In general, to handle such and related scenarios, we design a swarm path planning and control framework to realize a flexible coordinated collective motion while also satisfying the following requirements.

- Robustness: the flexible coordinated motion algorithm does not depend on the ordering and identification of member robots. It copes well with the addition or loss of individuals.
- Scalability: the algorithm scales well with different group sizes. The introduction or removal of members does not drastically change the performance of the control algorithm.
- Distributed: the flexible coordinated collective motion algorithm performs well under the absence of a leader or a centralized controller.
- Flexibility: the flexible coordinated collective motion algorithm shall apply to a broad spectrum of tasks and environmental conditions.
- Correctness: the distributed algorithm shall converge to the correct value; that is, the robots shall reach consensus on the correct value.

3.3.1.3 Objectives for Task T3-1 over the Course of Project

- **Objective Name:** Design, development and analysis of motion planning and control framework for coordinated collective motion of robotic swarms.

  Objective Type: Research

  Objective Description: The motion planning and control framework shall be robust to individual robots drop-out, scalable, and distributed to be applicable to a broad spectrum of tasks and environmental conditions. To achieve these goals, the main cases of interest are :

  1 Swarm path planning: to plan an optimal path for a coordinated collective motion of a robotic swarm. The swarm path planning returns suitable swarm center positions, swarm rotational angles (roll, pitch, and yaw angles), swarm lengths, swarm widths, and swarm heights to move a robotic swarm from an initial configuration to the goal configuration.

  2 Swarm pattern formation: to form the desired robotic swarm shape suitable to execute a given task. The swarm pattern formation framework commands the robots to attain the desired shape before transferring the group from starting location to destination location.

  3 Swarm formation and trajectory tracking: to collectively drive a robotic swarm while rotating, shrinking, and expanding the group about the desired target (path) to collaboratively complete tasks.

  Impact: Many existing works fail to address the motion planning and control problem that respects the robustness, scalability, leaderless (distributed), and flexibility requirements. Addressing these challenges is essential to bring the developed swarming technologies from an innovation space in controlled environments to real-world reliable and large-scale operations. More precisely, the planning and control framework allows a human operator or a task planner to command the group of robots with few meaningful parameters, independent of a large number of robots in the swarm, without imposing unnecessary constraints such as rigidity constraints, robot ordering, and seeding, and robot labeling on the swarm. Also, the developed algorithms are easy-to-use, robust to robot drop-outs, robust to network failures, scalable to different swarm sizes, and perform correctly under the absence of a leader or a centralized controller/observer.

3.3.1.4 Progress Against Planned Objectives in Task T3-1

The objective of this task is accomplished. Several novel algorithms addressing the collective motion of robotic swarms in the presence of robot drop-outs, network failures, and limited communication bandwidth capacity are developed. The main accomplishment of this task are:

- A swarm path planning algorithm that plans the swarm center position, rotation angle, length, width, and height suitable for a coordinated collective motion. The proposed algorithm is based on a distributed RRT* algorithm on a radial configuration space with origin at the initial location of the swarm.

- A distributed swarm pattern formation algorithm that steers the robots to form the desired shape collectively and cooperatively. The swarm pattern formation appears in the first stage

of the coordinated collective motion of the robotic swarm. The convergence analysis of the proposed swarm pattern formation algorithm is also provided.

- Two distributed dynamic average consensus estimators that do not require special initialization. The proposed consensus estimators are robust to network failures, do not require the knowledge of derivatives of signals at each robot, do not exhibit chattering phenomena, and have zero steady-state errors. The consensus estimators are used to estimate the current value of the global behaviors of the swarm. The convergence analysis of the proposed dynamic average consensus estimators is also provided.

- A distributed time-varying formation control and tracking algorithm for robotic swarm with directed and undirected underlying communication topologies. The proposed algorithm does not require specifying formation parameters for individual robots in the swarm; rather, all the robots are given similar global specifications, independent of the number of robots in the group. Furthermore, a detailed stability analysis is carried out to determine suitable control parameters and develop the formation feasibility conditions.

- A distributed and scalable leaderless robotic swarm control framework that allows the group to navigate an environment with obstacles in a coordinated manner collectively. The framework allows the robotic swarm to track a given path while achieving the desired geometric pattern parametrized by the high-level swarm descriptors. The developed framework is also robust to member robot drop-outs and communication link failures among the robots. Furthermore, the detailed stability and converge analysis of the proposed coordinated collective motion algorithm is provided to determine suitable control parameters.

- A distributed and scalable swarm control framework for a swarm of robots with non-linear kinematics and dynamics communicating over a network with limited bandwidth capacity.

3.3.1.5 Technical Accomplishments in Task T3-1

In this task, to meet the mission requirements and enable scalable human-swarm interaction, we develop a hierarchical framework layered into various hierarchies. In a hierarchical control structure, the complexity of the design process can be distributed into different layers. In our proposed hierarchical swarm control framework, each layer abstracts away the details of other layers in the framework. A swarm pattern formation, path planning, time-varying formation tracking, and collision avoidance algorithms were embedded in the framework to safely arrange the team into the desired shape and navigate towards the desired destination collectively. More precisely, the swarm pattern formation first rearranges the swarm into the desired pattern using sorting and max-consensus algorithms. Then, the swarm time-varying formation tracking controls the swarm to track the desired trajectory of the swarm descriptors. The desired trajectory of the swarm descriptors is generated by the swarm path planning algorithm using a distributed RRT$*$ algorithm on the radial configuration space.

This task proposes a distributed hierarchical control framework for the swarm time-varying formation tracking, using tools from control theory, graph theory, and statistics. Based on local sensory input and information from local neighbors, the framework transfers a robotic swarm from the current location to the desired destination location while tracking the desired pose and shape (length, width, and height) trajectory. For this purpose, all the individual robots will be given the

global specification (desired center trajectory, orientation, length, width, and height of the swarm) from a swarm path planner. Our distributed hierarchical control architecture has two layers: the high-level (swarm) control and low-level (robot) control layers. In the high-level layer, we estimate the global properties of the swarm using a dynamic average consensus estimator and generate the desired trajectory for the low-level control layer. Our dynamic average consensus estimator does not require special initialization, does not suffer from chattering phenomena, and converges to an adjustable bounded error in a finite time. In the low-level control layer, we design a trajectory tracking control and collision avoidance control to achieve a desired coordinated collective motion while avoiding obstacles in the environment. Furthermore, the developed framework ensures scalable control computations, robustness to robot drop-outs, and communication link failures. The proposed control laws are independent of robot permutations and do not assign unique label identification to individual robots in the swarm. The proposed swarm control framework is easy to integrate with a global planner or interface with humans to provide specifications.

Next, we the coordinated collective motion of robotic swarms from global specifications. In many applications, tasks are assigned to the swarm in terms of reaching goals [84–87] and/or tracking a specified trajectory that capture the collective behaviors of the swarm [88–91]. Here, individual (local) robot controllers should be developed so that the robotic swarm performs a specified collective (global) behavior required to execute a given task effectively.

There are various control design approaches presented in the literature of swarm control algorithms (see, e.g., [92, 93] and reference therein), which include density-based [85], potential field [89, 94], optimization [88], behavior-based [84, 95, 96], consensus-based [97–99], leader-follower [100–103], and virtual structure control [98] methods, to name a few. The density-based, potential field-based and optimization approaches are mainly employed to solve the swarm pattern formation problem [85, 88, 94, 104]. Similarly, behavior-based approaches have been employed to design local behaviors for robots to perform swarm navigation to achieve the desired performance collectively. The work in [96] develops a decentralized behavior-based architecture, requiring fewer communications among the robots in the swarm. However, in general, behavior-based approaches are analytically challenging to establish proofs of their convergence.

On the other hand, in the leader-follower and virtual structure methods, the desired trajectory of the swarm are assigned to leader robot(s), virtual leader(s), or virtual structures. Olfati-saber [89] employed formation graphs to capture the robots' dynamics and inter-robot constraints, and then combined them with a potential field and virtual leader approach to drive a group of agents along a specified path. To improve the scalability of the swarm control algorithm, Belta et al. [90] proposed an abstraction-based control framework that drives a swarm of robots along a given path. However, the centralized architecture in [90] makes the design vulnerable to observer failures and communication link losses. Recently, Shiyu Zhao [91] presented a new approach based on stress matrices of graphs to achieve multi-agent formation maneuvers. The author adopted a distributed leader-follower approach to solve the formation maneuver control problem for a team of single-integrator, double-integrator, unicycle, and non-holonomic agents. However, the calculation stress matrices in [91] is nontrivial. Freeman et al. [105] designed a distributed estimation algorithm to estimate first-order and second-order moments of the swarm's distribution. They combined their estimation algorithm with motion controllers for each robot to regulate the shape and position of the swarm [106]. Nevertheless, the proposed PI estimator exhibits slow convergence rates, and the combined estimator/controller algorithm in [105] is limited to single integrator agents.

In this work, given a global specification (swarm formation and trajectory), we propose a scalable

and robust distributed control framework for synthesizing control laws for local (individual) robots so that they, as a group, can switch to any time-varying affine transformation of initial swarm formation while the swarm is tracking a desired bounded C^1 trajectory. For this purpose, we develop a distributed control algorithm for swarm formation control using feedback linearization and dynamic average consensus estimation. A salient feature of the proposed method is handling the losses or addition of robots from/to the swarm. This is due to the flexible and distributed architecture of the proposed framework versus the fixed and centralized network architecture in [90] where a communication loss with an observer results in a complete failure of the swarm system. Furthermore, many existing leader-follower based swarm algorithms require robot labeling [91, 102]. For example, in [102], a unique swarm leader and a unique tail robot are required for the swarm to navigate along the desired trajectory. However, our leaderless swarm formation control formulation do not require special robot ordering and labeling. More importantly, compared to swarm algorithms with a fixed inter-robot distance (see, e.g. [89]), our control design formulation allows the swarm to shrink, expand, rotate, translate, or perform compositions of these operations. All these features make our swarm control framework suitable for diverse applications.

3.3.1.5.1 Problem formulation Consider a swarm S of N identical rear-wheel driving car-like robots deployed to execute task T in a world-frame W (with center O_{W} and basis vectors $\{\boldsymbol{x}_{\mathrm{W}}, \boldsymbol{y}_{\mathrm{W}}\}$). The governing kinematics of Robot i are given by

$$\dot{x}_i = g_i(x_i)u_i, \qquad i = 1, \cdots, N, \tag{73}$$

where $x_i = [\bar{x}_i \quad \bar{y}_i \quad \theta_i \quad \phi_i]^{\mathrm{T}} \subseteq \mathrm{R}^4$ is the state vector, $u_i = [v_i \quad \omega_i]^{\mathrm{T}} \subseteq \mathrm{R}^2$ is the control input vector, $[\bar{x}_i \quad \bar{y}_i]^{\mathrm{T}}$ is the position vector, θ_i is the heading angle, ϕ_i is the steering angle, v_i is the linear velocity, ω_i is the steering velocity, and $g_i(x_i) = [g_{i1}(x_i) \quad g_{i2}(x_i)]$, where $g_{i1}(x_i) = [\cos\theta_i \quad \sin\theta_i \quad \frac{1}{L}\tan\phi_i \quad 0]^{\mathrm{T}}$, $g_{i2}(x_i) = [0 \quad 0 \quad 0 \quad 1]^{\mathrm{T}}$, and L is the wheel base of Robot i, respectively. Let $\mathrm{G}(t) = (\mathrm{V}(t), \mathrm{E}(t))$ be a time-varying communication graph of the swarm S at time t, where $\mathrm{V}(t)$ is the set of robots in the swarm and $\mathrm{E}(t) \subseteq \{(i, j) : i, j \in \mathrm{V}(t), i \neq j\}$ is the set of communication links among the robots in the swarm. The communication graph $\mathrm{G}(t)$ changes i) when new robots join the swarm; ii) when the swarm loses some member robots, and iii) when the communication links among the member robots fail.

Assumption 1. *The communication graph* $\mathrm{G}(t)$ *is assumed to be a slowly time-varying graph. Also, we assume that* $\mathrm{G}(t)$ *is a strongly connected bidirectional graph at each time t.*

Let the set of neighbors of Robot i at time t be given by $\mathrm{N}_i(t) = \{j \in \mathrm{V}(t) : (i, j) \in \mathrm{E}(t)\}$. The swarm configuration $x_s \in X_s$ of a swarm S is defined as $x_s = col^{N}_{i=1}(x_{s_i})$, where $x_{s_i} = [\bar{x}_i \quad \bar{y}_i]^{\mathrm{T}}$, $i = 1, \cdots, N$, and the operator $col(\cdot)$ stacks the argument vectors. The swarm structure χ is then defined as the tuple $\chi = (\mathrm{V}(t), \mathrm{E}(t), x_s)$. Now, our objective is to design a swarm formation control law u that steers a swarm of robots S given by

$$\dot{x} = G(x)u, \tag{74}$$

where $x = col^{N}_{i=1}(x_i)$, $G(x) = \mathrm{diag}(g_1(x_1), \cdots, g_N(x_N))$, $u = col^{N}_{i=1}(u_i)$, along the desired path. Given a large number of robots evolving in the swarm configuration space X_s, solving the aforementioned control problem is non-trivial as the dimension of the swarm system in (74) depends on the number

of robots in the swarm. To remedy this, we capture the motion of the swarm in terms of the motion of an abstract shape. An abstract shape is a convex closed curve S circumscribing the convex hull of configuration x_s of the swarm structure χ. In the Cartesian coordinate $(\bar{x}, \bar{y})$, the abstract shape S is given by

$$\left|\frac{\bar{x} - \mu_x}{s_w}\right|^{m_a} + \left|\frac{\bar{y} - \mu_y}{s_l}\right|^{n_a} = 1, n_a \geq 2, m_a \geq 2, \tag{75}$$

where (μ_x, μ_y) is the coordinate of center of the abstract shape, s_l and s_w are the length of the semi-major and semi-minor axes of the abstract shape, respectively. Let $a \in \mathrm{R}^5$ be the vector of parameters of abstract shape (see Section 2.3.1.5.2). Also, let a surjective submersion

$$\Phi : \mathrm{R}^{2N} \to \mathrm{R}^5, \quad \Phi(x_s) = a. \tag{76}$$

relate the abstract shape parameters a and the swarm configuration x_s. Then, we can specify the desired path for the swarm as the trajectory of the abstract shape parameters a. Specifying the swarm's desired trajectory in terms of the trajectory of the abstract shape parameters is more practical than providing the desired trajectory for every member robot in the swarm. Also, in this approach, the swarm's trajectory is independent of the number and permutation of robots in the swarm.

Now, we aim to design a distributed control law u_i for each Robot i in the swarm so that the abstract shape parameters a track the desired trajectory ζ specified by the user (motion planner or human). However, the synthesis of control law u_i requires each Robot i in the swarm to know the position of all robots in the swarm to determine the abstract shape parameters a. In [90], this requirement is handled by introducing a central observer that moves with the swarm. The observer collects the position information of all robots in the swarm, computes the abstract shape parameters, and broadcasts the computed value of the abstract shape parameters to all robots in the swarm. However, this approach requires all the robots to be in the communication range of each other or the observer, thus, prone to high bandwidth requirements or a single point of failure at the observer. To circumvent this problem and realize a distributed control architecture, we design a dynamic average consensus estimator for each Robot i to estimate the abstract shape parameters a only based on the information collected from Robot i's neighboring robots. More precisely, we state the formation and trajectory tracking control problem as follows:

Problem 2. *Under Assumption 1, given a time varying desired trajectory* $\zeta \in \mathrm{R}^5$ *of an abstract shape,*

a) *Construct the abstract shape parameters a.*

b) *For the abstract shape parameters a, design a distributed neighbor-based estimator so that each robot's estimation of abstract shape parameters* $\bar{a}_i$ *converges to a in a finite time, for all* $i \in \{1, \cdots, N\}$.

c) *Generate the desired trajectory for Robot i,* $i \in \{1, \cdots, N\}$, *such that the abstract shape parameters a tracks* ζ.

d) *Design a feedback control law* u_i *for Robot* i, $i \in \{1, \cdots, N\}$, *so that each robot tracks its desired trajectory, and the swarm system (74) tracks the given trajectory* $\zeta \in \mathrm{R}^5$.

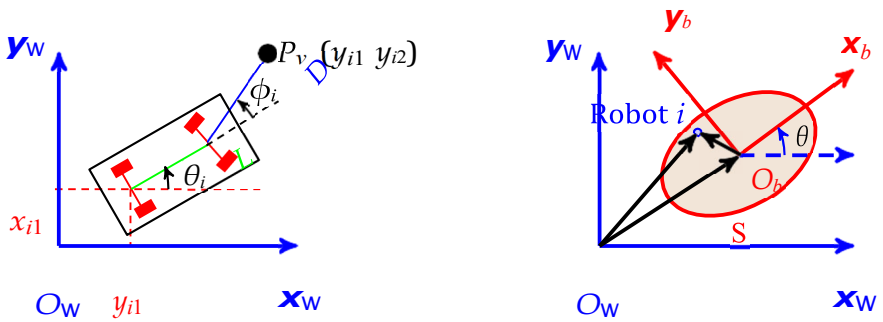
3.3.1.5.2 The leaderless swarm formation control framework A swarm formation control is a challenging problem. Complexities of the robot kinematics and the swarm dynamics often lead to intractable control problems. This section employs tools from differential geometry, consensus, and control theory to systematically design local control laws for Robot i in the swarm to realize a specified swarm formation ζ. The desired time-varying swarm formation is given in terms of the abstract shape parameters a. For Robot i, we propose a control law that drives the pose and shape of the swarm (the motion of the abstract shape) to track the desired trajectory ζ. For this purpose, Robot i estimates the value of the abstract shape parameters a via a dynamic consensus estimator from the information available at neighboring robots. The detailed design of the proposed control framework is presented in the following sections.

3.3.1.5.3 Trajectory-tracking control law This Section solves a trajectory tracking con- trol problem (Problem 2.d) using input-output linearization. For this purpose, we define the Robot i's output $y_i = h_i(x_i)$, $h_i(x_i) : \mathrm{R}^4 \to \mathrm{R}^2$ in such a way that the decoupling between Robot i's linear input-output dynamics and internal dynamics is achieved:

$$\begin{aligned} h_{i_1} &= x_{i_1} + L\cos x_{i_3} + D\cos(x_{i_3} + x_{i_4}) \\ h_{i_2} &= x_{i_2} + L\sin x_{i_3} + D\sin(x_{i_3} + x_{i_4}), \end{aligned} \tag{77}$$

where r is the total relative degree of Robot i and $D \neq 0$ is the "look-ahead" distance. The output function h_i, defines the position of a virtual point P_v (see Figure 114a) in front or behind of Robot i based on the sign of D to simplify the control design by decoupling input-output dynamics and internal dynamics. Let the augmented function $\bar{h}_i(x_i) : \mathrm{R}^4 \to \mathrm{R}^{4-r}$ be chosen as $\bar{h}_i(x_i) = [x_{i_3} \quad x_{i_4}]^{\mathrm{T}}$ such that the state transformation $T_i(x_i) = [q_i^{\mathrm{T}} \quad \dot{q}_i^{\mathrm{T}}]^{\mathrm{T}} = [h_i^{\mathrm{T}}(x_i) \quad \bar{h}_i^{\mathrm{T}}(x_i)]^{\mathrm{T}}$ is a diffeomorphism. Also, let the control input u_i to Robot i be given as $u_i = \bar{\alpha}_i(x_i) + \bar{\beta}_i(x_i)\bar{v}_i$, where $\bar{\alpha}_i(x_i) = 0$ due to the kinematic model being drift free, and $\bar{\beta}_i(x_i) = \Delta_i^{-1}(x_i)$ where $\Delta_i(x) = \begin{bmatrix} L_{g_{i_1}} h_{i_1}(x_i) & L_{g_{i_2}} h_{i_1}(x_i) \\ L_g h_i(x_i) & L_g h_i(x_i) \end{bmatrix}$ is the decoupling matrix, $L_{g_{i_j}} h_{i_k}(x_i)$ is the Lie derivative of function $h_{i_k}(x_i)$ along a vector field $g_{i_j}(x_i)$, for $j = 1, 2$ and $k = 1, 2$. Using the state transformation $T_i(x_i)$ and state feedback control law u_i, we transform (73) into

$$\dot{q}_i = \tilde{f}_i(q_i, \dot{q}_i), \dot{q}_i = A_i q_i + B_i \bar{v}_i, y_i = C_i q_i, \tag{78}$$



(a) (b)

Figure 136: (a) Robot reference frames and definition of virtual point Pv, (b) Swarm reference frames and the abstract shape that circumscribes the region occupied by swarm of robots.

where $q_i = [h \quad h]^T$, $\dot{q} = f(q, q) = \frac{\partial h_i(x_i)}{\partial x_i} \dot{x}$ is the internal dynamics, and $A = [0\ 0]$, $B = I$, $C_i = I_2$, $v_i = [v_{i1} \quad v_{i2}]^T$. Based on this, we design a linear control law v_i so that q_i can track desired trajectory q_{id}. Assuming that all states of the control affine system in (73) are measurable, for stable zero dynamics [107], we employ the control law

$$v_i = \dot{q}_{id} + K_i(q_{id} - q_i), K_i > 0, \tag{79}$$

where K_i is the control gain, for Robot i to exponentially tracks the desired trajectory q_{id}. In many application area of swarm of robots, specifying reference trajectory q_{id} for each individual robot is not practical. Therefore, we design q_{id} by solving the multi-input multi-output (MIMO) input-output linearization problem in the next section.

3.3.1.5.4 Trajectory generation The collective behavior of robots in the swarm is cap- tured by the motion of the abstract shape, which is represented by the trajectory of the abstract shape parameters a. For each robot in the swarm, we design a distributed swarm controller so that the abstract shape parameters a tracks the desired trajectory ζ. In this setting, we design a dynamic average consensus estimator to estimate the abstract shape parameters a. Then, for each Robot i, the swarm controller output will be converted to the desired trajectory q_{id} to be tracked by the trajectory tracking control law (79). For Robot i, the approaches to the desired trajectory generation are discussed next.

We start by putting together the linearized input-output dynamics given in (78) to form a new swarm system as

$$\dot{q} = Aq + B\bar{v}, \quad y = Cq, \tag{80}$$

where $A = \text{diag}(A_1, \cdots, A_N)$, $B = \text{diag}(B_1, \cdots, B_N)$, $C = \text{diag}(C_1, \cdots, C_N)$, $q = col_{i=1}^{N}(q_i)$, $v = col_{i=1}^{N}(v_i)$. The output y of the swarm system in (80) is the collection of the output of individual robots (local behaviors). However, the control specifications for the swarm is given in terms of collective (global) behaviors of the swarm. To address this issue, we transform (80) from the robot configuration space to the abstract shape space using the input-output linearization technique. To input-output linearize the swarm system in (80), we construct a new output function y to be the abstract shape parameters a of abstract shape. Let the abstract shape be described in the world coordinate by Frame {W} (shown in Figure 114b with center O_W and basis vectors $\{\boldsymbol{x}_W, \boldsymbol{y}_W\}$) and in the body coordinate by Frame $\{b\}$ (shown in Figure 114b with center O_b and basis vectors $\{\boldsymbol{x}_b, \boldsymbol{y}_b\}$). The position vector of virtual point of Robot i with respect to Frame {W} is represented by q_i and the position vector describing the origin of Frame $\{b\}$ with respect to Frame {W} is denoted by O_b^W. Let $R_b^W \in SO(2)$ be the rotation matrix of Frame $\{b\}$ with respect to Frame {W} and p_i be the position vector of virtual point of Robot i with respect to Frame $\{b\}$. Using geometry, p_i is given by

$$p_i = [p_{ix} \quad p_{iy}]^T = -R_b^{W\,T} O_b^W + R_b^{W\,T} q_i, \tag{81}$$

where p_{ix} and p_{iy} are the components of vector p_i in Frame $\{b\}$. The origin O_b^W of Frame $\{b\}$, μ, is the mean of position vectors of virtual point of each Robot i in Frame {W}. It represents the center of the abstract shape : $O_b^W = \mu = \frac{1}{N}\sum_{i=1}^{N} q_i$. We utilize the co-variance matrix of robot distribution in the region circumscribed by the abstract shape to define the shape and orientation of the swarm. The co-variance matrix of the ensemble of the robots in Frame $\{b\}$ is given by

$$\Sigma_1 = \begin{bmatrix} \frac{1}{N-1}\sum_{i=1}^{N}(p_{ix}-0)^2 & 0 \\ 0 & \frac{1}{N-1}\sum_{i=1}^{N}(p_{iy}-0)^2 \end{bmatrix}. \tag{82}$$

Similarly, the co-variance matrix of the ensemble of the robots in Frame {W} is given by

$$\Sigma_0 = \begin{bmatrix} \sigma_{xx} & \sigma_{xy} \\ \sigma_{xy} & \sigma_{yy} \end{bmatrix}. \tag{83}$$

The covariance matrix Σ_0 is related to the co-variance matrix Σ_1 in Frame $\{b\}$ as

$$\Sigma_1 = R_b^{W\mathrm{T}} \Sigma_0 R_b^W = \begin{bmatrix} s_1 & s_{12} \\ s_{12} & s_2 \end{bmatrix}. \tag{84}$$

Solving (84), the orientation θ of the abstract shape will be

$$\theta = \frac{1}{2} tan^{-1}\left(\frac{2\sigma_{xy}}{\sigma_{yy}-\sigma_{xx}}\right). \tag{85}$$

Now, consider the convex hull that captures p percentage of robots in the swarm. Then, the width and length of the abstract shape in Frame $\{b\}$ can be captured by $s_w = \sqrt{c_p s_2}$ and $s_l = \sqrt{c_p s_1}$, respectively, where $c_p = -2\ln(1-p)$. The abstract shape parameters is given by a 5−dimensional vector $a = [\mu = [\mu_x, \mu_y], \theta, s_1, s_2]^{\mathrm{T}}$. Besides, assuming that $q \approx x$, from Definition 76 we have $\Phi(x_s) \approx \Phi(q) = a$. Now, to address Problem 2.a, we define the mapping Φ as $\Phi(q) = a = [\Phi_1(q) \cdots \Phi_5(q)]^{\mathrm{T}}$, where $\Phi_1(q) = \mu_x$, $\Phi_2(q) = \mu_y$, $\Phi_3(q) = \theta$, $\Phi_4(q) = s_2$, and $\Phi_5(q) = s_1$. Then, the state feedback control law will be

$$v^- = \alpha(q) + \beta(q)w, \tag{86}$$

where

$$\alpha(q) = -\Delta_s^{-1}(q).\left[L_f^{r_{i_1}}\Phi_1(q) \quad \cdots \quad L_f^{r_{i_5}}\Phi_5(q)\right]^{\mathrm{T}} = 0,$$

$$\Delta_s(q) = \begin{bmatrix} L_{g_1}L_f^{r_{i_1}-1}\Phi_1(q) & \cdots & L_{g_N}L_f^{r_{i_1}-1}\Phi_1(q) \\ \vdots & \ddots & \vdots \\ L_{g_1}L_f^{r_{i_5}-1}\Phi_5(q) & \cdots & L_{g_N}L_f^{r_{i_5}-1}\Phi_5(q) \end{bmatrix},$$

$$= \begin{bmatrix} \frac{1}{N} & \cdots & \frac{1}{N} \\ \vdots & \ddots & \vdots \\ (q_1-\mu)^{\mathrm{T}}R_{s_1} & \cdots & (q_N-\mu)^{\mathrm{T}}R_{s_1} \end{bmatrix}, \quad \beta(q) = \Delta_s^{-1}$$

$$\Delta_s(q) = \begin{bmatrix} \frac{1}{N-1} & \cdots & \frac{1}{N-1} \\ \vdots & \ddots & \vdots \\ (q_1-\mu)^{\mathrm{T}}R_{s_1} & \cdots & (q_N-\mu)^{\mathrm{T}}R_{s_1} \end{bmatrix}, \quad f = Aq = 0$$

$$R_{s_1} = \begin{bmatrix} 2\cos^2\theta & \sin 2\theta \\ \sin 2\theta & 2\sin^2\theta \end{bmatrix}, \quad [g_1 \quad \cdots \quad g_N] = B,$$

where $r_{i_j} \in \mathrm{N}, i = 1, \cdots, N$ $j = 1, \cdots, 5$, is the vector relative degree of (80), $L_f\Phi_j(q)$ is the Lie derivative of function $\Phi_j(q)$ along a vector field f, and $L_{g_k}L_{f_i}\Phi_j(q)$ is the Lie derivative of function $\Phi_j(q)$ along a vector field f_i and along another vector field g_k, where $k = 1, \cdots, N$. The surjective submersion $\Phi(q)$ and the state feedback control law v^- in (86) transforms (80) into

$$\Xi_2 = \{\dot{a}(t) = \bar{A}\,a + \bar{B}\,w, \qquad y^- = a, \tag{87}$$

where $\bar{A} = 0_{5\times5}$, $\bar{B} = I_{5\times5}$, and w is the control law that steers abstract shape parameter a to track the desired trajectory ζ. For the sake of reducing the computation cost, we simply design w as a linear control law, given as

$$w = \bar{K}(\zeta - \bar{a}) + \dot{\zeta}, \tag{88}$$

where $\bar{K}$ is the control gain and $\bar{a}$ is the estimate of a.

Now, we can calculate v_i from w as $v_i = \alpha_i(q_i) + \beta_i(q_i)w_i$, where $\alpha_i(q_i)$ and $\beta_i(q_i)$ are i^{th} row of $\alpha(q)$ and $\beta(q)$, respectively. We then compute the reference trajectory q_{id} by solving

$$\dot{q}_{id} = \alpha_i(q_i) + \beta_i(q_i)(\bar{K}(\zeta - \bar{a}) + \dot{\zeta}), \tag{89}$$

with $q_i(0) = x_{s_i}(0)$ being known.

3.3.1.5.5 The dynamic consensus estimator Determining the abstract shape parameters vector $a \in R^5$ requires a centralized communication architecture or all-to-all communication among the robots in the swarm. This process is prone to failures associated with the centralized observer and with communication links between the observer and individual robots in the swarm. Therefore, rather than relying on a central observer to compute the abstract shape state vector a, we estimate the abstract shape state vector a (Problem 2.b) by exploiting the underlying graph structure of the network of robots in a distributed way using dynamic average consensus. To leverage the technique of average consensus, we represent all the components of the abstract shape parameters a in terms of the average of suitable expressions. First, we re-write (85) as $\theta = \frac{1}{2}\tan^{-1}\left(\frac{\sigma_1^*}{\sigma_2^*}\right)$, where $\sigma_1^* = \frac{2\sigma_{xy}}{N}$ and $\sigma^* = \frac{\sigma_{yy}-\sigma_{xx}}{N}$. Then, we introduce as:

$$z_i = \begin{bmatrix} z_{i_1} \\ z_{i_2} \\ z_{i_3} \\ z_{i_4} \\ z_{i_5} \\ z_{i_6} \end{bmatrix} = \begin{bmatrix} q_{ix} \\ 2(q_{iy} - \mu_y)(q_{ix} - \mu_x) \\ q_{iy} - \mu_y \\ q_{iy} - \mu_y{}^2 - q_{ix} + \mu_x{}^2 \\ p_{ix}^2 \\ p_{iy}^2 \end{bmatrix}. \tag{90}$$

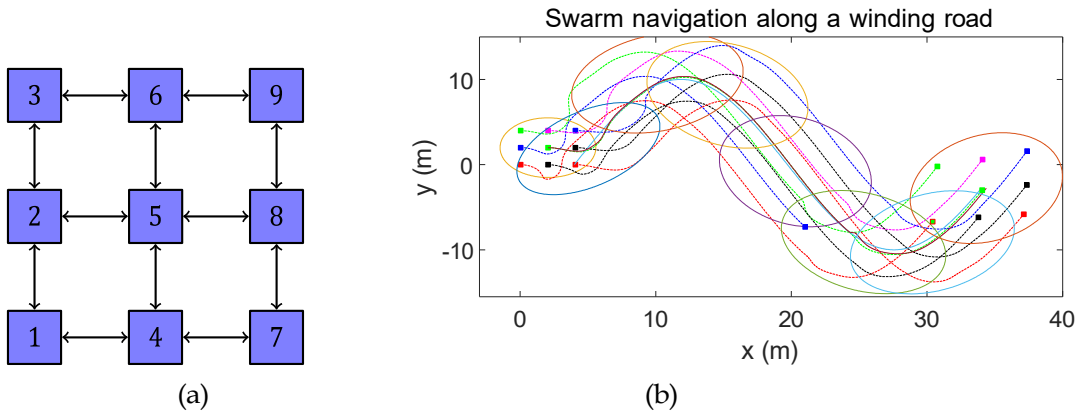
Accordingly, for swarm of N robots, parameters $[\mu_x\ \mu_y\ \sigma^*\ \sigma_2^*\ s_1\ s_2]^{\mathrm{T}}$ are expressed as the average of z_i, that is, $\frac{1}{N}\sum_{i=1}^{N} z_{i_k}$. To estimate $\frac{1}{N}\sum_{i=1}^{N} z_{i_k}$, where $k = 1, \cdots, 6$, we implement an edge-based dynamic consensus estimator of the form

$$\begin{aligned} \dot{\eta}_{i}^{+}{}_{j_k} &= -\rho \tanh\{c(\gamma_{i_k} - \gamma_{j_k})\} \\ \dot{\eta}_{i}^{-}{}_{j_k} &= -\rho \tanh\{c(\gamma_{j_k} - \gamma_{i_k})\},\ c \geq 1,\ j \in \mathrm{N}_i \\ \gamma_{i_k} &= \sum_{j\in\mathbf{N}_i} \eta_i^{+}{}_{j_k} - \sum_{j\in\mathbf{N}_i} \eta_i^{-}{}_{j_k} + z_{i_k}, \quad k = 1, \cdots, 6, \end{aligned} \tag{91}$$

where $\eta_i = [\eta_i^{+}{}_j \quad \eta_i^{-}{}_j]^{\mathrm{T}} \in \mathrm{R}^{2\mathrm{N}i}$ is the internal state of the estimator on Robot i, $\rho \in \mathrm{R}$ and $c \in \mathrm{R}$

are global estimator parameters, and $\gamma_i \in \mathrm{R}$ is the estimate of $\frac{1}{N}\sum_{i=1}^{N} z_i$ where $k = 1, \cdots, 6$. From (91), it is clear that the edge dynamics captures the state of the disagreement between Robot i and Robot j. Further, the use of $tanh(\cdot)$ in (91), makes the proposed estimator smooth, avoiding the chattering phenomena [108]. This approach makes the protocol robust to agents joining or leaving the network, and to communication link failures among the agents. The proposed estimator has three stages due to the fact that the estimation of the average of some of the components of z_i requires the knowledge of the average of other components of z_i. In the first stage, we estimate the average of z_{i_1} and z_{i_2} by the mean estimator. Using the information from the mean estimator stage, the average of z_{i_3} and z_{i_4} is then estimated in the second stage by the orientation estimator. Similarly, in the third stage, using the information from the orientation estimator, we estimate the average of z_{i_5} and z_{i_6} by the width and length estimator. Then, the estimate of the components of abstract shape parameters a at each Robot i is given by $\bar{\mu} = [\gamma_{i_1} \quad \gamma_{i_2}]$, $\bar{\sigma}_1^* = \gamma_{i_3}$, $\bar{\sigma}_2^* = \gamma_{i_4}$, $\bar{s}_1 = \gamma_{i_5}$, and $\bar{s}_2 = \gamma_{i_6}$. Based on this, the estimation of the abstract shape orientation is given as $\bar{\theta} = \frac{1}{2} tan^{-1}(\frac{\bar{\sigma}_1^*}{\bar{\sigma}_2^*})$. Further, the estimate of the length of semi-minor axis $\bar{s}_w$ and the semi-major axis $\bar{s}_l$ of the abstract shape are given as $\bar{s}_w = \sqrt{\bar{c}_p \bar{s}_2}$ and $\bar{s}_l = \sqrt{\bar{c}_p \bar{s}_1}$, respectively. Accordingly, the estimate of the abstract shape parameters $\bar{a}$ is given by

$$\bar{a}(t) = [\bar{\mu} \quad \bar{\theta} \quad \bar{s}_2 \quad \bar{s}_1]^{\Gamma}. \tag{92}$$



(a) (b)

Figure 137: (a) The communication graph of the swarm of robots in the conducted simulation. (b) Navigation of swarm of robots along a desired trajectory ζ. The robots are initially in a square formation. Their formation evolves to rectangular and parallelogram shapes along the road while tracking ζ.

3.3.1.5.6 Simulation results In this Section, we present numerical simulation results to illustrate the performance of our leaderless swarm formation control system. We consider a group of 9 identical rear wheel driving car-like robots with the virtual reference point of each robot located at $D = 0.05m$ away from its center. The robots' initial locations, heading angles, and steering angles are given as (0,0,0,0), (0,2,0,0), (0,4,0,0), (2,0,0,0), (2,2,0,0), (2,4,0,0), (4,0,0,0), (4,2,0,0), and (4,4,0,0) for Robots 1-9, respectively. The underlying communication graph is given in Figure 115a. The initial formation is a $4m \times 4m$ square grid, circumscribed by a circle with radius of $3.6091m$.

Consider that the swarm of robots is tasked to navigate along a winding road, given by the trajectory $\zeta(t) = [[\mu_{x_d} \quad \mu_{y_d}] \quad \theta_d \quad s_{2_d} \quad s_{1_d}] = [t + 4 \quad 10\sin(0.2t) \quad \tan^{-1}(2\cos(0.2t)) \quad 10.513 \quad 13.57]_T$ and $\dot{\zeta}(t) = [1 \quad 2\cos(0.2t) \quad -0.4\frac{\sin(0.2*t)}{(4*\cos^2(0.2*t)+1)} \quad 0 \quad 0]^T$. The width of the road is $10.513m$ and we want the length of swarm to be $13.57m$. All the robots have the knowledge of ζ and $\dot{\zeta}(t)$. After

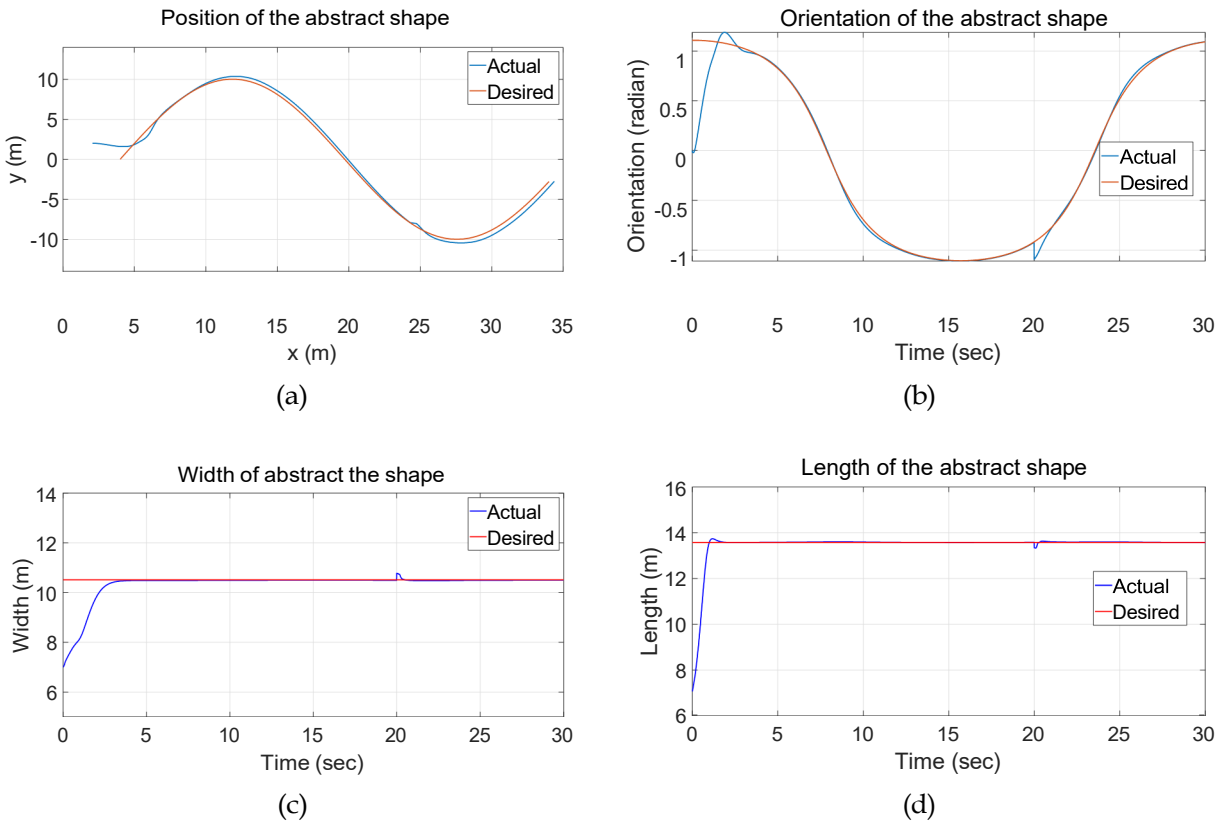


Figure 138: The abstract shape parameters trajectory: (a) Swarm position tracking, (b) Swarm orientation tracking, (c) swarm width tracking, (d) swarm length tracking.

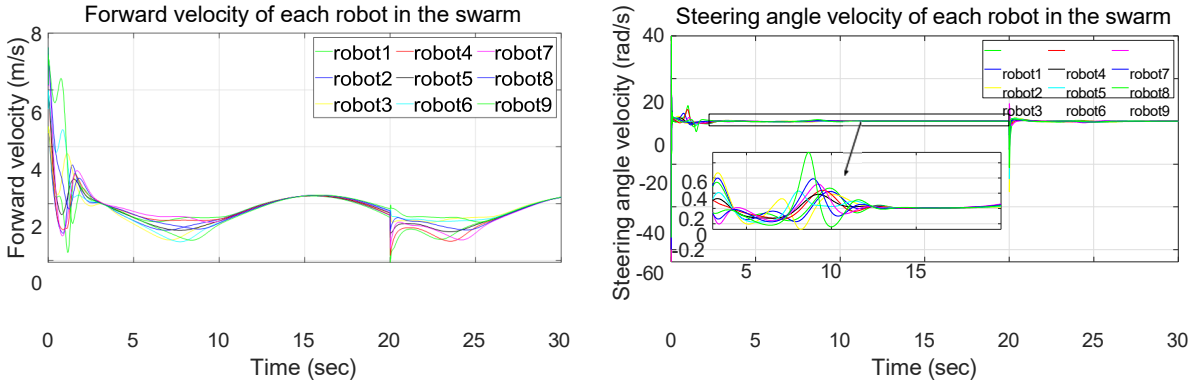
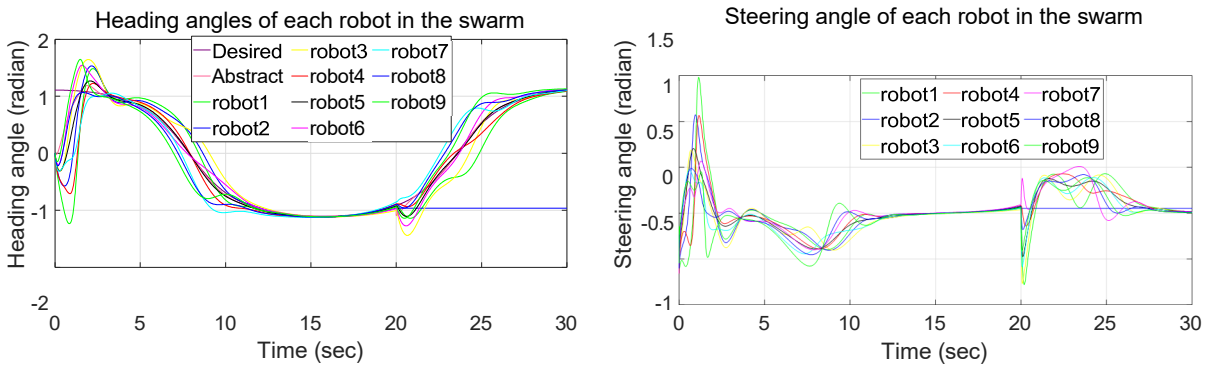


Figure 139: Inputs to the robots: (a) the driving velocity of each robot, (b) the steering angle velocity of each robot.

20s, we disable Robot 2 to test the robustness of the proposed control algorithm. To track the given path, we use control gains $\bar{K}_i = [2.5I_{2\times 2} \quad 3 \quad 0.06 \quad 0.08]^T$ and $\check{K}_i = [0.0008 \quad 0.0008]^T$ for the swarm controller and the robot position controller on each Robot i, respectively. The dynamic consensus estimator is used to estimate the values of the abstract shape parameters a to be used in the desired trajectory generation. The consensus parameters are selected to be β 79 and ϵ 2. Based on this, the simulation results of leaderless swarm formation control are presented in Figure



(a) (b)

Figure 140:(a) The heading angle of each robot, (b) the steering angle of each robot in the swarm.

116.

Looking at individual robots in Figure 115b, the swarm of robots tracks affine transformations of the initial square grid formation, where the swarm is expanded and elongated by changing its formation between a rectangular and parallelogram shape. Figure 115b also shows the motion of the abstract shape of the swarm captured by different snapshots of ellipses (m_a 2, n_a 2). To further investigate the history of the swarm's configuration, the abstract shape parameters trajectory including μ, θ, s_w and s_l, are shown in Figure 116. The simulation results in Figure 117 show the steering and forward velocity as inputs to individual robots. Further, Figure 118 shows the heading angle and steering angle of individual robots in the swarm. From these simulation results, it can be observed that the robots in the swarm have almost similar velocity and heading angle while navigating the road. Also, the proposed algorithm performs well against robot failures and communication link failures as long as the communication graph $\mathcal{G}(t)$ remains connected aftermath of the failures. To demonstrate this, we made *Robot* 2 to stop moving at $t = 20s$ and disabled its communication links with its neighbours. Accordingly, the simulation results show that after failure of Robot 2, the swarm again converges to the desired shape and continues tracking the desired trajectory.

3.3.1.5.7 Conclusion In this work, we introduced a distributed swarm formation control framework for transferring a swarm of robots from a current location to the desired location while allowing the shrinkage, expansion, elongation, and compression of the swarm along a reference time-varying path. For this purpose, we represented the swarm by an abstract shape that circumscribes the convex hull of robots' positions. Then, for each robot in the swarm, we designed a distributed control law to track a suitable trajectory that allows the swarm to follow a desired time-varying swarm formation without relying on any leader. We also developed a dynamic average consensus estimator algorithm to estimate the abstract shape states in a distributed manner for use in a trajectory generation. We demonstrated the effectiveness and robustness of the designed control system through simulations by introducing failures to individual robots and their communication links.

3.3.2 Project Progress for Task T3-2 (Developing a compositional verification approach for LSASV)

3.3.2.1 Period of Performance under

Task T3-2 Start Date: August 2016

Conclusion Date: August 2021

Faculty lead: Dr. Ali Karimoddini

3.3.2.2 General Description of Task T3-2

In robotics applications, it is typical to construct a pre-planned series of actions to achieve a goal, particularly for industrial robots used in factories and assembly lines. These robots do a task in a cyclic fashion with great precision in a predefined environment. During the operation of the robots if a need arise to conduct a new task, an operator has to reconfigure the robots or update the software manually. Even if such kind of behavior or mode of operation is appropriate in a pre-defined static environment, it is not desirable for robots operating in a highly dynamic environment. For example a battle robot has to adapt to the enemy movement or strategy in real-time without out waiting for instruction or a helper robot should be able to navigate avoiding moving obstacles. These kind of tasks requires the robot to have an intelligence for replanning the immediate next action based on the observable events. We propose to capture the intelligence using behavior trees (BTs) which are modular, hierarchical and more readable control architecture in comparison to mostly used finite-state-automata (FSA).

Further, with advances in communication, computation, and control technologies, it is now becoming possible to deploy Intelligent in the form of a heterogeneous team of autonomous vehicles with different capabilities (sensors and actuators) to collectively accomplish complex missions and tasks, which are distributed in time and space and may not be possible to be achieved individu- ally. A cooperative control strategy not only can handle such complex scenarios, but also could significantly reduce the cost, enhance the resilience of the overall system, and improve the team functionality through sharing resources and distributing tasks and loads. Nonetheless, multi-agents cooperation introduce challenges and complexities including but not limited to task decomposition, task assignment, communication, task execution, and task monitoring. A common method for tasking multi-agent systems is to employ scheduling mechanisms. For example, [109] provides a reliable scheduling algorithm for a team of agents with the capability to conduct dynamic rescheduling. Alternatively, one can use consensus-based algorithms to minimize the cost or maximize the number of tasks assigned to an agent. The challenge is that generally tasks are sequences of actions that have to be completed is a particular sequence and often in collaboration with other agents, which is beyond the scheduling problem as it requires a coordinator(s) to synthesize and execute a scheduling and sequencing plan in a collaborative setting.

To address the challenge of coordinating multiple intelligent agents, in this work, we distributively and reactively synthesize the local BTs on-the-fly for a set of streamed tasks, so that each agent is responsible for synthesizing its own BT. Moreover, our proposed technique incorporates a tasking mechanism by assigning the tasks via a market-based auctioning algorithm to minimize the cost. In the proposed framework, collaboration among agents is needed if and only if a single robot cannot do the task alone, thus resource utilization is improved leaving other robots for new tasks.

The developed method is illustrated via several examples, and the effectiveness of the proposed approach is verified.

3.3.2.3 Objectives for Task T3-2 over the Course of Project

- **Objective Name:** Automatic behaviour tree synthesis for the coordination of multi-agents

  Objective Type: Research

  Objective Description: Develop behavior tree (BT)-based automatic tasking, synthesis, and execution framework for the coordination of heterogeneous agents with different capabilities to meet the goal of a series of tasks. In the proposed framework there are two-levels of auctioning where agents compete to win either a task (has to be expanded into a sequence of actions) or an action. Further, collaboration among agents is on a need basis, i.e., if an agent lacks the capability to perform an action, that action could be completed by delegation.

  Impact: The developed behavior tree based multi-agent coordination platform allows the deployment of a team of sophisticated autonomous vehicles with different capabilities to collectively accomplish complex missions and tasks that are distributed in time and space.

3.3.2.4 Progress Against Planned Objectives in Task T3-2

1. **Objective 1** Developed BT based framework for the collaboration of heterogeneous agents to accomplish a task either individually or in collaboration with other agents using market based auctioning.
2. **Objective 2** Provided proof of correctness and complexity of the proposed framework
3. **Objective 3** Implemented the BT based framework for a single and multiple-agents in robotics operating system (ROS) environment.

3.3.2.5 Technical Accomplishments in Task T3-2

In the literature, different techniques are developed to address these issues. In [110], an optimal multi-robot tasking framework is introduced by modeling each robot as a weighted transition system and composing the model of the robots with the mission requirement expressed as a linear temporal logic (LTL). However, this framework is centralized and as the number of participating robots increases, scalability becomes a bottleneck. In [111], a bottom-up decentralized multi-agent control strategy is presented, where each agent synthesizes a local controller to ultimately meet a global mission. However, since the agents have no prior knowledge about each local task, conflicts among agents could arise which has to be resolved by a centralized mission controller. In [112], an automated supervisory control framework models the system and its corresponding specifications as a discrete event system (DES) where the supervisor centrally coordinates the robots by enabling or disabling controllable events. In turn, each robot chooses to execute or not to execute the enabled events that are communicated back to the supervisor to enforce team behaviour.In [113] mixed integer linear programming (MILP) formulation is used for multi-UAV task allocation, multi-robot target tracking, and optimal area coverage problems in a centralized way. In [114], decentralized optimal task

allocation algorithms are introduced. These (Centralized or decentralized) optimal task allocation methods rely on offline computations, requiring the information about the environment and all tasks in advance. Therefore, if new tasks are introduced or if the environment changes, these methods require iterative and expensive repetition of the entire optimization process.

In this work, we propose to employ Behaviour Trees (BTs) [115] and integrate them with auctioning mechanisms to simultaneously allocate and execute the tasks. BTs are graphical mathematical models for execution of tasks with inherent hierarchical, modular, and reactive properties. With BTs it is fairly more convenient to manage, modify, and add tasks or subtasks due to the modular and scalable structure of BTs. In [116], given a global BT controller and assuming that the tasks are decomposable, a heuristic approach is employed to create local BTs for each agent. However, the question of how to determine the global BT is left unanswered. In [117], a procedure is pro- vided to obtain a BT for a single robot that meets a mission specification. To extend this method to multi-agent systems, one way is to use the method in [117] to develop a global BT and then decompose the global BT to local BTs using the method in [116]. However, this approach is not computationally efficient and may end up with the state explosion problem for larger systems.

To address these challenges, we synthesize the local BTs for individual agents in a decentralized manner to accomplish a given set of tasks. The proposed technique integrates the developed BT synthesis mechanism with an auctioning algorithm to minimize the overall cost. In the proposed framework, collaboration among agents is needed if and only if a single robot cannot do the task alone, thus resource utilization is maximized leaving other robots for new tasks. In the proposed framework, the task allocation and task execution are handled simultaneously by synthesizing local BTs which generate an appropriate sequence of actions to meet the goals of the tasks. This method provides the agents with autonomy capabilities to handle newly introduced tasks on the fly, while reacting to changes in the environment. The developed method is illustrated via several examples and the effectiveness of the proposed approach is verified.

3.3.2.5.1 Problem formulation In this section, we use BTs to formulate the coordination and tasking for multi-agent systems over the following components:

1. The set R which includes a team of robots $R = \{R_1, \cdots, R_M\}$ where $M \in \mathbb{N}$ is the number of agents. Here, the terms agents, robots, and vehicles are used interchangeably.

2. The set A is the global action bank and contains a set of actions A_k, $k = 1, \cdots, L$, where $L \in \mathbb{N}$ is the total number of actions. We define a set of action capability indicators $\hat{a}_{ik}$, $i = 1, \cdots, M$, $k = 1, \cdots, L$, for which $\hat{a}_{ik} = 1$ if the robot R_i can accomplish Action A_k, otherwise $\hat{a}_{ik} = 0$. Here, the robots are assumed to perform single action at a time.

3. The set T which includes a set of complex Tasks (a task can be decomposed into multiple set of actions that could satisfy the same task goal in different ways [118]) T_j, $j = 1, \cdots, N$, where $N \in \mathbb{N}$ is the number of tasks. The accomplishment of each task, T_j, can be captured by meeting a condition C_j. For example, if the task T_1 is to "reach a goal region", then C_1 is "being at the goal region." We also define a set of task indicators x_{ij}, $i = 1, \cdots, M$, $j = 1, \cdots, N$, for which $x_{ij} = 1$ if the task T_j is assigned to R_i to handle it individually or in collaboration with other robots, otherwise $x_{ij} = 0$. Similarly, we define a set of action assignment indicators $\hat{x}_{ijk}$, $i = 1, \cdots, M$, $j = 1, \cdots, N$, $k = 1, \cdots, L$, for which $x_{ijk} = 1$ if action A_k of R_i is assigned for task completion of T_j. To reach the "goal" of a task T_j, depending on the agent that is responsible

to handle the task, a series of actions from the action bank A should be completed, where the last action should meet C_j. In our proposed framework, only a robot that can accomplish an action which meets C_j, can be a candidate for being selected to handle T_j. Such a robot can complete an action to meet C_j, and may delegate the prerequisite actions to other agents if necessary. Further, we define the indicators $\hat{a}_{ijk}$ $i = 1, \cdots, M, j = 1, \cdots, N, k = 1, \cdots, L$, for which $\hat{a}_{ijk} = 1$ if action A_k from robot R_i is needed to complete the task T_j, otherwise $\hat{a}_{ijk} = 0$.

4. The set F includes a set of values $f_{ij} : R \times T \rightarrow \mathrm{R}^+$ to describe the cost of handling the task T_j by R_i based on performance, energy, and proximity. Robot R_i can accomplish the actions in T_j individually or delegate the actions to other robots if necessary. We define a cost function $\hat{f}_{ik} \in \mathrm{N}$, which indicates the cost of accomplishing an individual action A_k by the agent R_i.

5. We define the set $\hat{C}$ which includes a set of preconditions $\hat{c}_{ikp}$ $i = 1, \cdots, M, k = 1, \cdots, L$, and $p = 1, \cdots, P_k$, where P_k is the number of preconditions for action A_k, and $\hat{c}_{ikp}$ specifies pth preconditions for completing action A_k by robot R_i. We also define action status indicator $\hat{c}_{ik}$ where $\hat{c}_{ik} = 1$ if action A_k is executed and completed by R_i, otherwise $\hat{c}_{ik} = 0$.

6. Consider a discrete clock clk with a granularity of 1sec, i.e., $clk = clk + 1$ (this can be of smaller step sizes if needed). The clock clk represents the elapsed time starting from the first task announcement. Then, we define Δt_{ik}, $i = 1, \cdots, M, k = 1, \cdots, L$, which represents the duration the agent R_i needs to complete the action A_k. We also define an action timeline indicator t_{io}, $i = 1, \cdots, M$, $o = 1, \cdots O$, where $O \in N$ is the last sample time, and $t_{io} = 1$ during the time that R_i is assigned to perform one of the actions A_*, which takes R_i for Δt_{i*} time units.

7. We define an operation $R_i \models con$ which checks if the agent R_i, $i = 1, \cdots, M$, satisfies the condition con at its current state, where the condition con can be a condition for a task, i.e., C_j, or a precondition for an action, c_{ikp}.

Further, to do automatic tasking for multi-agent systems, similar to [117], we need to make the following assumptions:

Assumption 1: Each agent can verify if an action has succeeded, failed or if it is running.

Assumption 2: Each agent can verify if a condition is true or false.

Assumption 3: For each goal and for each initial configuration of the agents, there exists a sequence of actions that can be taken by the agents leading to the achievement of the goal. This assumption guarantees that each goal is achievable at least by one of the agents.

Assumption 4: The effect of the dynamic environment can void the accomplishment of the actions at most a finite number of times. This assumption is made to avoid sticking in a live-lock of repeating an action and being voided by the environment over and over, preventing the agent to achieve its goal.

Assumption 5: Given two actions A_i and A_j, if the execution of A_i requires the execution of A_j, A_j must not require the execution of A_i. This assumption prevents deadlocks due to cyclic dependency.

Assumption 6: All actions are ultimately reversible. That is, each action can be undone through a finite sequence of actions.

Assumption 7: For each action, there exists at least one agent to achieve it, which can be accomplished by a low-level controller embedded in the agent in a finite time.

Now, given R, T, F, A, C, and $\hat{C}$, and making assumptions 1– 7, the tasking problem for multi-agent systems can be stated as:

Problem 3. *Consider a Mission consists of several tasks T_j $j = 1, \cdots, N$, to be completed by a set of robots R_i $i = 1, \cdots, M$, that (some of them) are capable of accomplishing the actions A_k, $k = 1, \cdots, L$, within Δt_{ik} time units to achieve the mission. Also, consider that there is no order and dependency among the tasks, other than the order in which tasks are issued (one at a time). Synthesize decentralized BT_i to coordinate the individual robots R_i to collectively achieve a set of tasks T_k.*

3.3.2.5.2 Automatic Behavior Tree Synthesis To address Problem 3, we propose a decentralized method for generating the local BTs by combining a market-based auctioning algorithm with a reactive BT synthesis technique, so that the generated local BTs can collectively satisfy the mission specification. **Task Assignment for Coordination of Multi Agent Systems:** To fairly assign tasks and avoid conflicts, we adopt a two-level market-based auctioning algorithm. Generally in a market-based auctioning, even-though there is a collaboration among agents we assume each agent acts on its own interest, i.e. to maximize the reward or to minimize cost. An auctioning process has four steps, starting with a task announcement by the coordinator (announcement stage), followed by the bidding stage where capable agents send a bid. Based on the cost, the auctioneer selects the best agent (the selection stage) and finalize the auction by forming a contract with the selected agent (contract stage). In the proposed framework the *Mission Controller* (MC) announces a task T_j, where capable agents (agents that can meet C_j), R_i, $i = 1, .., M$, participate in the bid. To complete the task T_{ij}, the candidate agent has to identify the sequence of actions either from the local action bank or by delegation, where these actions are used to estimate the total cost f_{ij} before issuing the bid. Based on the estimated cost f_{ij} from each agent, the MC selects an agent and form a contract. Mathematically, this is equivalent to:

$$\begin{aligned} &\min_{x_{ij}} \sum_{i}^{M} f_{ij} x_{ij}, \quad \forall j \\ &subject\ to \sum_{i}^{M} x_{ij} = 1 \, \forall j \\ &x_{ij} \in \{0, 1\}, \quad \forall i, j \end{aligned} \tag{93}$$

where f_{ij} is the cost of task T_j when handled by R_i and x_{ij} is an indicator that task T_j is assigned to R_i.

If an agent delegate an action to complete a task, then the agent has to act as the auctioneer and perform a second level auctioning to identify a suitable agent. Therefore, the total cost for a task is the sum of local and delegated actions: $f_{ij} = \sum_{k=1}^{L} \hat{a}_{ijk} (\hat{a}_{ik} \hat{f}_{ik} + (1 - \hat{a}_{ik}) f_D(ijk)), \forall i, j$, where $f_D(ijk)$ is the cost of the delegated action A_k for the task T_j by R_i, provided that the involved robots are available to complete the actions at the time they are needed. To check availability of the robot, we introduce the function $\nabla(t_{io}, T_j, A_k)$ where $clk(T_j, A_k)$ represents the time that the action A_k is needed for the task T_j. If $t_{io} = 0$ for $clk \geq clk(T_j, A_k) + \triangle t_{ik}$, then $\nabla(t_{io}, T_j, A_k) = 1$, otherwise it returns 0. In addition, $\nabla(t_{io}, T_j, A_k)$ returns the nearest time slot that the agent R_i

can accomplish an action. This indeed is equivalent to the following minimization:

$$f_D(ijk) = \min_{\hat{x}_{djk}} \sum_{d}^{M} \hat{x}_{djk} \hat{f}_{dk}, \ \forall k$$
$$d = 1 \cdots M, \ d \neq i,$$
$$subject\ to \sum_{d}^{M} \hat{x}_{djk} = 1 \ \forall j, k,$$
$$\nabla(t_{do}, T_j, A_k) = 1 \tag{94}$$

where $\hat{f}_{dk}$ is the cost of action A_k when done by R_d and $\hat{x}_{djk}$ indicates if action A_k of task T_j is assigned to agent R_d or not.

Once an action or a task is assigned to an agent, the availability indicator t_{*o} is updated from 0 to 1 for $\triangle t_{*o}$ using the function $\hat{\nabla}(t_{*o}, \triangle t_{*k})$ to avoid double assignment.

Decentralized Behavior tree synthesis algorithm: The overall procedure to generate the BTs for individual agents is explained in Algorithms 1–3. First, the mission controller announces a task T_j (level-I auctioning). Then, any capable agent estimates the task cost and sends a bid. The estimation of the cost is calculated starting from the goal and recursively identifying the precondition of the successor action until the action can be done at the current state of the robot. Upon receiving the bid from the agents, the MC selects the best agent and form a contract (Algorithm 3). The winning agent R_i synthesizes a BT using Algorithm 1 while Algorithm 2 is used to identify actions locally or by delegation (level-II auctioning) to meet the conditions needed to complete the task.

Algorithm 1: *Main BT Synthesis and Execution*

```
1  function MainBTSynthesisandExecution (C_j) ;
   Input  : C_j : Condition for assigned task of agent i
   Output: T_ij = Syntheisized BT
2  T_ij ← C_j // Start the BT for task T_j from the condition C_j, which is
     used to cheek if the task is completed or not
3  T_all_i ← Parallel (T_ij, T_all_i ) // T_all_i represents all BTs of an agent running in parallel
     to execute multiple tasks including bidding and auctioning
4  while True do
5      do
6          r, ĉ_ik ← Execute(T_ij)
7          if R_i ⊨ C_j then
8              Set x_ij = 0 // Task T_j is completed
9              break // End execution of T_ij
10     while r = Executable;
11     c_if ← GetConditionToExpand(T_ij) //Identify the the reason why T_i is not executable
12     T_ij, T_subtree_ij ← ExpandTree(T_ij, c_if) //Resolve the cause by Algorithm 2
13     while Conflict(T_ij) do
14         T_ij ← IncreasePriority(T_subtree_ij)
```

Assume that the task T_j is assigned to the robot R_i as it can meet the condition C_j. Algorithm 1 then synthesizes the local BTs. Algorithm 1 starts from the "goal" input, which describes the condition for the accomplishment of a "task" indicated by the condition C_j (Line 1). By first

assigning the condition C_j to the BT (Line 2) (this condition will be used to determine if the task is completed or not), the algorithm iteratively updates the BT until a sequence of actions is obtained which as a whole realizes the task and achieves the goal (Lines 4-17). Since each task requires its own BT, to execute multiple tasks, the BTs for each task are composed in parallel with the existing BTs, T_{all_i} (Line 3). In a *do while* loop, the BT actions are tested to determine whether they are executable (Lines 5-11). If the condition $R_i \vDash C_j$ is satisfied by the execution of the BT, the agent is free to accept a new task (Lines 7-10). Otherwise, if the BT is not executable, Line 12 identifies the *cause* of failure, c_{if}. The identified *cause* will become a condition in a subtree to resolve the problem by finding alternative actions or other agents (Line 13), as will be described in Algorithm 2. After updating the BT, due to the addition of a new subtree, $T_{subtree_{ij}}$, a conflict could arise. To resolve the conflict, the function *conflict*(T_{ij}) increases the priority of $T_{subtree_{ij}}$ by moving the subtree toward the left. As an example, in response to avoid an obstacle the robot decides to pick an obstacle (object), but picking up an object has to be done if the robot arm is free.

Algorithm 2 essentially synthesizes a subtree that satisfies the condition c_{if}. In Line 2 of Algorithm 2, the function *GetlocalActionwithPrecondtion* ( ) returns the optimal action, which satisfies the condition c_{if}. If the returned action is not empty, then the identified action A_k along with its preconditions, $\hat{c}_{ikp}$, are composed by a sequence node to form $T_{seq_{ij}}$ (Lines 5-9). Further, $T_{seq_{ij}}$ is composed with $T_{sel_{ij}}$, defined as c_{if} (Line 3), by a selector node, to enforce the execution of $T_{seq_{ij}}$ only in situations where c_{if} is not satisfied (Line 10). To avoid double assignment, the time-line and availability indicators for R_i are also updated (Lines 11-12). However, if no local action exists, the *AuctioningModule* (similar to Algorithm 3) is activated to conduct an auction in pursuit of finding an agent that can accomplish c_{if} (Lines 14-16). Finally, the condition c_{if} is replaced with a sub-tree that can meet c_{if} (Line 17).

Algorithm 2: *Expand Behavior Tree Module For R_i*

```
1  function ExpandBT (T_ij, c_if );
   Input : c_if = condition (cause) for T_j not being executable
   Output: T_ij = Expanded BT
2  A_k ← GetLocalActionwithPrecondtion(c_if) // Identify local actions that satisfy c_if
3  T_sel_ij ← c_if
4  if GetLocalActionwithPrecondtion(c_if) ≠ ∅ then
5  |  c_ik = GetPrecondtionforAction(A_k)
6  |  for c_ikp in c_ik do
7  |  |  T_seq_ij → Sequence (T_seq_ij, c_ikp ) // sequence BT with the condition of action
8  |  T_seq_ij ← Sequence(T_seq_ij, A_k) // Generate a sequence subtree containing action
   |     A_k and its preconditions
9  |  T_sel_ij ← Selector(T_sel_ij, T_seq_ij)
10 |  V̂(t_io, T_j, A_k) // t_io is set to 1 for △t_ik time units
11 |  set x̂_ijk = 1 // Action A_k of T_j is assigned to R_i
12 else
                                ( ) //
13 |  AuctionModule_L2 c_if  If there is no action to meet a condition,
   |     initialize the Level IIAuction Module for delegation
14 T_ij ← Substitute(T_ij, c_if, T_sel_ij) // add the subtree T_sel_ij to T_ij replacing c_if
15 return T_ij, T_sel_ij
```

Algorithm 3 performs an auction to find a suitable agent following a standard market based

Algorithm 3: *Auctioning Module*

1 function AuctioningModule (c_{if}) ;
  Input : c_{if} : *condition to be delegated*
2 $selected_f \leftarrow \emptyset$
3 $Announcing(c_{if})$ //broadcasting condition c_{if}
4 s = *ReceiveSubmission* () // agents with the spesfied action replies
5 $selected_f$ = *Selection*(s) // choose the agent that minimizes cost $\hat{f}_{if}$
6 $Contract(selected_f, c_{if})$

auctioning mechanism (Lines 2-6). The auction terminates with a contract (Line 6).

3.3.2.5.3 case study Single Agent: Search and Delivery UAV mission: The mission objective is to deliver an object o at a specific place marked by m near position p. The UAV has to search for the marking m in close vicinity of p, N_p, before delivering the object o. Then, the problem is given the action bank in Table 18, generate a BT using Algorithms 1 & 2 to achieve the task.

Algorithm 1 starts from the goal, "o at p", i.e., the object o should be at position p, as shown in Fig. 119a. Since initially the goal is not satisfied yet and the generated *BT* (Line 6 of Alg1) is not executable, the function *GetCondtionsToExpand* is called to identify the preconditions (Line 12). From Table 18, the *Deliver* action can meet the precondition of Algorithm1 and hence, the *ExpandBt* function (Line 13) uses this action to update the BT by composing the conditions of *Deliver* action via a sequence node and the goal by a selector node (Lines 4-13 of Alg2) as shown in Fig. 119b. Again since the preconditions, *uav at* N_p and *m is detected*, are not true, they have to be expanded, following the same procedure, by their corresponding actions *MoveTo* and *Detect* as shown in Fig. 119c.

Table 18: Action templates for case study.

Global Action Template			
No	Action	Precondition	Effect
1	MoveTo(*p, path*)	*path is collsionfree*	*uav at p*
2	Detect(*m*)	*uav at* N_p	*m is detected*
3	Deliver(*i, m*)	*uav at Np* *m is detected*	*o at p*

Table 19: Mission tasks expanded using Algorithm 1

Mission			
No	Task	Condition	Sequence of actions
1	T_1	C_1	A_1, A_3, A_2
2	T_2	C_2	A_6, A_4, A_3
3	T_3	C_3	A_5, A_1
4	T_4	C_4	A_4
5	T_5	C_5	A_2, A_8 *or* A_3, A_7

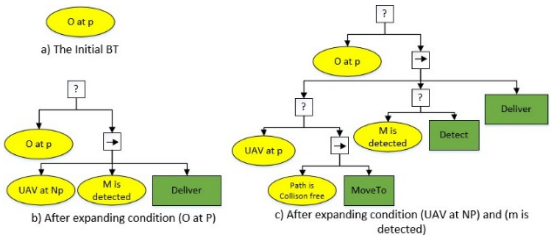


Figure 141: Synthesizing a BT for a UAV to search and deliver an object to a particular position

Resource		
No	Agent	Agent capability - $\{A_k, (\hat{f}_{ij}, \Delta t_{ik})\}$
1	UAV_1	$\{A_1, (0.1, 1)\}$ $\{A_2, (0.2, 2)\}$ $\{A_8, (0.6, 2)\}$
2	[illegible]	[illegible]
3	UAV_3	$A_3,\ 0.7, 1\ ,\ A_4,\ 0.7, 1\ ,\ A_6,\ 1, 4$

Table 20: Agents capability along with the cost $\hat{f}_{ij}$, and duration, Δt_{ij}, of an action A_k

Auctioning steps for assigning the tasks T1, ···, T5					
Step	Task/ Action	time	Auctioneer	Candidates	Contract
1	T_1	1	$MC,\ L1$	$\{UAV_1 : A_2, \hat{f}_{12} = 0.2\}$	-
2	A_3	4	$UAV_1,\ L2$	$\{UAV_2 : \hat{f}_{23} = 0.5,\ clk(T_1, A_3) = [4-6]\}$ $\{UAV_3 : \hat{f}_{33} = 0.7,\ clk(T_1, A_3) = [4-5]\}$	UAV_2
3	T_1		$MC,\ L1$	-	$\{UAV_1 : f_{11} = 0.8,\ clk(T_1, A_*) = [1-8]\}$
4	T_2	2	$MC,\ L1$	$\{UAV_2 : A_3, \hat{f}_{23} = 0.5\}$ $\{UAV_3 : A_3, \hat{f}_{33} = 0.7\}$	-
5	A_6	2	$UAV_2,\ L2$	$\{UAV_3 : \hat{f}_{36} = 1,\ clk(T_2, A_6) = [2-6]\}$	UAV_3
6	A_4	6	$UAV_2,\ L2$	$\{UAV_3 : \hat{f}_{34} = 0.7,\ clk(T_2, A_4) = [6-7]\}$	UAV_3
7	T_2		$MC,\ L1$	$\{UAV_2 : f_{22} = 2.2,\ clk(T_2, A_*) = [2-9]\}$ $\{UAV_3 : f_{32} = 2.4,\ clk(T_2, A_*) = [2-8]\}$	$\{UAV_2 : f_{22} = 2.2,\ clk(T_2, A_*) = [2-9]\}$
8	T_3	3	$MC,\ L1$	$\{UAV_1 : A_1, \hat{f}_{11} = 0.1\}$	-
9	A_5	3	$UAV_1,\ L2$	$\{UAV_2 : \hat{f}_{15} = 0.5,\ clk(T_2, A_5) = [9-12]\}$	UAV_2
10	T_3		$MC,\ L1$	$\{UAV_1 : f_{13} = 0.6,\ clk(T_3, A_*) = [10-15]\}$	$\{UAV_1 : f_{13} = 0.6,\ clk(T_3, A_*) = [3-15]\}$
11	T_4	4	$MC,\ L1$	$\{UAV_3 : A_4, \hat{f}_{34} = 0.7\}$	$\{UAV_3 : f_{34} = 0.7,\ clk(T_4, A_*) = [4-8]\}$
12	T_5	5	$MC,\ L1$	$\{UAV_1 : f_{15} = 0.8,\ clk(T_5, A_8) = [9-12]\}$ $\{UAV_2 : f_{25} = 0.9,\ clk(T_5, A_*) = [13-16]\}$	$\{UAV_2 : f_{25} = 0.9,\ clk(T_5, A_*) = [5-12]\}$

Table 21: Task Assignment: T_1 represents task 1, "$MC,\ L1$" represents level-one auctioning by the mission controller and "$UAV_*, L2$" represents level-two auctioning by an agent

Multiple Agent Multiple Task: Given multiple, $R = \{UAV\ 1_1, UAV_2, UAV_3\}$, along with their capabilities described by the action bank in Table 20, our aim is to synthesize BTs in a decentralized way to satisfy the tasks listed in Column 2 of Table 19. To avoid repeating the procedure of generating a sequence of actions for each task, each task is expanded to a sequence of actions which is described in Column 3 of Table 21. Then, following Algorithms 1 - 3, the details of the BT generation are given in Table 21. As an example, consider the expanded task T 1 with the sequence A_1, A_3, A_2 (Row 1 of table 21), where UAV_1 is the only candidate and winner of T_1 (since only UAV_1 can do the last action, i.e., A_2). However, in task T_1, action A_2 cannot be executed by UAV_1 because UAV_1 cannot perform action A_3 which precedes action A_2. Hence UAV_1 initiates a level-two auctioning to assign action A_3 (Row 2 of table 21), where UAV_2 wins the auction with minimum cost. Now the final action A_1 in T_1 can be handled by UAV_1. This concludes the action assignment for the task T_1 with a total cost of 0.8 in the time interval $[1, 8]$ (Row 3 of Table 21).

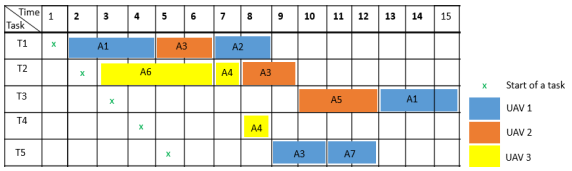


Figure 142: Task assignment along a timeline

The assignment of all tasks follows the same procedure. Sometimes it may be the case that agents are not available at the time a task is requested, like in T_3. When T_3 is assigned, even though UAV_2 is free, the available time before A_3 of T_1 starts execution is not enough to complete the action A_5 of T_3 completely. So action A_5 is deferred to a later time (Row 9 of Table 21). This can be seen more clearly in Figure 120 which shows the tasks and UAVs' assignments along with time axis. The final task, T_5, can be accomplished by UAV_1 and UAV_2 in a non-unique way. This shows that tasks are not necessarily a fixed sequence of actions rather multiple capable agents can do a task in different ways to meet the goal.

3.3.3 Project Progress for Task T3-3 (Developing Formal Methods for Human-robot Collaboration)

3.3.3.1 Period of Performance under Task T3-3

Start Date: August 2017

Conclusion Date: April 2021

Faculty lead: Dr. Ali Karimoddini

3.3.3.2 General Description of Task T3-3

Sufficient information about the components of a large scale system does not always exist. Furthermore, uncertainty in the model parameters and external disturbances may challenge model-based analysis and synthesis techniques. Therefore, to further improve the proposed verification approach, learning techniques will be augmented simultaneously and incrementally to capture the non-modeled parameters of the system while evaluating the performance of the system. Hence, to develop Human-Robot Cooperation (HRC) using formal method analysis in the multi-agent system considering the human operator in the loop has been reviewed to establish the cooperative control followed by an agent, in a model-free system.

One of the challenging problems in HRC settings is to characterize physical and cognitive workloads of human operators and the robots and to assess their impacts on the establishment of trust for developing a safe and optimized approach for task sharing amongst a team of humans and robots. The goal of this task is to develop the systematic approaches for analyzing human factors in HRCs involving single/multi-robot(s). First, a novel time-driven human cognitive performance modeling approach for human-robot collaborative actions is mathematically developed in terms of human cognitive workload, robot performance, and human physical performance. Novel about the proposed model is its ability to relate human cognitive workload and the task complexity to a uti- lization factor, which is functionally correlated with the robot's mistake probability. Second, the

developed human cognitive performance model is used as a basis for proposing a computational trust model to estimate the evolution of human operator's trust in robot operation. For this purpose, a time-driven performance-aware mathematical model for trust in the robot operation is proposed for an HRC framework. The human operator's performance is modeled based on both the physical and cognitive performances, while the robot performance is modeled over its unpredictable, predictable, dependable, and faithful operation regions. This developed time-driven mathematical model for trust is then extended to HRCs with multiple robots. Finally, the impact of learning capability of robot on human cognitive performance and the evolution of trust on robots are modeled. The developed models are validated via simulations of different case-studies confirming that a learning capability allows robots to reduce human workloads and improves human performance and trust on robots.

3.3.3.3 Objectives for Task T3-3 over the Course of Project

- **Objective Name:** Performance-Aware Trust Modeling within a Human-Multi-Robot Collaboration Setting

  Objective Type: Research

  Objective Description: The modeling of human performance and trust in the robot oper- ation within an HRC is a challenging task. To address this problem, a novel time-driven mathematical model was proposed to capture the human operator's performance and trust in the robot's operation within an HRC framework. The proposed models were based on the performance of both the robot and the human operator. Both physical and cognitive performances of the human operator were considered for modeling the human operator's performance.

  Impact: By developing the human performance and trust model within an HRC setting, formally non-model parameters were captured that exist within the human-in-the-loop system. The models can be extended to the complex dynamic system including autonomous vehicles, manufacturing setting, and so on, where human is taking part physically and cognitively. The derivable of this task, on the other hand, has made a new addition to the human-centered control by leveraging human science and robot information to control the system operation that underlies human behaviors during the interaction.

3.3.3.4 Progress Against Planned Objectives in Task T3-3

The objective of this task is accomplished. Several models are developed to:

- mathematically represent a time-driven human cognitive performance model for Human-robot Collaboration (HRC) actions considering the human cognitive performance as a function of human cognitive workload, robot performance, human physical performance, and complexity of shared tasks.
- capture a time-driven performance-aware mathematical model for trust in the robot operation for a Human-robot Collaboration (HRC) setting taking into account the human operator physical and cognitive performances, the robot performance, and complexity of shared tasks.

- extend the time-driven performance-aware mathematical model for trust in a human-multi-robot collaborative actions considering the trust as a function of human cognitive workload, human physical and cognitive performances, the robots' performances, and complexity of shared tasks.

Besides, several mathematical formulations are proposed to illustrate the capability of the models in solving complex problems with some assumption. Single or multiple robots are involved in those scenarios to analyze the time-variant nature of human performance and non-linear characteristics of human trust in the robot operation.

3.3.3.5 Technical Accomplishments in Task T3-3

To accomplish the objective of this task, this research proposes a novel time-variant mathematical model for an effective human cognitive performance model within an HRC framework. It considers human cognitive performance as a function of human cognitive workload, robot-added workload, and human physical workload. The human cognitive workload is formulated in terms of maxi- mum and minimum bounds of cognitive workload, human utilization factor, and the complexity of tasks. Human utilization has been quantified and directly related to the robot's mistake probabil- ity that defines the robot performance. The proposed model considers the cognitive workload as the dominant factor in quantifying human cognitive performance. Further, the developed model takes into account the effect of human physical and robot performances on human cognitive perfor- mance. Simulation results and performance analysis have been provided for a real-life scenario in the manufacturing industry. Besides, the effect of change of robot performance on human cognitive performance is investigated. In particular, the model shows that if the robot can learn from its mistakes and improve its performance, human cognitive performance will be enhanced.

This research work also proposes a novel time-driven mathematical model to capture the human operator's trust in the robot's operation within an HRC framework. The proposed model is based on the performance of both the robot and the human operator. Further, both physical and cogni- tive performances of the human operator have been considered for modeling the human operator's performance. The human operator's trust in the robot's operation is influenced by the robot-added workload, task complexity, and the human operator's physical and cognitive workloads. The developed trust model considers the change of the human operator's performance over time due to factors like fatigue and recovery. In comparison to the static trust models in [119–121], the proposed model is able to capture the trust in a dynamic way. In addition, the proposed model considers both the human operator's physical and cognitive performances for modeling trust which is more realistic than the models that consider only the physical performance for the human operator [122]. Further, simulation results have been provided to show the trust development for the manufacturing industry that can be applied to other situations that involve HRC. The results show that as the robot learns about the task handling at the workspace through the guidance provided by the human operator, the human operator's trust in the robot's operation for decision-making improves over time.

The contribution of this task is, therefore, the development of a novel time-driven mathematical model for an effective human cognitive performance model in order to capture the human operator's trust in the multi-robot operation. The proposed cognitive performance model considers the cognitive workload as the dominant factor in quantifying human cognitive performance and is based on the performance of both the multi-robot and the human operator, which takes into account the

cognitive performance as a function of human physical workload, cognitive workload, and robot-added workload. Further, the robots-added workload, task complexity, and the human operator's physical and cognitive workloads influence the human operator's trust in the robot's operation. The proposed trust model also studies the change of the human operator's performance due to factors like fatigue and recovery and is able to capture the evolution of trust in comparison to the static trust models in [119–121]. Besides, the proposed model within human-multi-robot setting considers both the human operator's physical and cognitive performances for modeling trust that is more real- istic than the models that consider only the physical performance of a human operator and a robot setting [122] or workload-adaptive cognitive approach in the teaming of manned and unmanned vehicles [123]. Simulation results for the human operator and the robots' performance analysis have been provided for a real-life scenario in a manufacturing case-study to show the efficiency of the developed trust model. In addition, the effect of robots' learning capability on the established trust on robots is investigated. The results show that as the robots learn about the task handling through the guidance provided by the human operator, the human operator's trust in the robots' operation for decision-making improves over time.

In summary, here is a list of the main contributions of this task:

- A novel mathematical representation of dynamical human cognitive performance model has been proposed for an HRC framework based on the human operator's physical workload, the robot added workload, and the human operator's cognitive workload. The developed model has been verified via simulation for a shared task between a human operator and a robot. Simulation results have been provided to validate the proposed collaboration model and analyze the effects of variations of the involved parameters.

- A novel time-driven performance-aware mathematical representation of human operator trust in the robot operation has been developed for an HRC framework based on the human operator's physical workload/physical performance, the human operator's cognitive workload/cognitive performance, and the robot added workload/robot performance. The developed model has been verified via simulation for a shared task between a human operator and a robot. Simulation results have been provided to validate the proposed collaboration model and analyze the effects of variations of the involved parameters.

- A human-multi-robot performance-aware trust model has been developed considering the human operator's physical performance and workload model, human operator's proposed cognitive performance and workload model, human operator' performance model, and robot added workload/robot performance. The developed trust model for human-multi-robot performance has been verified via simulation for a shared task among a human operator and multiple robots. Simulation results have been provided to validate the proposed collaboration model and analyze the effects of variations of the involved parameters.

- The developed models are extended by considering the robots with a learning capabilities to capture the impact of robots' learning on human cognitive performance and the evolution of human trust on robots.

Next a novel time-variant human cognitive performance modeling for human-robot collaborative actions is discussed.

3.3.3.5.1 Human-Robot Collaboration Setting Consider a human-robot collaboration (HRC) scenario, where a human and a robot are collaborating to perform a shared task. In this HRC setting, the robot can be utilized to perform repeated routine work while human intelligence can handle more complex tasks such as decision-making. For example, the human operator can supervise the robot to complete the assigned tasks by guiding the robot via cognitive signals and helping the robot by physically changing the object orientation appropriately so the robot can pick the object easily. The robot performs the task based on instructions received from the human in the form of cognitive signals. The more the robot can perform a task successfully, the less human supervision is required. Conversely, if the robot mistake rate increases, more assistance from the human operator is needed. Fig. 121 shows a symbolic diagram for an HRC setting, in which a robot transfers heavy objects from the source conveyor belt to the destination (packaging) conveyor belt. The human operator supervises the robot and observes (takes feedback from) the robot's actions, the source where the robot has to pick objects, and the destination where the robot delivers objectives for packaging. If the robot commits a mistake in picking/placing the objects from/in wrong conveyors, the human operator sends the robot corrective cognitive signals (e.g., in the form of push-button, vocal, or EEG signals) to correct its actions and guide it for successful completion of the task. Once the object is available on the destination conveyor, the human operator controls the conveyor to transfer it out of the workspace. Robot performance is measured in terms of its capability to accommodate a human operator's instruction(s). Therefore, the robot performance has a significant impact on the human cognitive workload during the collaboration between the human operator and a robot. The robot performance, $R_P(t)$, for a given time instant t depends on its success in the completion of tasks, which can be modeled as:

$$R_P(t) = R_{P,max} - \frac{S_R(t-1) - (1 - P_{mR})D_R(t-1)}{S_R(t-1)} \tag{95}$$

where $R_{P,max}$ is the maximum value of the robot performance, P_{mR} is the robot's mistake proba- bility, $S_R(t-1)$ is the source rate (the feeding rate of source conveyor) and $D_R(t-1)$ is the delivery rate (percentage of items being handled by the robot and put on the destination conveyor) at the preceding time instant.

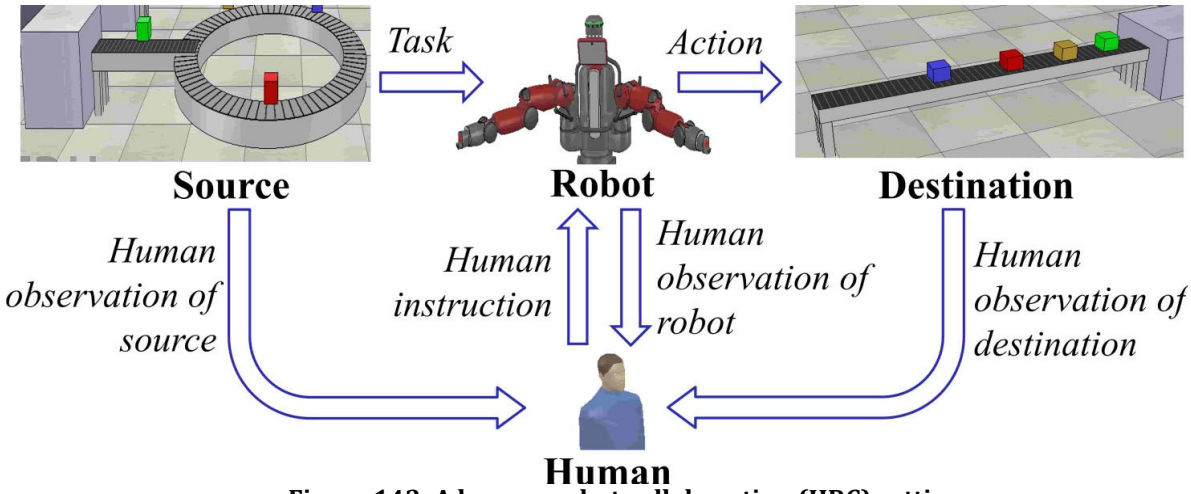


Figure 143: A human-robot collaboration (HRC) setting.

3.3.3.5.2 Human Performance Modeling: In this section, we model human performance. In particular, we provide a model for human cognitive performance, which indicates its cognitive capability to perform the mental work [124]. A human's cognitive performance is at a maximum when it is subjected to a minimum workload, no fatigue, and maximum robot performance. On the other hand, human performance will be minimum when the human is experiencing maximum workload and fatigue level, and minimum robot performance. If the robot did not perform the immediate preceding task satisfactorily, the human operator instructs it to make a correction. The interactions between human and robot increase human utilization factor and cognitive workload.

Human Physical Performance Model A human's physical performance can be related to the muscular contraction and expansion system and can be tied to the fatigue level of the muscles and their recovery. The fatigue and recovery models of muscles affect the human physical performance that can be modeled as [124]:

$$P_P(t) = \frac{F_{max,iso}(t) - F_{th}}{MVC - F_{th}} \tag{96}$$

where $P_P(t)$ is the human physical performance at time instant t and F_{th} stands for the threshold force which is calculated at the equilibrium point where the fatigue and the recovery balance out each other. $F_{max,iso}(t)$ stands for the maximum value of isometric force. The isometric force, $F_{iso}(t)$ is generated when the human muscles apply force but the length of muscles does not change [124]. *Maximum Voluntary Contraction (MVC)* stands for the maximum value of isometric force that one can produce at rest or the initial state (at zero-level of fatigue) [124]. Clearly, $F_{max,iso}(t)$ decreases over time due to muscle fatigue. Adopted from [124–126], we use the following first-order Euler approximation to represent the dynamic calculation of the maximum isometric force:

$$F_{max,iso}(t) = F_{max,iso}(t-1) - C_f F_{max,iso}(t-1)\frac{F(t-1)}{MVC} + C_r(MVC - F_{max,iso}(t-1)) \tag{97}$$

where C_f and C_r stand for the fatigue and recovery constants, respectively, and $F(t)$ denotes the real-time applied force that reduces over time due to increase in fatigue levels.

In this model, continuous-time fatigue and recovery processes are used to represent the dynamic evolution of the maximum isometric force. It can be verified in Eq. 97 that fatigue increases when muscles continuously apply the force. On the other hand, when no force is used or if the applied force is relatively small, the muscles will recover, i.e., $MVC - F_{max,iso}(t-1)$ will be increased. $F_{max,iso}(t)$ is maximum when the human operator starts the task, i.e., $F_{max,iso}(t=0) = MVC$. Therefore, based on Eq. 96, $P_P(t=0) = 1$ at the beginning when $F_{max,iso}(t=0) = MVC$ [124], but then it reduces to zero when $F_{max,iso} = F_{th}$. Further, it can be verified that in Eq. 96, the isometric force is affected by the fatigue level. Higher fatigue levels result in lower isometric force, which in turn will reduce the overall human performance values.

Human Cognitive Workload Model: The cognitive workload refers to the amount of mental work to be performed in a given period. A human operator's performance degrades for high cognitive workloads and/or while handling complex tasks. During the inactive mode (when no cognitive work is performed), the human operator's cognitive performance level gradually increases. This recovery process can improve the cognitive performance up to the Optimum Level of Arousal (OLA) point [124, 127, 128]. Here, we model the human operator's cognitive workload for a given time, $C_w(t)$ as a function of the complexity of the task(s) to be performed and the human operator's utilization factor as:

$$C_W(t) = (C_{W,max} - C_{W,min})\left(\frac{u(t)}{1 - C(t)}\right)^{1-C(t)}\left(\frac{1 - u(t)}{C(t)}\right)^{C(t)} + C_{W,min} \quad (98)$$

where $C_{W,min}$ and $C_{W,max}$ are the minimum and maximum cognitive workloads respectively, which may vary from person to person, depending on the individual's capabilities to handle the tasks. $C(t)$ is the complexity of the task (a relative value between 0 to 1) being handled at time t and $u(t)$ is the human operator's utilization factor which can be captured as:

$$u(t) = u(t-1) + \triangle u(t) \quad (99)$$

where $\triangle u(t)$ stands for the change in the utilization factor which is a function of the robot's mistake probability for a given time as:

$$\triangle u(t) = \frac{P_{mR}(t) - u(t-1)}{\tau} \quad (100)$$

where τ is a positive integer number representing the time constant. In other words, τ is the time that the human operator takes to respond to the changes in the robot's mistake probability, thereafter the effect of changes in robot's mistake probability appears in human operator utilization factor.

Remark 1. *If the human operator is doing the same task all the time, then the value of $C(t)$ will be a constant number, otherwise its value changes depending on the complexity of the task being handled at each time instant.*

Remark 2. *In the proposed HRC framework, individual tasks are assumed to be independent events. Further, a task either can be failed by the robot, which requires human operator's intervention, or it will be successfully handled by the robot, which does not need the human operator to be utilized. This allows us to use a binomial form in Eq. 98 to describe the human cognitive workload based on successfully handled tasks and failed ones.*

Human Cognitive Performance Model: Human(s) cognitive workload primarily impact(s) its cognitive performance. Further, human physical workload or the additional workload due to mistakes of the robot affect the cognitive performance of the robot as well. Incorporating all these factors, human cognitive performance, $C_P(t)$ for a given time, can be modeled as:

$$C_P(t) = C_{P,max} - \alpha C_w(t) - \beta H_w(t) - \gamma H_R(t) \quad (101)$$

where $C_{P,max}$ is the maximum cognitive performance for a given time, $H_w(t)$ is the human physical workload for a given time, $H_R(t)$ is the additional workload added due to mistakes of the robot, and α, β, and γ are positive real numbers with $\alpha + \beta + \gamma = 1$.

The human physical workload and the additional workload due to mistakes of the robot can be indirectly estimated from the human physical performance and the robot performance for a given time is as follows:

$$H_W(t) = P_{P,max} - P_P(t) \quad (102)$$

$$H_R(t) = R_{P,max} - R_P(t) \quad (103)$$

where $P_{p,max}$ and $R_{P,max}$ are the maximum human physical performance and robot's maximum performance, respectively. $P_P(t)$ and $R_P(t)$ have been derived in Eq. 96 and Eq. 95, respectively.

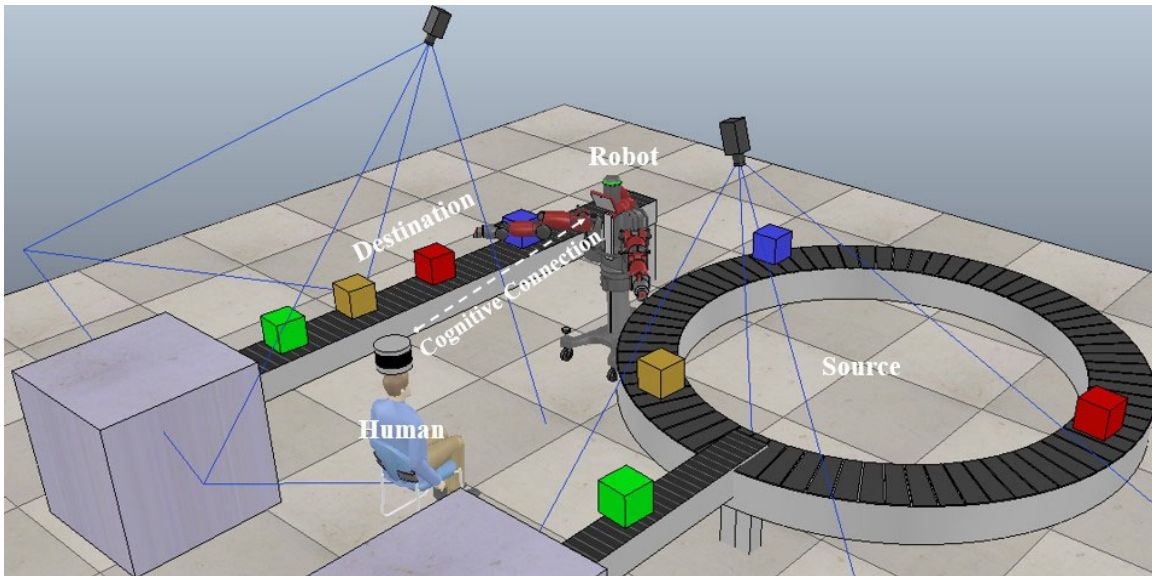
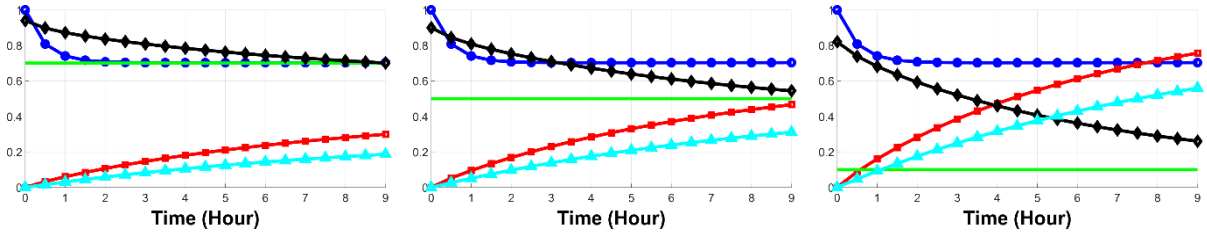


Figure 144: A human-robot collaboration scenario: robot transfers heavy objects from the "source" conveyor belt to the "destination" conveyor belt, and the human operator guides the robot to successfully complete the task.

3.3.3.5.3 Analyzing the Human Cognitive Performance within the Proposed HRC Setting Consider a manufacturing workspace in which a human operator and a robot can collaboratively work to transfer the produced items as shown in Fig. 122. The source conveyor carries the incoming boxes to the workspace, and the destination conveyor moves the inspected items to the packaging area. The robot's task is to physically move the objects from source conveyor to packing conveyor based on the human's instructions. The human operator supervises and instructs the robot to complete its task correctly. The human operator is assumed to help the robot by physically changing the object orientation appropriately so the robot can pick the object easily. Also, the human operator is involved in (minimal) physical activity, such as regulating the speed of the destination conveyor, maintaining the logs, etc.

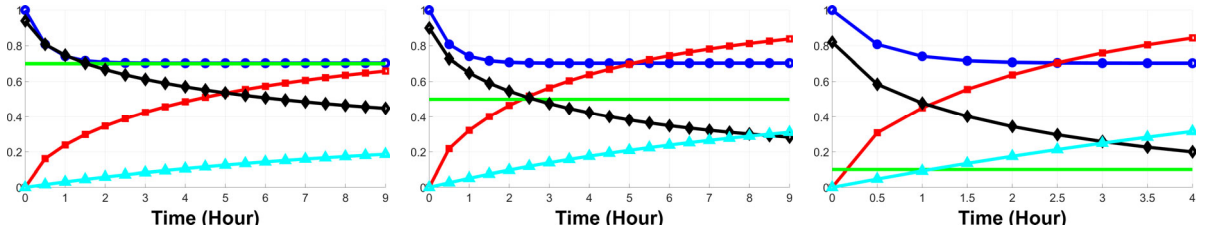
To simulate this scenario, we use the derived set of equations from Eq. 95 to Eq. 103 in which the simulation parameters are chosen as summarized in Table 22. Some of the parameters' values including C_f, C_r, MVC, and F_{th} are set similar to those in [124]. Since in human cognitive performance, the human cognitive workload is a dominant factor, in Eq. 101, the value of α is selected larger than the values for β and γ.

The simulation runs over an operating period of ten hours with the sample time of 30 minutes. The minimum level of human cognitive performance, $C_P(t)$, at any given hour is assumed to be more than 0.2, below which the human operator is considered to be incapable of performing the cognitive workload. In order to maintain simplicity, the robot's mistake probability has been assumed to be constant over time.



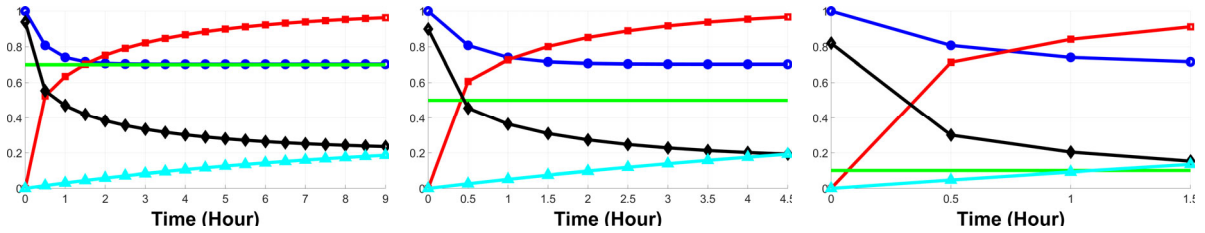
(a) $R_P(t) = 0.7$ (b) $R_P(t) = 0.5$ (c) $R_P(t) = 0.1$

Figure 145: The change of physical performance (blue line), cognitive workload (red line), robot performance (green line), cognitive performance (black line), and utilization factor (cyan line) are represented over time for the task complexity, $C(t) = 0.1$



(a) $R_P(t) = 0.7$ (b) $R_P(t) = 0.5$ (c) $R_P(t) = 0.1$

Figure 146: Figure 146: The change of physical performance (blue line), cognitive workload (red line), robot performance (green line), cognitive performance (black line), and utilization factor (cyan line) are represented over time for the task complexity, $C(t) = 0.4$



(a) $R_P(t) = 0.7$ (b) $R_P(t) = 0.5$ (c) $R_P(t) = 0.1$

Figure 147: Figure 147: The change of physical performance (blue line), cognitive workload (red line), robot performance (green line), cognitive performance (black line), and utilization factor (cyan line) are represented over time for the task complexity, $C(t) = 0.7$.

Table 22: Human Cognitive Performance Simulation Parameters

Name & Symbol of Parameters	Value
Fatigue constant, C_f	10^{-4} ×
Recovery constant, C_r	2.4 10^{-4}
MVC	200
Minimum threshold force, F_{th}	151.9
Time constant, τ	10
Max cognitive workload, $C_{W,max}$	1
Min cognitive workload, $C_{W,min}$	0
Cognitive workload co-efficient, α	0.7
Physical workload co-efficient, β	0.1
Additional robot workload co-efficient, γ	0.2

The simulations are performed using three different task complexity values, $C(t)$ = 0.1, 0.4, and 0.7, each simulated for three different values of robot performance, $R_P(t)$ = 0.7, 0.5, and 0.1. The simulation results are provided using three different task complexity values, $C(t)$ = 0.1, 0.4 and 0.7 in Fig. 123, Fig. 124, and Fig. 125 respectively for each $R_P(t)$ = 0.7, 0.5, and 0.1.

Analyzing the results of the three subplots in Fig. 123, it can be seen that for a fixed value of task complexity, the human cognitive performance, $C_P(t)$, decreases, while its utilization factor, $u(t)$, increases for decreasing values of robot performance, $R_P(t)$. For higher values of $R_P(t)$, higher values of $C_P(t)$ are observed, at the corresponding time instants. For example, in the case when $R_P(t)$ = 0.1, the robot is making mistakes frequently and requires assistance from the human operator much often, resulting in lower values for $C_P(t)$ than those for the case when $R_P(t)$ = 0.7, at the corresponding time instants. Similar trends are observed in the comparison of the three subplots in Fig. 124 and Fig. 125.

The impact of task complexity on the performance values can be analyzed by comparing the results in the subplots in Fig. 123a, Fig. 124a, and Fig. 125a for similar values of robot performance. The results reveal that for a fixed value of robot performance, the values for human cognitive performance, $C_P(t)$, decrease, while the utilization factor, $u(t)$, increases for increasing values of task complexity, $C(t)$. For lower values of $C(t)$, higher values of $C_P(t)$ are observed at the corresponding time instants. For example, in the case when $C(t)$ = 0.7, the robot requires frequent instructions from the human operator, resulting in a lower value for $C_P(t)$ than the case when $C(t)$ = 0.1, at the corresponding time instants. Similar trends are observed in the comparison of the other corresponding subplots shown in these figures, i.e., Fig. 123b, Fig. 124b and Fig. 125b as well as Fig. 123c, Fig. 124c and Fig. 125c.

Furthermore, it can be observed that in the subplots Fig. 124c, Fig. 125b and Fig. 125c, the maximum operating time t_{max} is less than the ninth hour. This is because the human operator's cognitive capabilities have exhausted already, i.e., the overall human cognitive performance values, $C_P(t)$, are below 0.2.

3.3.3.5.4 Conclusions In this paper, a dynamical human cognitive performance model has been proposed for a Human-Robot Collaboration (HRC) framework. This model has been simulated for a shared task between a human operator and a robot. Simulation results have been used to

validate the proposed collaboration model and analyze the effects of variations in the values of the involved parameters. A mathematical representation of human cognitive performance has been provided in terms of the associated human physical workload, robot added workload, and human cognitive workload. Also, the complexity of tasks at hand and the associated human utilization factor for different values of human capabilities have been considered to model the HRC framework.

Simulation results show that for a fixed complexity of the task, a decrease in robot performance increases human utilization factor and the associated cognitive workload, which in turn degrades the human cognitive performance. It has also been shown that human cognitive workload decreases as the robot performance improves. For fixed values of robot performance, the human utilization factor increases as the task complexity increases which in turn increases the human cognitive workload and degrades the human cognitive performance.

3.3.4 Project Progress for Task T3-4 (Formulation of LCS to Learn Emergent Behaviors)

3.3.4.1 Period of Performance under

Task T3-4 Start Date: Fall 2016

Conclusion Date: 15 May 2021

Faculty lead: Abdollah Homaifar, Ph.D

3.3.4.2 General Description of Task T3-4

Learning Classifier Systems use reward-mechanism to capture internal processes of a system based on the system's environmental variables. Here, the environment for the LCS is the LSASV under test and all of the entities and features in the simulation environment. Unlike a single agent, one major challenge to address when dealing with multiple agents is a methodology to standardize the different inputs of the different components of the LSASV for computer processing. This will be handled using advanced data normalization techniques. The inputs from the LSASV (like position, heading, speed, etc.), the messages it transmits as well as the messages and rules it transmits, can be coded in the form of string bits or classifiers which are the fundamental units of information in the LCS implementation. The outputs or actions of the LCS correspond to the output (emergent behavior) of the LSASV. The rules that the LCS learns are the ones that govern the transitions between emergent behavior states of the LSASV. The solution domain comprises a set of rules that collectively give a model of the LSASV's interaction with its environment as shown in Figure 3. LCS needs a trainer to provide a reinforcement signal in the form of a reward or punishment using just a sufficient number of rules (called training data). The evaluator receives the response of the LSASV for a given input and compares it with the corresponding action of the LCS system and generates feedback to the LCS block. The intention here is to train the LCS block using this known set of rules. Once sufficiently trained, the LCS can be utilized for predicting correct state transitions for previously unknown input scenarios.

3.3.4.3 Objectives for Task T3-4 over the Course of Project

- **Objective Name:** Development of Online Clustering-based Technique for Classifying Emergent Behaviors of LSASV

Objective Type: Theoretical development and simulation implementation.

Objective Description: Provide Objective Description

Impact: Provide the impact

3.3.4.4 Progress Against Planned Objectives in Task T3-4

Due to the dynamic and complex environmental conditions in real-world scenarios, the characteristics of autonomous vehicles' behaviors usually change over time and LCS can not fully capture the non-stationary patterns of emergent behaviors. Therefore, we formulated it as an online classification problem and developed a clustering-based stream data classification framework. With the developed clustering-based data stream classification framework, Task 3-3 is accomplished and the technical accomplishments are summarized in the following subsection.

3.3.4.5 Technical Accomplishments in Task T3-4

Considering the exponential growth of data generated from autonomous vehicles, non-stationary data stream classification has been extensively investigated in recent years. Unlike traditional data classification problems, the non-stationary nature of data streams requires the learning algorithms to cope with the change of data patterns over time. The infinite length and high dimensionality of data streams pose another layer of challenge in terms of memory and time complexity. More importantly, the scarcity of data labels and expensive labeling efforts also challenged the existing techniques to classify the non-stationary data stream.

Generally, the non-stationary nature of the data streams can be captured by two phenomenons [129, 130]: (1) *concept drift*, and (2) *concept evolution*. The *concept drift* refers to the change of the existing data distributions over time and it usually appears as the change in the class boundary. On the other hand, *concept evolution* arises when unknown concepts appear in the data stream and comes as novel class distributions. To handle *concept drift*, most of the existing approaches employed a forgetting mechanism in the learning algorithm to monitor the model performance using the latest knowledge from the data stream [131–136]. The labels of the data stream are required to handle the *concept drift*. However, due to the fact that the labeling of data streams is both time and resource consuming, it is necessary to develop a more effective labeling procedure in addressing the *concept drift*. For *concept evolution*, the two-phase procedure is widely used in the literature [137]. First, a warm-up phase is initiated to develop a classification model using the initial training data. Then, the initially developed model is used to classify the incoming data samples and distinguish novel classes. With the two-phase procedure, many *concept evolution*-based data stream classification techniques are developed [138–142]. However, these approaches assume that classes are separated by a distinct boundary function and ignore the potential overlap among classes. Similar to the *concept drift* methods, the scarcity of data labels and expensive labeling costs are the other two technical challenges in the literature. In summary, the limitations of the existing data stream classification methods are described below:

- Initial model training: most of the existing approaches require an initial labeled set to create a model for identifying novel concepts from the data stream. However, the real-world data streams are unlabeled and very little knowledge is available initially.

- Overlap among classes: many data stream classification frameworks assume that samples from the same class should be close to each other. This assumption may not hold when some classes are highly overlapped.
- Label scarcity and costs: the unrealistic assumption on the availability of data labels is widely used in state-of-the-art techniques. Solutions are required for handling the label scarcity and high labeling costs in data stream classification tasks.

3.3.4.5.1 Problem Formulation Let DS be an unknown data stream and CH_t is a chunk of data from DS such that $DS = \bigcup_{t=1}^{T} \{CH_t\}$ where t and T refer to the time index of the current and final data chunk, respectively. For each data chunk, $CH_t = \bigcup_{i=1}^{|CH_t|} \{x_t^i | x_t^i \in \mathrm{R}^m\}$ where x_t^i denotes a data sample in CH_t and m is the dimension of x_t^i. Assume $HS_t = \{GS_t, LS_t\}$ is the summary of a data stream where GS_t and LS_t represent the macro-level and the micro-level summary of DS. We use C_t^i to represent the i^{th} cluster center of DS at time t, and $SC_t^{i,j}$ refers to the j^{th} sub-cluster center of cluster i at time t. The notation $y_t^{i,j}$ denotes the class label of $SC_t^{i,j}$. Considering the non-stationary nature of data streams, the following assumptions are considered:

- The characteristics of data streams can change abruptly or gradually.
- Multiple novel concepts may arise simultaneously.
- Overlapped classes will appear over time.

3.3.4.5.2 Overview of Clustering-based Data Stream Classification Framework With the mathematical notations and assumptions described above, an algorithm description of the developed clustering-based data stream classification framework, namely CDSC-AL framework, is presented in Algorithm 4. As shown in algorithm 4, CDSC-AL framework extended the recently developed density-based stream clustering algorithm, namely dynamic fitness proportionate sharing (DFPS-clustering) [143], to perform the classification of data streams considering several aspects. First, a new merge procedure between clusters from the incoming data chunk and historical clusters is employed and the modified merge procedure is used to detect novel classes and drifted classes. Second, two levels of cluster summary are maintained continuously to reflect the characteristics of the data stream through an active learning procedure. Third, an effective classification procedure using the k-nearest-neighbor (KNN) rule [144] is introduced to classify the incoming data chunk based on the summary and queried labels. Additionally, the overlap among classes is addressed by exploring the sub-cluster information from the micro-level summary.

3.3.4.5.3 Concept Drifts and Evolution Detection through Clustering To capture the non-stationary property of data streams, we modified the DFPS-clustering algorithm substan- tially by employing a new cluster merge procedure between the historical clusters and new clusters. Then, we use the new DFPS-clustering method to distinguish novel concepts from drifted concepts. Algorithm 5 summarizes the new cluster merge procedure for the detection of drifted and novel concepts.

As shown in Algorithm 5, the new merge procedure utilizes the density of boundary instances to decide whether a merge should happen between a historical cluster and its neighboring cluster from

Algorithm 4: An overview of CDSC-AL framework using the modified DFPS-clustering

Input: DS
Parameters: C_t: the set of clusters at time t; CH_t: the current data chunk; C_{CH_t}: a set of clusters discovered in CH_t; Q_t: a small set of samples for active label querying in CH_t; Y_{Q_t}: the label set of the queried samples Q_t; Y_{CH_t}: the label set for CH_t.
Output: HS_t
while $t = 1$ *to* T **do**
    Conduct recursive density evaluation on CH_t and rank all samples of CH_t according to their density values
    Perform the search of possible clusters in CH_t
    Merge highly overlapped clusters to obtain C_{CH_t}
    if $t == 1$ **then**
        $HS_t \leftarrow \emptyset$, $C_t \leftarrow C_{CH_t}$ $[Q_t, Y_{Q_t}]$ = ActiveQuery(HS_t, C_t, CH_t)
        Y_{CH_t}=Classify(HS_t, Q_t, Y (Q_t), CH_t)
        HS_t=ClusteringModel(HS_t, C_{CH_t} , CH_t)
    else
        C_t=CheckMerge(HS_t, C_{CH_t} , CH_t)
        $[Q_t, Y_{Q_t}]$=ActiveQuery(HS_t, C_t, CH_t)
        Y_{CH_t}=Classify(HS_t, Q_t, Y_{Q_t} , CH_t)
        HS_t=ClusteringModel(HS_t, C_{CH_t} , CH_t)

Algorithm 5: New cluster merge procedure

Parameters: L_{NC}: a list of paired clusters between a historical cluster and its neighboring clusters in CH_t, X_B: a set of boundary samples between each pair of clusters in L_{NC}
Function: CheckMerge(HS_t, C_{CH_t}, CH_t)
Identify the paired neighboring historical clusters in HS_t for C_{CH_t} to obtain L_{NC}
Extract X_B for each pair of neighboring clusters in L_{NC}
Evaluate the density of X_B and check for the density drop for each pair of neighboring clusters
Merge each pair of neighboring clusters when there is no density drop in X_B
Mark unmerged clusters as *novel clusters* and merged clusters as *updated clusters*
Validate the existence of *novel clusters* using Remark 3
Return C_t = [*novel clusters*, *updated clusters*]

CH_t. Then, it generates two types of clusters: (i) *novel clusters* and (ii) *updated clusters*. Clus- ters that are not merged with historical clusters are considered as *novel clusters* while the merged clusters are defined as *updated clusters*. Based on these two types of clusters, the novel concepts are captured by *novel clusters* and drifted concepts are identified as *updated clusters*. To validate the existence of *novel clusters*, we use the mean and standard deviation of the density values of historical clusters to compute a dynamic density threshold. Let C_t^{no} be a *novel cluster* and $F_{C_t^{no}}$ be its density value. The mean and standard deviations of historical clusters are denoted as $\mu(F_{C_t})$ and $\sigma(F_{C_t})$, respectively. The following Remark is defined for novel cluster detection.

Remark 3. *If* $F_{C_t^{no}} \geq |\mu(F_{C_t}) - \sigma(F_{C_t})|$, *then* C_t^{no} *is a true novel cluster.*

Remark 3 is derived from the three-sigma principle of Gaussian distribution used in [143] and it means a cluster is considered as a valid *novel cluster* if its density value falls inside the one-sigma distance from the average density of historical clusters at time t. With Remark 3, a novel cluster detector with a dynamic density threshold is used for validating the existence of *novel clusters*.

3.3.4.5.4 Adaptation of Drifted and Novel Concepts using Active Learning Considering time and space constraints, we maintain a summary of the data stream rather than keeping all historical samples from DS. The maintained summary HS_t consists of two levels of summary on DS: (i). macro-level summary GS_t, and (ii) micro-level summary LS_t. The definitions of these two levels of summary are expressed as:

$$GS_t = \bigcup_{i=1}^{|C_t|} [Z_t^i, F_t^i, R_t^i], \tag{104}$$

and

$$LS_t = \bigcup_{i=1}^{|C_t|} \{ \bigcup_{j=1}^{|SC_t^i|} [SC_t^{i,j}, y_t^{i,j}]\}. \tag{105}$$

Where F_t^i and R_t^i denote the density value and radius of the i^{th} cluster at time t, respectively. $y_t^{i,j}$ refers to the class label of $SC_t^{i,j}$ and all samples from $SC_t^{i,j}$ share the same label. LS_t is used to explore the sub-cluster structure of each cluster when classes are highly overlapped. Specifically, we split each cluster into a set of sub-clusters such that each sub-cluster only has a unique class label. Instead of computing the mean vector for each sub-cluster, we consider only the sample with the highest density value in each sub-cluster as the sub-cluster center and use it for label propagation.

These two levels of summary are continuously updated to adapt to the change of DS using an active learning procedure. Since two types of clusters can be obtained from the clustering analysis, a hybrid active learning strategy of informative-based and representative-based sampling is introduced to reduce the labeling costs. The adaptation procedure of these two levels of summary is provided in Algorithm 6. In Algorithm 6, for *novel clusters*, the representative-based query [145] is performed by sampling from the centers of clusters. On the other hand, we conduct the informative-based query [145] in *updated clusters* through a distance-based strategy. Unlike the entropy-based sampling [146] strategy, samples that are relatively far from the *updated clusters* are selected as informative samples for label querying. Let Q_t be the set of queried samples and Y_{Q_t} be the label set for Q_t. After the active label query, the label propagation procedure begins to predict the label of the remaining samples in CH_t using Y_{Q_t} and HS_t. Finally, the predicted labels are used to update the LS_t with a two-step procedure. First, we update the centers of sub-clusters within *updated clusters* with new samples that have higher density values. Second, we create a set of new sub-clusters for each *novel cluster* to capture the characteristics of novel concepts.

3.3.4.5.5 Classification through Label Propagation To classify an incoming data chunk, we employ an effective label propagation procedure based on HS_t and Q_t. First, a set of prototypes with label information are obtained from HS_t and Q_t. Then, the KNN-based classification procedure is employed to propagate the labels of the prototypes to samples in CH_t. Here, we set the value of k to five and the classification procedure is presented in Algorithm 7.

To evaluate the efficacy of the CDSC-AL framework, nine multi-class benchmark datasets from [147], are used in the experiments for performance evaluation. Table 23 summarizes these datasets in terms of sample size, dimensionality, number of classes, and class overlap. We compared the CDSC-AL framework with supervised approaches. Four supervised methods, including Leverage Bagging (LB) [148], OZA Bag ADWIN (OBA) [149], Adaptive Hoeffding Tree (AHT) [150], and SAMkNN [135], are used for the second comparison study. The results are presented in Table 24. To account for imbalanced class distributions in benchmark datasets, we use the *balanced classification*

Algorithm 6: Adaptation of drifted and novel concepts

Parameters: X_{no}: a set of samples from *novel clusters*; X_{up}: a set of samples from *updated clusters*; Q_I: a set of queried samples using the Informative-based sampling; Q_R: a set of queried samples using Representative-based sampling; Y_{CH_t}: the label set for CH_t.
Function: ClusteringModel(HS_t, C_t, CH_t)
[Q_t, Y_{Q_t}]=ActiveQuery(C_t, CH_t)
Y_{CH_t}=Classify(HS_t, Q_t, Y_{Q_t}, CH_t)
Update the GS_t according to C_t
Update the LS_t using Y_{CH_t}
return HS_t = [GS_t, LS_t]
Function:ActiveQuery(C_t, CH_t)
Extract *novel clusters* and *updated clusters* from C_t
Identify samples that are close to the *novel clusters* as X_{no}
Identify samples that are close to *updated clusters* as X_{up}
Representative-based sampling for X_{no} to obtain Q_R
Informative-based sampling for X_{up} to obtain Q_I
Q_t = ($Q_I \cup Q_R$)
Query labels from human experts to obtain Y_{Q_t}
return [Q_t, Y_{Q_t}]

Algorithm 7: Classification through label propagation

Parameters: Y_{CH_t}: the label set for CH_t; Sub_r: a set of representatives from sub-clusters; P_t: a set of prototypes with labels.
Function:Classify(HS_t, Q_t, Y_{Q_t}, CH_t)
Extract sub-cluster centers and its labels from HS_t as Sub_r
P_t = $Sub_r \cup$ [Q_t, Y_{Q_t}]
Propagate labels from the prototype set to samples in CH_t using KNN rule and obtain the predicted label set Y_{CH_t}
return Y_{CH_t}

accuracy (*BA*) [151] and the macro-average of *F-score* (F_{macro}) [152] as performance evaluation metrics. We recorded these two metrics over the entire data stream classification and reported the average values for performance evaluation.

Table 24 presents the results of CDSC-AL and the four supervised methods. Using only 10% of the labels, Table 24 demonstrates that the CDCS-AL method achieves the best performance on six of the benchmark data streams including Synthetic-1, Synthetic-2, Sea, Gas Sensor Drift, MNIST, and CIFAR-10. For data streams with abrupt concept drifts, CDSC-AL presents slightly better or comparable performance relative to supervised approaches. In summary, the comparison study with supervised methods reveals that CDSDF-AL could always provide statistically better or comparable performance with the supervised methods using a small proportion of labeled data.

In conclusion, we developed a clustering-based data stream classification framework through active learning (CDSC-AL) to handle non-stationary data streams. The comparison studies with the state-of-the-art supervised data stream classification methods justify the efficacy of the proposed method on both synthetic and real data.

Table 23: Dataset descriptions.

Datasets	Sample size	Dimensions	No. of Classes	Overlap
Synthetic-1	18900	2	9	Non-overlapped
Synthetic-2	11400	2	10	Overlapped
Sea	60000	2	3	Non-overlapped
KDD cup 99	494021	34	5	Non-overlapped
Forest covtype	581012	11	7	Overlapped
Gas Sensor Drift	1391	128	6	Overlapped
Shuttle	58000	9	7	Non-overlapped
MNIST	70000	784	10	Non-overlapped
CIFAR-10	60000	3072	10	Non-overlapped

Table 24: Performance comparison with supervised methods. (Relative rank of each algorithm is shown within parentheses.)

Dataset	Metric	LB	OBA	AHT	SAMkNN	CDSC-AL
Synthetic-1	*BA*	0.7910(2)	0.6640(3)	0.6354(4)	0.6247(5)	**0.9459(1)**
	F_{macro}	0.7965(2)	0.6675(3)	0.6513(4)	0.6313(5)	**0.9490(1)**
Synthetic-2	*BA*	0.7124(2)	0.7204(3)	0.6926(4)	0.6784(5)	**0.8459(1)**
	F_{macro}	0.7218(2)	0.7219(3)	0.6977(4)	0.6864(5)	**0.8149(1)**
Sea	*BA*	0.8204(2)	0.7498(3)	0.7493(4)	0.7205(5)	**0.9691(1)**
	F_{macro}	0.8227(2)	0.7501(4)	0.7505(3)	0.7345(5)	**0.9729(1)**
KDD cup 99	*BA*	0.7585(4)	0.7812(3)	**0.8541(1)**	0.7495(5)	0.8364(2)
	F_{macro}	0.7564(4)	0.7798(3)	**0.8012(1)**	0.7682(5)	0.7921(2)
Forest covtype	*BA*	**0.8888(1)**	0.8707(2)	0.8612(3)	0.8545(4)	0.8465(5)
	F_{macro}	**0.8901(1)**	0.8709(2)	0.8688(3)	0.8588(4)	0.8230(5)
Gas Sensor Drift	*BA*	0.7185(2)	0.6345(4)	0.6111(3)	0.6357(5)	**0.8916(1)**
	F_{macro}	0.7199(2)	0.6361(4)	0.6188(3)	0.6412(5)	**0.8995(1)**
Shuttle	*BA*	**0.4789(1)**	0.4477(4)	0.4508(3)	0.4424(5)	0.4744(2)
	F_{macro}	**0.5187(1)**	0.5112(2)	0.4978(3)	0.4894(4)	0.4789(5)
MNIST	*BA*	0.8909(2)	0.8498(4)	0.8393(5)	0.8549(3)	**0.9669(1)**
	F_{macro}	0.8946(2)	0.8501(4)	0.8412(5)	0.8596(3)	**0.9676(1)**
CIFAR-10	*BA*	0.7199(3)	0.6208(5)	0.7366(2)	0.6218(4)	**0.7857(1)**
	F_{macro}	0.7208(2)	0.6325(4)	0.7381(3)	0.6295(5)	**0.7869(1)**
Mean ranks	*BA*	2.00	2.86	3.57	4.57	**1.86**
	F_{macro}	1.75	2.63	3.00	4.00	**2.00**

3.3.5 Project Progress for Task T3-5 (Formulation of FLS to Handle Uncertainty)

3.3.5.1 Period of Performance under

Task T3-5 Start Date: August 2016

Conclusion Date: Dec 2022

Faculty lead: Dr. Abdollah Homaifar

3.3.5.2 General Description of Task T3-5

To evaluate the LSASV's perceptual capability representative sensing tasks will be used such as the detection of enemy targets in order to launch an attack; formation execution; task allocation and detection and localization of landmarks. The task here is to translate experts' heuristic knowledge into fuzzy IF-THEN statements. Earlier work will be extended to automate the process of categorizing the input/output relations to associated decisions. The fuzzy logic approach can provide the

tester with the reasons for a particular cause of action in human understandable terms. The most fundamental aspect of this connection is that the uncertainty involved in any problem-solving situation is as a result of some information deficiency, which may be incomplete, imprecise, fragmentary, not fully reliable, vague, contradictory, or deficient in some other way. The general framework of fuzzy reasoning allows handling much of this uncertainty. The effects of uncertainty in a system can be handled in a better way by using type-2 fuzzy logic because it offers better capabilities to cope with linguistic uncertainties by modeling vagueness and unreliability of information.

Considering the multi-label classification problem, in which objects may belong simultaneously to several classes an observed instance may apply to multiple classes. One such problem is arises multi-sensor data collection and fusion problem that arise in LSASV's perception studies. Given a new instance representing the item of evidence from different sources (experts) will used to make belief probability assignments based on mass function assignments. The resulting multiple masses from neighboring instances or label representations may be combined by the methods studied pre- viously (for instance the conjunctive sum). For making decisions based on these observations, we may find labels with the greatest commonality and compare two different degrees of belief for each label which result in a multi-labels classifier.

Another research aspect we were exploring where uncertainty plays a significant role is in human-machine interaction and subsequent human-machine trust relationships.

3.3.5.3 Objectives for Task T3-5 over the Course of Project

- **Objective Name:** Multi-sensor data fusion using uncertainty measure

  Objective Type: Research

  Objective Description: Developing a bi-criteria evaluation of belief functions in a multi-sensor data fusion using a combination of both distance/similarity and uncertainty measure.

  Impact: Developed a multi-sensor data fusion solution based on the Dempster-Shafer(DS) theory of evidence.

- **Objective Name:** Conflict Resolution in the Dempster Shafer (DS) theory of evidence.

  Objective Type: Research

  Objective Description: Conflict among the different sources of information is one of the major issues threatening a successful implementation of multi-sensor data fusion systems. This occurs when a decision made based on one information source conflicts with another. A consensus based strategy is proposed to properly manage conflict among the different information sources.

  Impact: Identification of spoofed sources from hostile targets. Detection of faulty or malfunctioning sensors.

- **Objective Name:** Target classification.

  Objective Type: Research

  Objective Description: Target classification is the process of categorizing an observed target into one of the known set of target classes. To make a decision about the class of the target, the sensing devices are used to capture the various attributes of the target.

Information extracted from the observation of the different attributes are often characterized by uncertainty. To effectively manage potential conflict, a reliability factor based on the discriminating capability is proposed. This controls the relative contributions of the different information sources on the final decision.

Impact: This can offer a significant application in the identification of friendly/hostile target e.g. identification of friend and foe (IFF).

- **Objective Name:** Reduction of cardinality of Information Sources.

  Objective Type: Research

  Objective Description: The selection/reduction of information sources is not only to measure the significance/relevance of the different information sources with respect to output decision but also to capture the redundancy among the different sources. We proposed the average pairwise discordance index (APDI), a measure of discriminating capability to determine the relative importance of the different information sources.

  Impact: Besides getting rid of non-informative/noisy information sources, an important relevance of reducing the number of information sources is to decrease the processing time at the fusion node for the online target classification system.

- **Objective Name:** Human-autonomoy trust evaluation and explainability

  Objective Type: Research

  Objective Description: The objective of the trust evaluation and explainability study is providing a means to understand what factors affect human-machine trust relationship and to understand how trustworthy decisions are carried out by a machine.

  Impact: We devised various machine learning techniques to evaluate trust based on real- time sensing of trust markers, and the explainability study of trust using the SHAP explanations provided insights into the need to reduce the number of features needed for the classification problem. Additionally, in conjunction with variable importance evaluations, our explainability study indicated the subjectivity of trust propensities that implied the possibility of designing autonomous systems tailored to individual needs. However, it also implies that the generalizability of trust-related features is rather difficult with a small sample size of individuals studied.

3.3.5.4 Progress Against Planned Objectives in Task T3-5

The objective of this task is accomplished. Several novel algorithms addressing the notions of uncertainty modeling and sensor data fusions have been developed. Additionally, we devised a way to evaluate human-trust from real-time sensing and explainability of such evaluation approaches based on SHAP explainability approach. The main accomplishment of this task are:

- In the developed multi-sensor fusion based on uncertainty measures, we devised a DS theory of evidence employed in a multi sensor data fusion for classification of targets. Measurements acquired by different sensors for the various attributes of the observed target are represented by belief functions. The belief functions are then combined using a robust fusion algorithm to obtain a final belief function. The resulting final mass is transformed into probability distribution for decision making as to the class of the observed target.

- With regard to human trust evaluation based on real-time sensing of pyschophysiological signals, we developed novel approaches of feature extraction and classification methods that are able to distinguish trust/distrust behaviors based on features extracted from the pyschophysiological sensors. Additionally and even more importantly we devised a method toward explanability of machine decisions on the basis of Shapley Additive Explanations.

3.3.5.5 Technical Accomplishments in Task T3-5

- Multi-sensor data fusion is usually believed to produce more reasonable results than a single sensor in estimating the state of a problem. As rightly discussed in some of our earlier reports, the DS theory of evidence is an important framework that can be deployed for fusion of information from different sources. An integral component of the DS theory is the DS rule of combination which is the fusion engine. This engine tends to yield unreasonable results when information in form of the belief functions being reported by different sources are conflicting with one another. In the last quarterly report, we introduced a fusion strategy that assigns weights to every belief function before the deployment of the traditional DS rule of combination.The weight (evaluation) of each belief function is based on the distance between each belief function and the average belief function. This evaluation criteria is based solely on distance function. Measure of uncertainty has been used to assign weights to bodies of evidence. Since the weight of a belief function can be determined by using not only the distance function but also uncertainty measure. The use of both measures can be described as a bi-criteria evaluation approach. In this report, we consider a quick literature survey of some uncertainty measures in the theory of belief functions. The numerical example in the last quarterly report is used to demonstrate the usefulness of the different measures. Measuring uncertainty in evidence theory is simply the task of quantifying the information volume of a belief function. They can be broadly categorized into three: Conflict/discord, non-specificity/imprecision, and total uncertainty [153–157].

- Conflict management based on the average consensus belief function We proposed a new fusion strategy based on the average belief function (consensus belief function). In this approach, weights are assigned to each belief function based on its closeness to the average belief function. The closer, the higher the weight of the associated information source. The obtained weight is then used to adjust the effect of the individual information sources on the

  final decision.

- Information extrcated from the sensors are model as belief function to induced local declarations represented by m1; m2; :::; mN. A reliability factor is proposed to control the impact of the various sources on the global declaration denoted by m.

- Selection of information sources based on the average pairwise discordance index (APDI). The rationale behind the selection of information sources is to ensure that the selected information sources are: correlated with target classes, discriminatory with minimum degree of overlap among the different target classes as fewer as possible. With every attribute/feature considered as an information source, we proposed a

  filter-based selection strategy that utilizes the average pairwise discordance index (APDI).

- In light of the second problem at the outset of this work, real-time sensing and explainability of trust, the prediction/classification performance of the machine learning methods demonstrated the ability to use the features acquired by the psychophysiological sensors in a real-time manner. We devised ensemble classification methods and methods based on the reduced feature sets to classify trust in human-machine interaction teams, owing to a couple of reasons. First, the ensemble methods have inbuilt parameter optimization capabilities, which enable successively and iteratively improve their decision capabilities. Secondly, these methods exploit the informativeness among the correlative analysis they carry out among a larger set of features which, despite the redundancy of features, gives them an additional edge. Nevertheless, the traditional machine learning methods implemented based on the reduced sets of features that resulted from the feature selection process have also performed reasonably well on almost one-tenth of the available features to classify trust in human-machine teams under uncertainty representation. Both performances attest to the practicality of using real-time sensing of factors affecting trust [158–160].
- The explainability study of trust using the SHAP explanations provided a couple of insights as well. First, it intimated the need to reduce the number of features needed for the classification problem. Second, in conjunction with the variable importance evaluations, the SHAP summary indicated the subjectivity of trust propensities. This was demonstrated by the fact that individuals' final trust/distrust decisions depend on different features. This also might lead to the conclusion that there is a possibility of designing autonomous systems tailored to individual needs. However, it also implies that the generalizability of trust-related features is rather difficult with such a small sample size of individuals studied [161].

3.3.6 Project Progress for Task T3-6 (Train and Test LSASVs using PIE)

3.3.6.1 Period of Performance under

Task T3-6 Start Date: 15 August 2017

Conclusion Date: 23 December 2020

Faculty lead: Abdollah Homaifar, Ph.D

3.3.6.2 General Description of Task T3-6

Embedded in this task is the task of inferring the LSASV's perception of its environment because this perception determines what the LSASV's active behavior state should be. As the complexity of LSASV increases, efficiency and cost are considered critical factors in T&E; the search for near- exact models of behavior and perception inference becomes challenging. The method for developing these models must use observations of the LSASV and its interactions with its environment and learn from these interactions. LCS that embraces a hierarchically cooperating and competing set of classifiers can meet these needs. The uncertainties associated with the system can also be modeled with a type-2 Fuzzy Logic System. Once sufficiently trained, PIE will be able to correctly predict the outputs of LSASV given input scenarios that were not exposed to the PIE during the training phase of the PIE. The PIE infers the internal processes/transitions of the LSASV's and predicts the behavior of the LSASV next state, such as the ability to determine an enemy target. Because

the rules governing an LSASV's behavior are not fully known, nor are the capabilities of its sensor systems, the behavior and perception models require ongoing adaptation in order to approximate unknown scenarios intuitively. Secondly, the internal parameters governing the behavior of the LSASV might change with time, which may cause the PIE's behavior to deviate from the behavior of the LSASV. The online adaptation of the PIE enables it to adjust its training parameters to align itself with the LSASVs behavior.

3.3.6.2.1 A theoretical study of decision tree algorithm We studied the theory behind the decision tree algorithm. The ID3 algorithm is listed and implemented for a synthetic data set. The implementation results are reported.

3.3.6.2.2 A theoretical study of Probabilistic graphical model(PGM) PGM was studied for the development of PIE. In PGM, the Bayesian network had been investigated. The chain rule of probability, different reasoning techniques such as causal, evidential, and inter-causal reasoning were discussed. The vital concept of D-separation, Active trail, Factorization, and I-map were studied. Two types of structure learning technique were reported which were constraints based approaches and score-based approaches with an easy to follow the example.

3.3.6.2.3 Implementation of the studied algorithms on a synthetic dataset Two well-known algorithms are known as The Naive Bayes Classifier and The Tree-Augmented Naive Bayes Classifier were implemented on a synthetic dataset from UAV simulation. The dataset has 12 input variables and 4 output classes. We got an average accuracy of around 85% in this experiment.

3.3.6.2.4 A simulation environment is developed to collect synthetic data and an initial framework has been proposed We developed a search mission in the simulation environment. The scenario is developed using the Robot Operating System(ROS), Gazebo, and Pixhawk. Two different UAVs are used to design the scenario named as 3DR IRIS and 3DR SOLO. The synthetic data had been labeled to five different modes of operation named Hold, Takeoff, Hover, Search, and Land. Moreover, the definitions of each mode were provided. A decision tree algorithm is applied to these data for the prediction of the defined modes. The empirical results showed that the proposed testing framework can successfully predict the defined modes. At this stage, we achieved our first goal towards the development of a data-driven testing framework. The details of this accomplishment are provided in the following section.

Description of the Perception Inference Engine

The Perception Inference Engine (PIE) is composed of three main sub-systems. They are named as follows:

1. Data collection
2. Prediction model generation
3. Test & Evaluation

Each sub-system has its own components. The complete system is shown in Figure 126.

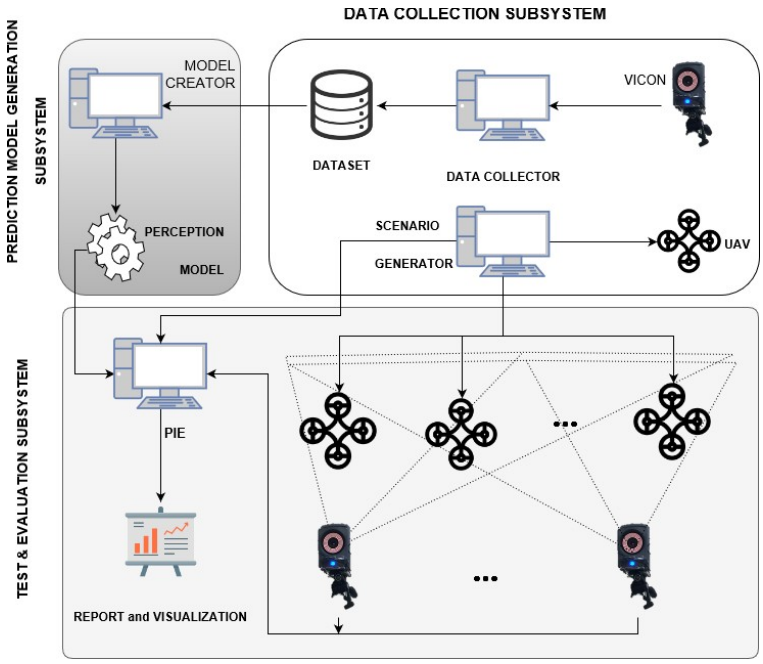


Figure 148: Complete PIE system diagram.

1. **Data collection sub-system**
   Data is collected in three steps. First, a scenario is generated which is composed of different high-level states describing, distinct behavioral modes, characteristics, or actions. Second, the UAV behaviors needed to execute the scenario are implemented. Third, the UAV behavior is captured as observed by an external sensor. An external sensor is an off-board observation of the UAV system which can be a VICON motion capture system in a lab environment or a radar system for outdoor experiments or other visual or sensor networks [162]. External sensor data are stored and used to generate a model capable of predicting future UAV behavior within the scenario. A sample scenario is described in Section 2.3.6.2.4. The high-level states are described in more details in Section 2.3.6.2.4. Measurement of different variables can be captured during the online operation such as position, orientation, linear and angular velocity, linear and angular acceleration, altitude or video of the UAV. Captured data is then forwarded to the prediction model generation sub-system.

2. **Prediction model generation sub-system**
   After getting data from the data collection sub-system, it is labeled by a human expert according to the definition of high-level states introduced later in Section 2.3.6.2.4. This labeling process can be automated by using some probabilistic approach during the data collection process [163, 164]. However, in our experiment, we did it manually. A block diagram of this sub-system is shown in Figure 127. The collected data are labeled first and split among three sets named as validation, training and testing. Then the training set is used for training the classification model and later the validation set is used for fine tuning the trained model and at the end the testing set is used for evaluation of the fine-tuned model. At this point, we have labeled data, which can be used to predict the label from the features that we recorded during

the online operation. In the machine-learning domain, it is known as a classification problem and it is a supervised learning problem because we know the label associated with the input data. In our system, we have multiple high-level states, which makes it a multi-class classification problem. In the machine-learning literature, there are many algorithms and approaches for solving multi-class classification problems such as Decision Tree (DT), Neural Network (NN), Support Vector Machine (SVM), Naïve Bayes (NB), Learning Classifier System (LCS), Fuzzy Logic (FL), and a Bayesian Network (BN) to name a few of them [165]. However, each algorithm or approach has its own strengths and weaknesses [165], for example: a neural network is known to be a "black-box" (in terms of the transparency, rationale or logic of its input-output mapping) and needs many iterations to converge [166]; inferencing and finding the optimal structure in a Bayesian Network is an NP-hard problem [167]; and the Naïve Bayes approach is prone to under-fit the data [167]. In our system design, we used a Decision Tree algorithm. There are different variants of the DT algorithm. In our implementation, we used the CART algorithm. The mathematical formulation of the decision tree is described in [168]. After proper analysis of the DT model, it is used by the test & evaluation sub-system.

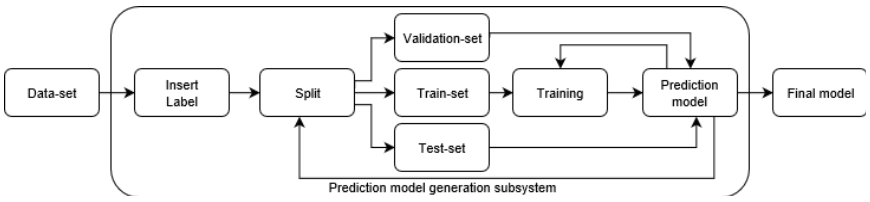


Figure 149: Block diagram of prediction model generation sub-system.

3. **Test & Evaluation sub-system**
   In the test & evaluation (TE) sub-system, the trained DT model is used to predict the current state of the UAV. As with the data collection sub-system, UAVs are updated with different scenarios and the external sensors are activated to observe the UAVs in operation. Data from the external sensor is directly fed to the DT model, which predicts the current states of the UAVs. PIE then matches these states with the expected states according to the scenario generator and produces a human-readable report about the flight. A sample report is shown in Figure 128. The report provides information about the capability of the UAV to carry out a particular mission. So, it is a proof of the performance of the UAV.

State Definition

We defined five high-level states during the simulation and hardware implementation. The high-level states are named as hold, takeoff, hover, search and land. We used four state variables to define the high-level states. The state variables are altitude (h), linear velocity in the X direction ($\dot{x}$), linear velocity in the Y direction ($\dot{y}$) and linear velocity in the Z direction ($\dot{z}$). The coordinate representation, used for state definition, is "ENU." ENU is a ground-fixed frame where X axis points East, Y points North and Z up. The UAV body frame is oriented with X towards the front, Z up and Y towards the left. The definitions of the high-level states are as follows:

Hold: The UAV is in hold state, when $h = 0$, $v_x = 0$, $v_y = 0$, and $v_z = 0$.

```
*********************************REPORT*********************************
The experiment follows the true scenerio  99.8 %.
Details:
The experiment follows  Hold state     100.0 %,
                        Hover state  97.96 %,
                        Land state    100.0 %,
                        Search state   99.83 %,
                        Takeoff state     100.0 %.
Recommendation: The following states need to be checked,  ['Hover', 'Search']
*******************************END REPORT*******************************
```

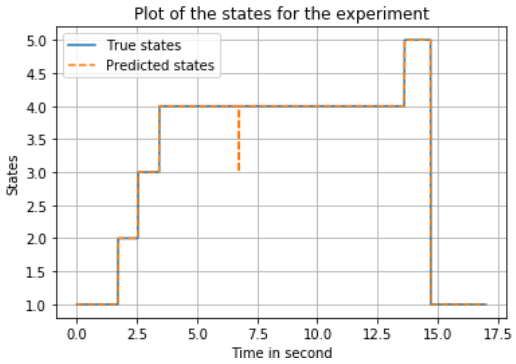


Figure 150: A sample report for a simulation experiment (here, Hold = 1.0, Takeoff = 2.0, Hover

Takeoff: The UAV is in takeoff state, when $h \neq 0$, $v_x = 0$, $v_y = 0$, and $v_z > 0$.
Hover: The UAV is in hover state, when $h = constant$ & $h \neq 0$, $v_x = 0$, $v_y = 0$, and $v_z = 0$.
Search: The UAV is in search state, when $h = constant$ & $h \neq 0$, $v_x \neq 0$, $v_y \neq 0$, and $v_z = 0$.
Land: The UAV is in land state, when $h \neq 0$, $v_x = 0$, $v_y = 0$, and $v_z < 0$.

Scenario Description

To prove our concept, we designed a scenario so that it contains all the predefined high-level states. In our scenario, we chose a rectangular area for the UAV to search with the aim to observe the total ground area beneath. The UAV starts from a predefined starting point known as the home location and then vertically takes off. It hovers for a few seconds at a constant altitude and then starts searching the area by following a lawnmower pattern using a way point navigation algorithm. After finishing the search it returns to the starting point, hovers there for a few seconds and lands at the home location. In this scenario, all of the five high-level states are implemented. If we discretize the scenario, we observe that it is a combination of hold-takeoff-hover-search-hover-land-hold. The whole scenario and state transition diagram are shown graphically in Figure 129.

Simulation Implementation

For simulating the system, we developed the scenario described in Section 2.3.6.2.4 in the Gazebo simulation environment. The specification of the simulation computer is Intel Core-*i*7(8 core) processor, 16 GB of RAM, 128 GB of Hard Disk and Ubuntu 16.04 LTS version as the operating system (OS). We used the 3DR IRIS and 3DR SOLO quadcopter models as simulated UAVs. These quadcopters are shown in Figure 130. We used the Robot Operating System (ROS) [169] to implement the scenario and used open source Pixhawk firmware for the UAV autopilot. We used the OFFBOARD mode to implement the way-point navigation. In OFFBOARD mode, we can send a

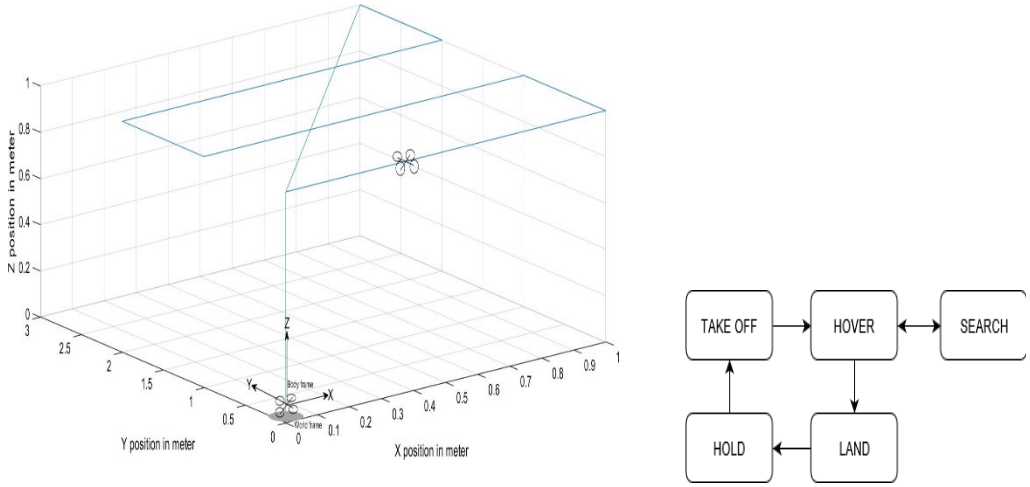


Figure 151: : Visual representation of the scenario and the state transition diagram of the scenario.

setpoint to the Pixhawk autopilot as a goal position.

We logged and saved the sensor data during the simulation in a comma separated value (CSV) formatted file. We saved position x, position y, position z, roll angle ϕ, pitch angle θ, yaw angle ψ, linear velocity in X direction $\dot{x}$, linear velocity in Y direction $y^{\cdot}$, linear velocity in Z direction $z^{\cdot}$, roll speed $\dot{\phi}$, pitch speed $\dot{\theta}$, and yaw speed $\dot{\psi}$. However, we used only position z, linear velocity in X direction $\dot{x}$, linear velocity in Y direction $y^{\cdot}$, and linear velocity in Z direction $z^{\cdot}$ to develop our prediction model.

We used this data to build our classification tree to predict the five predefined high-level states. We ran the simulation ten times to collect $19,884$ data-samples from the IRIS quadcopter and $26,011$ data-samples from the SOLO quadcopter which have five class labels. Then, we labeled the data according to the definition of the high-level states and merged them together. The details of the data samples are given in Table 26. We randomly split the data into two partitions known as a training set and testing set. Then we extract the classification tree from the training set and used that tree to predict for the testing data.



Figure 152: 3DR IRIS (left) and SOLO (right) quadcopter.

We calculate both training accuracy and test accuracy. Figure 131 shows the effect of partitioning the data using a different ratio on the accuracy of the model. For 50% training & validation and 50% testing samples, the prediction accuracy for the training dataset is 99.95% and for testing dataset 98.55%.

Table 25: Physical configuration of the drones used in the experiments.

Name	Motor-to-motor dimension(cm)	Height(cm)	Weight(gm) without battery	Airframe Configuration
3DR IRIS	55	10	1020	Quadrotor Wide
3DR SOLO	26	25	1000	Quadrotor x
Intel Aero Ready to Fly Drone	36	22.2	865	Quadrotor x
Parrot AR.Drone 2.0	45	15	380	Quadrotor x

Table 26: Details of the simulation dataset

No. of samples	Hold	Takeoff	Hover	Search	Land
IRIS	6323	684	6285	5597	995
SOLO	11817	1014	4586	7122	1472

Table 27: Precision, Recall and F_1-Score of the trained model

	Hold	Takeoff	Hover	Search	Land
P	0.99	0.97	0.99	0.98	0.98
R	0.99	0.98	0.99	0.98	0.98
F_1	0.99	0.98	0.99	0.98	0.98

The performance of the trained model is evaluated with precision, recall, F_1-Score and confusion matrix. The mathematical expressions of these performance indices are given below:

$$Precision,\ P = \frac{T_p}{T_p + F_p} \tag{106}$$

$$Recall,\ R = \frac{T_p}{T_p + F_n} \tag{107}$$

$$F_1 = 2\frac{P \times R}{P + R} \tag{108}$$

Here, T_p is the true positive, F_p is false positive and F_n is false negative. The performance of the trained model is given in Table 27 and the confusion matrix is shown in Figure 132.

3.3.6.2.5 A physical experiment has been done in the lab environment After the successful implementation of the framework in the simulation environment, we implemented the whole framework in the lab environment. We used the VICON motion capture system to track each UAV in the real operation. Two types of UAVs are used, one is Parrot AR. Drone 2.0 and the other one is Intel Aero Ready to Fly Drone. The hardware description of each drone was described. We got comparable results in these experiments with the simulation results. Moreover, we provided a detailed discussion regarding the slight discrepancy between simulation and real experimental results. The details of the hardware implementation are described in the following paragraphs.

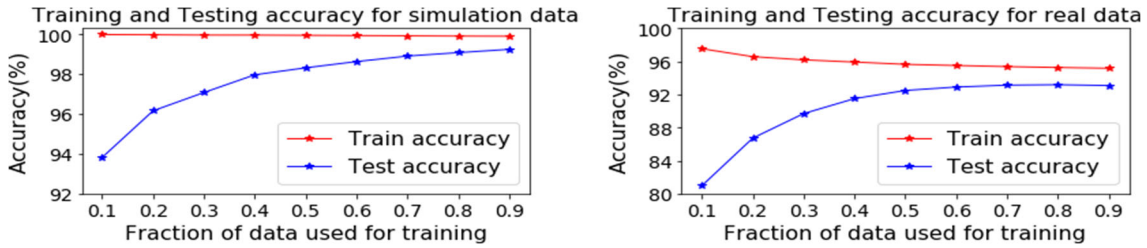


Figure 153: Effect of partitioning data on model accuracy (left for simulation data and right for real data). The accuracy is averaged over 20 trials.

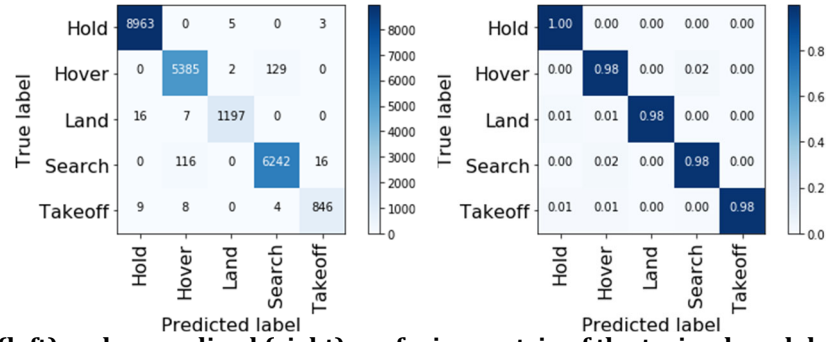


Figure 154: Unnormalized (left) and normalized (right) confusion matrix of the trained model.

Hardware Implementation

For hardware, we implemented the same scenario discussed in Section 2.3.6.2.4 in our laboratory. The dimension of the laboratory-testing arena is 48 15×15 ft^3. For implementing the proposed system in hardware, we used the following hardware and software tools.

Hardware Tools

1. As physical test UAVs, we used an Intel Aero Ready to Fly Drone and a Parrot AR.Drone 2.0. The Intel drone contains an Intel Aero Computer Board which has an Intel Atom x7- Z8750 processor, 2.56 GHz burst, quad-core, 2M cache, 64 bit, 4 GB LPDDR3-1600 RAM, and 32 GB eMMC storage. It can run the Ubuntu Linux OS. The drone was modified for propeller protection and to enable VICON tracking in the laboratory environment. The Parrot drone has an ARM Cortex A8 (1GHz) processor and 128 MB DDR2 RAM. It uses the Linux OS2.6.32 as its operating system.

2. For the external-sensor, we used the VICON motion capture system. Figure 133 shows the modified drones and the VICON motion capture camera setup in our laboratory; there are ten total cameras.

3. To get data from the VICON and to process the data, we used a PC with specification 8 core

Intel Xeon processor 3.7 GHz, 8GB of RAM, 4TB of Hard Disk and Windows 7 Professional as the OS.

4. The Netgear-R6300 WiFi router is used to communicate with the drone.

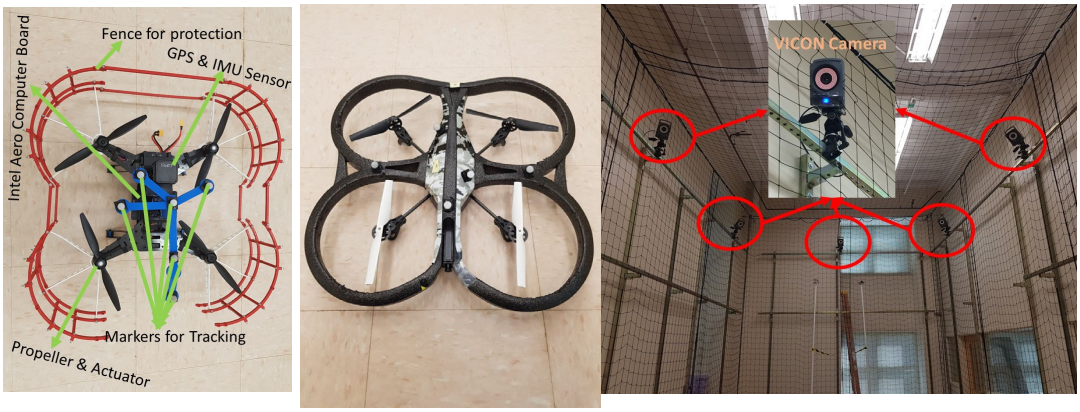


Figure 155: Modified Intel Aero Ready to Fly Drone (top-left), Parrot AR.Drone 2.0 (top-right) and VICON Motion capture system (bottom).

Software Tools

1. PX4 autopilot firmware is used for the drone's low-level control system.
2. ROS is used for developing the scenario.
3. VICON Tracker 3.5 software is used to track the UAV.
4. Python scikit-learn [170] tool is used for developing the prediction model.
5. MATLAB is used for data visualization, labeling, and cleaning.

The hardware implementation is quite similar to the simulation implementation. The main difference between the simulation and the hardware implementation is that in the hardware implementation we used the external sensor VICON to observe the UAV. However, in the simulation, we used the on-board IMU sensor. The steps for hardware implementation are given below:

Step-1: The scenario described in Section 2.3.6.2.4 was implemented using ROS, the OFF-BOARD mode of PX4 autopilot, and MAVROS (an extendable communication node for ROS with proxy for a ground control station) as the communication tools between our ROS node and the autopilot.

Step-2: The VICON motion capture system was set up to track the UAV while in operation and a ROS node implemented to record the four variables named altitude, velocity in the X direction ($\dot{x}$), velocity in the Y direction ($\dot{y}$) and velocity in the Z direction ($\dot{z}$) in CSV format.

Step-3: The saved data was used to visualize and label the data with the five high-level states defined in Section 2.3.6.2.4 using MATLAB.

Step-4: The prediction model was extracted from the labeled data using the scikit-learn tool in Python programming language and the model was evaluated with a different performance matrix and saved for use in PIE.

Step-5: The UAV experiment was run again, this time using PIE, the saved model, to predict the active state of the UAV in online operation. At the end of the experiment, PIE generated a report of the experiment.

We have a total of 13,979 data samples from the Intel drone and 46,891 data samples from the Parrot AR. drone. After labeling, the data are divided into five classes. The details of the data samples are given in Table 28.

Table 28: Details of the real dataset

No. of samples	Hold	Takeoff	Hover	Search	Land
Intel drone	9862	501	1767	1251	598
Parrot AR. drone	4633	4308	21453	14447	2050

We trained our model with real UAV data by splitting the whole dataset into 50% training data and 50% testing data. Data are chosen randomly for training and testing set. Then, we use the trained model for testing the test data. We have 95.54% train accuracy and 92.88% test accuracy. The performance index of the trained model is shown in Table 29 and the confusion matrix is shown in Figure 134.

Table 29: Precision, Recall and F_1-Score of the trained model with real dataset

	Hold	Takeoff	Hover	Search	Land
P	1.0	0.98	0.91	0.89	0.96
R	1.0	0.96	0.93	0.85	0.94
F_1	1.0	0.97	0.91	0.87	0.95

3.3.6.2.6 A custom Quadcopter was built After the lab experiment, we found significant drawbacks of using commercially available drones such as the size, computation capability, or attached sensors on the drone. Therefore, we built our drone which will be used for the physical testing in the lab environment. A detailed description of the hardware components and design procedure of the drone was provided. Moreover, the initial flight testing and its performances were reported. The developed drone was a quadcopter fully compatible with ROS and Pixhawk firmware. This drone gave us the full flexibility to develop different algorithms both in software and hardware. The custom built UAV is shown in Figure 135.

3.3.6.2.7 Lab experiment with the custom quadcopter We developed an identical scenario from the simulation implementation in the lab environment using the developed UAV

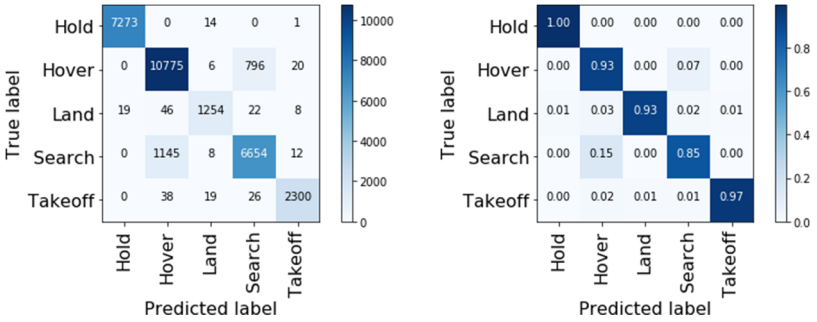


Figure 156: Unnormalized (top) and normalized (bottom) confusion matrix of the trained model for real data.

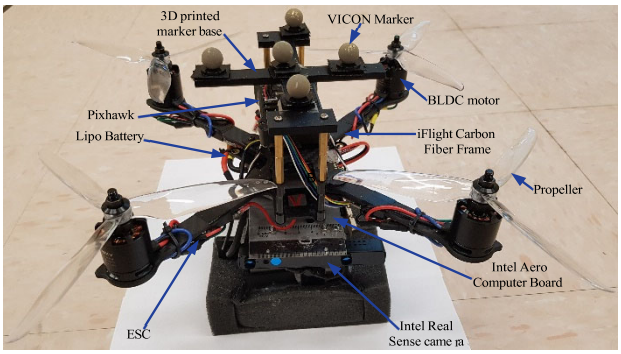


Figure 157: The developed UAV which is used to implement the scenarios and tested in the lab environment.

and collected real-world data from the experiments. The recorded data had been labeled by a human expert and used in three different classification techniques, named as Decision tree, Support vector machine, and Naive Bayes classifier. The comparison study showed that the decision tree outperformed compare to the other two techniques. We also studied the effect of noisy sensor measurement on the trained model and found that the trained model was robust to noise.

3.3.6.2.8 Scalability analysis of the proposed System After validating the system from a real-world dataset, we studied the scalability of the proposed system to the number of UAVs. We found that the memory complexity was O(1) and time complexity is O(N) where N is the number of UAVs for serial implementation of the framework. Conversely, the memory complexity was O(N) and time complexity is O(1) for parallel implementation. We also studied the physical space complexity and proposed two different implementation techniques. First, normal arrangement when the physical space complexity was O(N) and second, alternative arrangement when the complexity was $O(\frac{N}{T+1})$, T is the number of turns in a scenario.

3.3.6.2.9 A mathematical formulation is developed for the proposed system We developed a mathematical formulation of the proposed system and provided a formula to quantify the capability of a UAV in performing its mission in different scenarios. We empirically verified the

formulation with both simulation and real-world datasets.

3.3.6.2.10 A multi-UAV simulation study is conducted We designed a multi-UAV scenario for forest fire detection missions with a different type of UAVs. We proposed a two-step search and survey algorithm to accomplish the mission objectives. The simulation study shows that the proposed algorithm was robust to different environmental parameters such as wind gust or terrain of the environment. We also conducted a comparison study with state-of-the-art techniques and found our algorithm outperformed. The details of the proposed algorithms are described in the following paragraphs.

Problem Description

In this section, the basic definitions and problem description are provided to elucidate the scope of this work.

According to the specifications in the AMASE simulation interface [171] four basic definitions about the characteristics of the benchmark UAVs are described below.

Fire detection sensor: A binary detection sensor is used to generate an interrupt signal when the fire appears in the field of view of the sensor. The footprint of the sensor is a square with a $b \times b$ area (m^2) and its range is R (m) as shown in Figure 136.

Sprint UAVs: A fixed-wing UAV with a maximum speed of V_{sp} (m/s) and a fire detection sensor range of R_{sp} (m).

Survey UAVs: A second type of slower, fixed-wing UAVs with a maximum speed of V_{su} (m/s) and a fire detection sensor range of R_{su} (m).

Fire Zone: A set of $2D$ points (latitude and longitude) in space, consisting of the corner points of an irregular polygon.

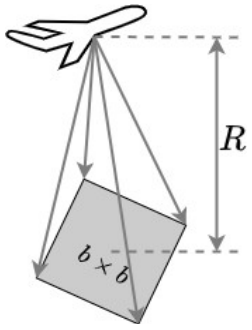


Figure 158: Illustration of the fire detection sensor.

Let A be the area-of-interest in a forest. The map of A, referenced with respect to the Geodetic coordinate system, provides relevant terrain (altitude for a given latitude and longitude) information. The area A has a set of n discrete fire zones $\{fz_1, fz_2, \ldots, fz_n\}$ with respective areas $\{a_1, a_2, \ldots, a_n\}$, where $a_i \subset A, \ \forall\ i \in \{1, 2, \ldots, n\}$; $fz_i \cap fz_j = \emptyset$ for every $i, j \in \{1, 2, \ldots, n\}$ with $i \neq j$; and $\sum_{i=1}^{n} \cup a_i \subset A$. The n fire zones are randomly distributed in the area-of-interest A at unknown locations. It is assumed that there are p *sprint* and q *survey* UAVs denoted as UAV_i, where $i \in \{1, 2, \ldots, p+q\}$ with $p > q$, randomly placed inside the area A. In this study, three principal problems are considered:

1. How the *p sprint UAVs* and *q survey UAVs* will search the area A for those n fire zones within the time interval of duration T?
2. When a fire nest is detected, how the UAVs will estimate the area of the detected fire zone areas?
3. How the *p sprint UAVs* and *q survey UAVs* will coordinate during the search and estimation mission?

While carrying out the search and survey missions, it is assumed that the UAVs (both types) crash to the ground if the following conditions are met:

1. Flying outside of area A for a Δt time interval.
2. Flying on top of any fire zone fz_i for a Δt time interval.
3. Flying without considering the topography of the area A (e.g., flying in low altitude, colliding with mountains or trees).

Proposed Methodology

In this section, a two-step search and survey procedure is proposed for a team of heterogeneous UAVs to perform the search and area estimation for forest fires. For clarity, we deconstruct our proposed multi-UAV system into two principal components: (i) The fire search procedure, and (ii) The fire boundary survey procedure. The details of each procedure are discussed below.

1. **Search for Fire Zones**

   Grid-based search strategies have been widely used in many search and rescue missions for UAVs. In this paper, the proposed method utilizes a grid map for the search of potential fire zones. With p available *sprint* UAVs, the search area A is divided into p equal square grid cells with area $A_c = \frac{A}{p}$. Then, all p *sprint* UAVs are preplanned to conduct the search of the fire zones, and each *sprint* UAV is assigned with a specific grid cell, as shown in Figure 137. At the same time, *q survey* UAVs are send to the center of the grid cells for loitering so that they can be quickly switched to the boundary surveying operation for any neighboring detected fire zones. However, unlike other existing search paths, the proposed approach adopts a novel search pattern for the local search within each grid cell. The shape of the proposed search pattern is similar to the $2D$ projection of a sand clock. It is because of this that is referred to as the *Sandclock* search pattern. As shown in Figure 137, the *Sandclock* search pattern is rotated by $\frac{\pi}{2}$ radian in the neighboring grid cell to avoid repeating the search for fires in the boundary regions, improving the overall search efficiency. The generated trajectory of the *Sandclock* search pattern is a set of way-points (latitudes, longitudes, and altitudes). The altitude of a way-point has been calculated by adding a safe height (100 m in our implementation) according to the terrain information. The calculated safe altitude reduces the chance of potential collision of the UAVs within the terrain. Assuming that all the UAVs have a robust low-level controller capable of following the given way-points reliably.

   Considering real-world forest fires usually spread quickly and have a large impacted area after a short period of time, UAVs have a higher chance to detect the fire zone by following the *Sandclock* search pattern. Using elemental geometry arguments, from Figure 137, it is seen that the *Sandclock* pattern guarantees the detection of an occurring fire zone if $a_{firezone} > \frac{A_c}{}$, 4

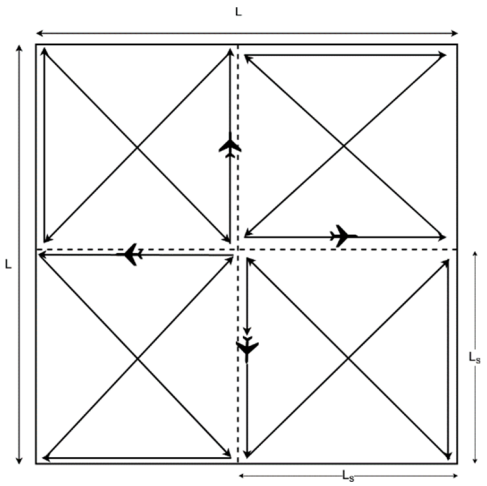
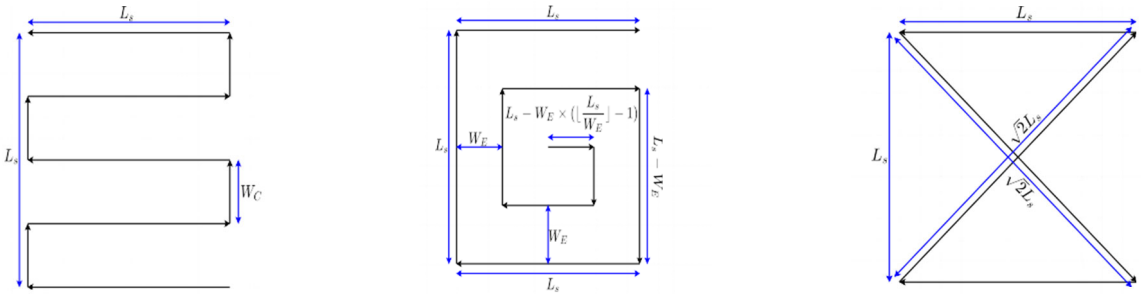


Figure 159: Proposed search pattern. The dotted line shows the boundary of each grid cell.



(a) *Creeping line* (b) *Expanding squares* (c) *Sandclock*

Figure 160: Graphical representation of the Creeping Line (CL-search), Expanding Squares (ES- search) and the proposed Sandclock search patterns within each grid cell.

where $a_{firezone} \in \{a_1, a_2, \ldots, a_n\}$. This condition indicates that the proposed search pattern will go through the fire zone area when the fire zone covers a quarter of the grid cell. Although the initial area of the fire zones may not satisfy this condition, fire zones are constantly expanding (over time) and their area increases accordingly, which will result in rapidly meet this condition.

To show the benefits of the proposed search pattern, we compare it with two standard search patterns, specifically, the *Creeping Line search* and the *Expanding Squares search*. Let L be the length of the entire search area and L_s be the length of each grid cell such that $L = \sqrt{A}$ and $L_s = \sqrt{A_c}$ where we have assumed a square search area. Below, we compare the path lengths of three search patterns.

Creeping Line search (CL-search): As shown in Figure 138a, let W_C be the width between two consecutive line segments; it is easy to show that there are $\lfloor \frac{L_s}{W_C} \rfloor + 1$ horizontal lines and $\lfloor \frac{L_s}{W_C} \rfloor$ vertical segments of length W_C within each grid cell. Thus, the total length L_{T_C} of the

Creeping Line search pattern is calculated as:

$$L_{T_C} = L_s \times (\lfloor \frac{L_s}{W_C} \rfloor + 1) + \lfloor \frac{L_s}{W_C} \rfloor \times W_C = \frac{L_s^2}{W_C} + 2L_s. \quad (109)$$

Expanding Squares search (ES-search): In Figure 138b, let W_E be the fixed width by which each square increases; then, $\lfloor \frac{L_s}{W_E} \rfloor$ squares are generated within a grid cell. The length of the outermost square can be expressed as $3L_s$. The next inner square has a of length $2(L_s - W_E)$ and the length of the innermost square is $2(L_s - W_E \times (\lfloor \frac{L_s}{W_E} \rfloor - 1))$. Therefore, the overall length L_{T_E} of the *Expanding Square* search is written as below:

$$\begin{aligned}
L_{T_E} &= 3L_s + 2 \sum_{i=1}^{\lfloor \frac{L_s}{W_E} \rfloor - 1} (L_s - W_E \times i) \\
&= 3L_s + 2 \sum_{i=1}^{\lfloor \frac{L_s}{W_E} \rfloor - 1} L_s - 2W_E \times \sum_{i=1}^{\lfloor \frac{L_s}{W_E} \rfloor - 1} i \\
&= 3L_s + 2L_s \times (\lfloor \frac{L_s}{W_E} \rfloor - 1) - 2W_E \times \frac{(\lfloor \frac{L_s}{W_E} \rfloor - 1) \times \lfloor \frac{L_s}{W_E} \rfloor}{2} \\
&= 3L_s + 2\frac{L_s^2}{W_E} - 2L_s - \frac{L_s^2}{W_E} + L_s \\
&= \frac{L_s^2}{W_E} + 2L_s.
\end{aligned} \quad (110)$$

Sandclock search: In Figure 138c, the proposed search pattern does not perform any further decomposition inside the grid cell, and it searches the space by following a path similar to the shape of a Sandclock. Thus, the total length L_{T_P} of the proposed search pattern becomes:

$$L_{T_P} = L_s + \sqrt{2}L_s + L_s + \sqrt{2}L_s = (2 + 2\sqrt{2}) \times L_s. \quad (111)$$

From Equation 111, the size of the grid cell L_s can be derived as $L_s = \frac{L_{T_P}}{2+2\sqrt{2}}$. By substituting L_s in equations 109 and 110 we get the following equations:

$$L_{T_C} = \frac{(\quad L_{T_P}^2}{\sqrt{\quad}} + \frac{2L_{T_P}\sqrt{\quad}}{\overline{2 + 2\quad \sqrt{2}}}, \quad (112)$$

$$2 + \quad _{2)} \quad W_C$$

and

$$L_{T_E} = \frac{L_{T_P}^2}{(2 + \sqrt{2})^2 W_E} + \frac{2L_{T_P}\sqrt{\quad}}{2 + 2\sqrt{2}}. \quad (113)$$

From Equation 112, there is a quadratic relationship between L_{T_C} and L_{T_P} when W_C is fixed. With a fixed value of W_E, Equation 113 shows that the overall length of the ES-search increases quadratically as the length of the proposed search pattern increases. From this aspect, the total travel distance of the proposed search pattern is smaller than the CL-search and ES-search patterns. To further compare the overall travel distances among these three search patterns, the following equations are obtained:

$$L_{T_P} > L_{T_C}, \text{ if, } \frac{L_s}{W_C} < 2\sqrt{2}, \tag{114}$$

$$L_{T_P} > L_{T_E}, \text{ if, } \frac{L_s}{W_E} < 2\sqrt{2}. \tag{115}$$

In Equations 114 and 115, the proposed search pattern shows longer travel distance when the ratio between the size of the grid cell and the width between lines or squares falls below $2\sqrt{2}$. This condition means that the number of lines or squares needs to be less than three (3) and a small grid cell is required. Accordingly, the total number of grid cells increases and more UAVs are required to complete the search task, significantly increasing the search cost. Therefore, the proposed search pattern is more efficient than the other two search patterns (CL-search and ES-search).

The search coverage area for three search patterns has been calculated using the specification of fire detection sensors. The entire search area A is partitioned into smaller grid cells of area $b \times b$ (area of the fire detection sensor's footprint). The central node assigns a value of 1 to a cell, if it has been visited by a UAV, or a value of zero, otherwise. If the current position of any UAV is within any of these small cells, then the cell is assigned a value of 1. If n is the number of all captured cells, by the sensor, then at the end of the mission the total coverage area denoted as A_{cov} is $n \; b \; b \times \; \times$

2. **Fire Boundary Survey Strategy**

After locating the fire zone, the next task for the UAVs is to survey the fire zone and estimate its area. Multiple UAVs are used to survey one fire zone. The UAV which found the fire zone will share its location with the rest of the UAVs in the system. A sub-group of $m \; m \quad q$ (closest) UAVs will fly to the location of the source and loiter around until the fire zone is identified due to its increasing size and perceptual shift in location. If this sub-group of UAVs detect the existence of fire around that location, they begin the survey phase of the monitoring process. Once the group of surveying units is informed about the location of a fire zone heading towards it, they may then discover another one. The detected fire boundary points are stored in a central database as a unified list, irrespective of which point belongs to which fire zone. To this end, a clustering algorithm is applied before estimating the area of a fire zone, as discussed in Section 3. The value of m depends on the availability of the UAVs. In the simulation, we called only from the q *survey* UAVs while the *sprint* UAVs will continue to explore the other areas in the search region. To survey the fire boundary, we developed a new strategy as shown in Figure 139. Consequently, there is the risk of losing a UAV when it hovers more than Δt seconds on top of the fire zone. This constraint is considered in the proposed survey strategy. In Figure 139, we denote by $\Delta\Theta$ the change of heading angle when the UAV identifies the existence of a fire boundary point from the fire detection sensor. The

UAV may enter into the fire zone because of its inertial motion. The angle ΔΘ can be tuned to reduce the time of stay on top of a fire zone. Smaller ΔΘ will increase the time of stay and larger ΔΘ reduces the time of stay. Because the shape of the fire zone is an irregular polygon, there is not a closed-form expression of the time of stay as a function of ΔΘ.

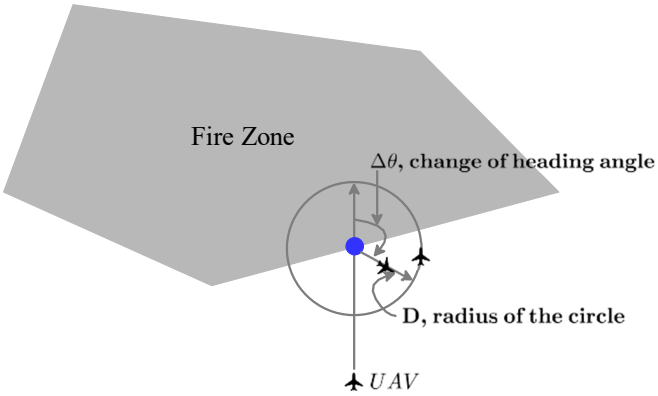


Figure 161: Survey procedure.

The blue dot in Figure 139 is the detected boundary point by the UAV. Since multiple UAVs survey the same fire zone, two consecutive UAVs follow the circle in two different directions. The first UAV surveys the fire zone by following the circular trajectory clockwise, while the second UAV will follow the circular trajectory counterclockwise (if it gets the fire source location from the first UAV) as illustrated in Figure 140. This directional approach restricts two consecutive UAVs to explore same boundary points if they are surveying the same fire zone. In the proposed survey procedure, *D* is considered as a safe distance, and it is included as a design parameter. If *D* is large, then the sampled points from the fire boundary will be far from each other, whereby for a smaller *D* the sampled points will be closer to each other. As such, the safe distance, *D*, will affect the final estimation of the fire zones' shape. As shown in Figure 140, a UAV could successfully survey the whole fire zone by following these cycles. Algorithm 8 summarizes UAV's survey procedure.

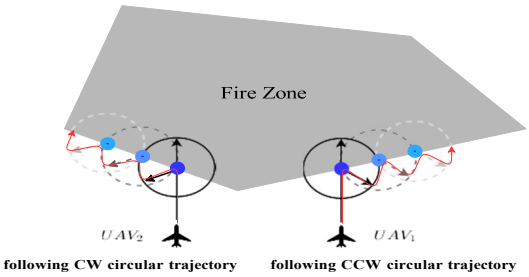


Figure 162: Continuation of the survey procedure for two UAV s. The red lines are the trajectories of the two UAV s during the survey procedure.

Algorithm 8: Survey algorithm

Input: $D, \Delta\theta, direction$

```
γ = reading from the fire detection sensor
while Simulation running do
    Change the heading angle of the UAV by Δθ
    C = Currently detected fire boundary point
    /* This is the position of the UAV while fire detection sensor detects
       fire and γ is True.                                                  */
    if direction == clockwise then
        τ = A CW circular trajectory with center at C and radius D
    else
        τ = A CCW circular trajectory with center at C and radius D
    Send trajectory τ to the UAV
    /* The UAV is assumed to follow the trajectory unless new command is sent
       to it.                                                               */
    while γ==False do
        Wait
/* A CW circular trajectory indicates that the UAV will follow the circle in
   a clockwise direction and for CCW, the UAV will follow the circle in a
   counter clockwise direction as shown in Figure 140. The terrain of area
   A is used to calculate the way-points of all the trajectories as
   described in section 1.                                                  */
```

3. **Estimation of the Fire Zone Area**

In this research, multiple UAVs are used to conduct the search and survey tasks for fire zones and one fire zone can be surveyed by more than one UAV. From the survey procedure, a collection of 2D sampled points are obtained and these sampled points are used to conduct the area estimation for fire zones. In Figure 141, fz_1 is being surveyed by UAV_1 and UAV_3. The blue dots are the sampled boundary points of the fire zone from the corresponding UAVs in Figure 141. We stored all the sampled points in a central database and a grouping algorithm is required for selecting the boundary points of a particular fire zone. For grouping the corresponding fire zone boundary points, we used the Mean-Shift Clustering algorithm. Unlike most of the existing clustering approaches, the mean-shift clustering method introduces the multivariate kernel density estimator to approximate the density distribution of the data and then extracts the cluster structure using the estimated density distribution.

Compared with the state-of-the-art clustering approaches, the Mean-Shift Clustering algorithm is able to group the input data points without knowing the true number of clusters. Since there is no prior knowledge about the number of fire zones, the Mean-Shift Clustering algorithm is a suitable choice for the problem presented. Moreover, the Mean-Shift Cluster- ing showed a strong robustness to the noises, or outliers, which may address potential data corruption in multi-UAVs fire search and survey missions. More importantly, the mean-shift clustering has been recently used for information processing in UAV-based sensing missions and has shown promisingly good performance [172]. The latitude (X) and longitude (Y) of

the sampled points are used as the primary feature representation for the subsequent data clustering analysis. Let $P = \{X, Y\}$ be the coordinates of a sampled point i; the set of all sampled points is denoted as $P_s = \{ P | i = 1, 2, \ldots, |P_s|\}$. The Mean-Shift Clustering algorithm takes the set P_s as the input to perform the cluster analysis. With the clustering procedure, all sampled points are grouped into a set of clusters $C = \{C_i | i = 1, 2, \ldots, |C|\}$, where $C_i = \{P_j^{C_i}, j = 1, 2, \ldots, |C_i|\}$ Each cluster of sample points captures the shape of the fire zones' boundary and the area of the fire zone is estimated by connecting the boundary points sequentially within each cluster. Every two consecutive boundary points from cluster C_i can be connected with different types of curves. In our implementation we used a straight line to connect them. The area of the fire zone is calculated using fundamental image processing techniques. Specifically, a polygon with the sampled boundary points is drawn inside a black background image with height h and width w. Then, the polygon is filled with white pixels. The area is calculated by counting the number of white pixels in the image. Using the same width and height, another image is created for the true polygon. A Bitwise *AND* operation is conducted on the two images to calculate the intersection area of the estimated fire zone and true fire zone. To measure the efficacy of the survey algorithm, we calculate the Intersection Over Union (*IOU*) using the true polygon area and the estimated polygon area. Let A_1 be the true area of a fire zone, A_2 the estimated area, and A_3 the intersection area of A_1 and A_2. Then, the *IOU* can be calculated as $IOU = \frac{|A_1 \cap A_2|}{|A_1 \cup A_2|} = \frac{|A_3|}{|A_1| + |A_2| - |A_3|}$. The *IOU* value is between zero and one, where a value of one indicates perfect estimation of the fire zone area and zero means the estimated area is completely outside of the true area.

The fire zone can expand, shrink, and translate over time in space due to the wind impact. Therefore, we adjust the estimated area of the fire zone based on wind speed and direction in each time stamp. The wind speed and direction can be directly obtained from the on-board sensor of the UAV. All the sampled fire boundary points are updated by wind speed and direction according to

$$\begin{aligned} X_{new} &= X_{old} + v_w \cos\theta_w, \\ Y_{new} &= Y_{old} + v_w \sin\theta_w, \end{aligned} \tag{116}$$

where $\{X, Y\}$ are the temporal coordinate of the fire boundary point, v_w is the wind speed, and θ_w is the wind direction with respect to the Geodetic coordinate system.

3.3.6.2.11 The proposed framework is used with multi-UAV scenarios We defined two new modes of operation and designed five new scenarios for testing multiple-UAVs simultaneously. We collected data from simulation environment to train a LSTM model and verified with the real-world data recorded from the lab environment. We performed a comparison study with decision tree algorithm but LSTM worked better than decision tree. The trained model is deployed to the lab experiment and found satisfactory result from the experiment. Both simulation study and real-world experiment leads us to the same conclusion that the proposed testing framework can successfully evalute the performance of a UAV or a group of UAVs while it/they are executing a real-world mission.

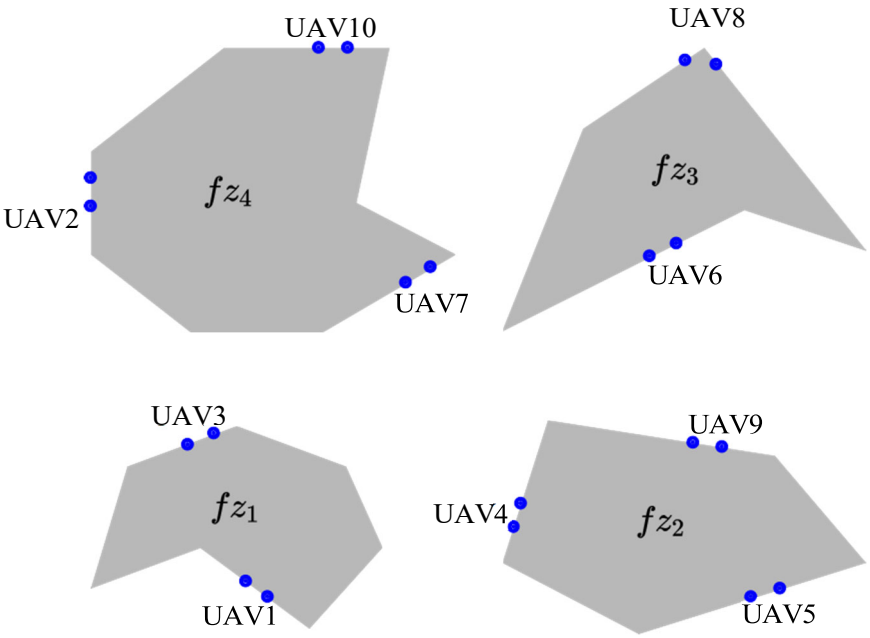


Figure 163: A sample scenario while multiple UAVs are surveying multiple fire zones.

3.3.7 Project Progress for Task T3-7 (Test uncertainty in human perception of system states)

3.3.7.1 Period of Performance under

Task T3-7 Start Date: May 2016

Conclusion Date: December 2020

Faculty lead: Dr. Joseph DW Stephens

3.3.7.2 General Description of Task T3-7

In Task T3-7, we performed research to test and evaluate human perception of system states in operations of autonomous systems of vehicles. This is critical for the success of missions with large-scale autonomous systems of vehicles (LSASV); no matter how finely the system has been engineered, it must have the oversight of humans. This is required for many reasons, including LSASV operations, ethical reasons, in case of a change of plans, system errors, unexpected situations, and situations the LSASV is not equipped to handle. Humans have the flexibility and ability to make decisions that automated systems may not be equipped to handle. Humans are also required to make decisions that involve human life and safety. For example, in a rescue situation being evaluated by an LSASV, the operators must locate people in need of rescue. It is necessary for the human to decide where to send help because errors by the system may result in loss of life. Humans must also be in the loop for everyday operations. Humans need to be there to decide when to take a closer look, where to direct the swarm, what actions need to be taken, how best to complete the

mission, whether the mission needs to be abandoned, when safety concerns arise for people in the environment of the LSASV, and many other situations

To be able to perform this role successfully, the operators need to be aware of what is going on in the system and in the environment. This knowledge is called situation awareness. Maintaining situation awareness while removed from the situation is very difficult. This is further compounded by the fact that most of the sensory cues from the situation are removed or translated into a different sense that would be experienced in person. LSASV interfaces rely heavily on visual cues. Some information may be given by auditory output such as communication between operators and alerts, but the human in the loop does not typically experience the sounds that exist in situ. Similarly, humans do not experience the sense of acceleration and movement (vestibular senses) that they would in person, and peripheral vision is typically limited in visual LSASV displays. In addition to all this, the human in the loop is responsible for paying attention to huge amounts of information for long periods of time. This may include the locations of the LSASV, health of the system, error states, adversary positions, alerts, communications from other people involved in the mission, area map, local map, heading, movement, elevation, terrain, visibility, entities in the environment, physical surroundings (such as buildings, mountains, or lakes), and more.

Because of the incredible complexity and importance of situation awareness in LSASV operations, we have evaluated the ways humans perceive the information being communicated about the system. We have done so by reviewing the literature, performing two types of experiments and designing a simulated user interface.

We reviewed the literature to determine best practices for development of interfaces of LSASV to best support situation awareness and accurate decision making. We found that the best model is to place the human in the loop at a high level of supervisory control, rather than being responsible for many detailed choices at all times.

Based on the levels of automation set forth in the article by [173], the concept of adding a machine estimate places the decision to be made into Level II or III in the Levels of Automation. The computer offers a set (Level II) or a complete set (Level III) of available decisions or actions. By creating an automation estimate, the system can increase the level of the autonomy of the decision making process and offer support to the operator. The types of decisions that are appropriate for machine aid include decisions in the highest level of control. [173] state that this level of control includes decisions that require knowledge, experience, abstract reasoning, and judgment. A user interface that integrates a machine estimate for operations of large-scale autonomous systems of vehicles such as the one developed in our experiments will allow users to make better decisions. It will allow for high level monitoring, increasing operators' situational awareness and decreasing their risk of cognitive overload or cognitive tunneling.

The first experiment evaluated the ways in which humans integrate a decision aid with their own judgment when perceiving novel stimuli. The purpose was to train participants to identify novel stimuli, then test how much they would depend on a decision aid when the stimuli were presented with varying conditions of uncertainty. First, participants were trained on a computer to identify novel stimuli with the nonsense labels "leebish" and "grecious." The stimuli are pictures of shapes from the YUFO stimulus set. The YUFO stimuli had two different groups with several different individuals in each group. Each group had similar characteristics that made it possible to differentiate them. The purpose of this experiment was to understand how humans will react when they need to identify new We chose to use the YUFO dataset as stimuli that the participants have not been exposed to before. Visual representations of motion, heat, chemical sniffer readouts, and

other sensors do not correspond to the way humans experience the world. Operators of unmanned vehicles must be trained to identify and understand the sensor information given to them through the interfaces so that they can maintain situation awareness. To train the participants to recognize the stimuli, we used the following procedure. Participants were instructed to imagine that they were explorers on a new planet. On this planet they would encounter two different types of aliens, leebish and grecious. The leebish aliens were to be approached and the grecious aliens were to be avoided. This was used to create a behavioral response to each type of alien rather than associating any particular key with each type of alien. In each trial, a picture was presented of a single stimulus with a picture of an astronaut next to it. The participants had to decide whether to approach or avoid the stimulus. Participants made their response using the arrow keys. After each response, the name of the stimulus type was displayed above the picture and a bell or buzzer was sounded to indicate a correct or incorrect answer. After the training session, the testing portion of the experiment was administered. Participants were instructed that they needed to identify the aliens once again, this time by pressing the "f" key for leebish or the "j" key for grecious. This time, there were no feedback responses for the responses. The particpants were instructed that a machine estimate was available to help them. It would display the likelihood of the alien belonging to the groups. The estimate was designed to be statistically correct–that is, if the machine estimate said there was a 75% chance the alien was leebish, then the alien was one of a group of four aliens made of three leebish and one grecious alien. This method meant that if one were to round that estimate, it would be wrong one in four times. (Note that the trials were in randomized order so there was no pattern in the stimuli or machine estimate.) The machine estimate gave likelihoods of 75% leebish, 50% leebish, or 25% leebish (and thus 25%, 50%, and 75 % grecious by deduction, respectively). The first and fourth experiments used a numeric machine estimate; it displayed the results as a simple written number and percent. The second and third experiments used a visual machine estimate. We introduced uncertainty in the testing part of the experiment by manipulating the stimuli. The training session used a given set of aliens presented from the front view. The testing session added in a side view in half of the trials. It also added new leebish and grecious aliens that the participants had not seen before. In a given trial, the alien could be from the trained (front) view or the side (untrained) view, and it could be a trained (familiar) alien or an untrained (new) alien.

In the first experiment, the researchers found that as the level of uncertainty increased, the dependence on the machine estimate also increased. This is desirable as human operators will not need the help of a decision aid when they are certain about what is going on in the environment, but they will need assistance when they are not sure. At the lowest level of uncertainty (trained alien and trained view), the participants relied the least on the machine estimate. As the level of uncertainty increased, so did the level of dependence on the machine estimate. For example, with machine estimates of 75%, 50%, and 25% leebish, the percent of aliens correctly identified by all subjects as leebish in the trained alien, trained view category was 93.1%, 75%, and 58.62% respectively. This group of aliens is the most familiar and well-trained category. In the least familiar/most uncertain category, the percentages of aliens correctly identified as leebish at machine estimates of 75%, 50%, and 25% leebish are 78.28%, 50%, and 39.29%. This demonstrates that on average, the participants relied more heavily on the machine estimates as their level of uncertainty increased.

A four-way repeated-measures ANOVA was performed to performed to analyze the results of the testing phase and to determine whether or not any of the independent variables were associated with a change in response patterns. A 2(category of alien) x 2(training level of view) x 2(training level of stimulus) x 3(machine estimate)-way repeated-measures ANOVA found that the machine estimate

had a highly significant effect; $F(2,27) = 16.409$, $p < .001$. The type of alien (leebish or grecious) was also found to be significant ($F(1,28) = 21.533$, $p < .001$), which validates that the type of alien did indeed affect the decisions made by the subjects and that subjects were able to learn to distinguish between the two families. In addition, the interaction between training level of alien and machine estimate was found to be significant ($F(2,27) = 3.681$, $p = .039$) while the interaction between the level of training of the view and the machine estimate did not reach statistical significance ($F(2,27) = 2.481$, $p = .102$). This supports our statement that as the level of uncertainty increased, subjects relied more heavily on the machine estimate.

After reviewing the experiment, we discovered a pattern that suggested the participants may have been rounding the machine estimate to make their decisions. For example, 75% leebish may have been rounded up to 100% to make a classification decision of leebish, and 25% leebish may have been rounded down to 0% leebish to choose "grecious." Therefore, we decided to test the experiment with a visual machine estimate to more clearly convey the information being given by the machine estimate. We also changed the visual stimulus sets to two different groups of stimuli from the YUFO dataset. This was done to test whether the particpants' responses were influenced by distinguishing visual characteristics of the stimuli.

In the second experiment, the subjects relied on average much less heavily on the machine estimate than in the previous study, and they used it the most to identify stimuli in the untrained viewpoint classification category. The first study which used the numeric machine estimate had an average correctness score of 63.427%, but the most recent study which used the visual machine estimate had an average correctness score of 78.971%. It was unclear whether this was due to the use of a different stimulus set, or if it was due to the use of the sliding bar machine estimate. Further study would be required to determine the cause.

The increased accuracy of the scores may be due to the visual machine estimate's representation of ambiguity. The visual representation may provide a better indication of ambiguity that does not facilitate the use of a rounding rule as readily. That may encourage subjects to observe the stimulus more carefully before making a decision, understanding that the visual estimate at the condtion of 50% provides essentially no information.

A four-way repeated-measures ANOVA (2(category of alien) x 2(training level of view) x 2(training level of stimulus) x 3(machine estimate)) showed that once again, the machine estimate had a highly significant effect at $F(2,30) = 19.567$, $p < .001$. This is in keeping with the effect of the machine estimate in the previous experiment. Family x trained and untrained view was also highly significant at $F(1,31) = 15.463$, $p < .001$. The type of alien was once again found to be significant at $F(1,31) = 49.134$, $p < .001$, which validates that the subjects were in fact able to learn the features of the aliens and categorize them by type. In addition, the training level of the viewpoint (trained or untrained) x the machine estimate had a strong effect at $F(2,30) = 5.199$, $p = .008$. Family x machine estimate also had a significant effect at $F(2,30) = 5.134$, $p = .009$.

Overall, the results show that in this second study, subjects were able to integrate the machine estimate into their decision making process in the classification of the stimuli. The accuracy of subjects' responses in this study was higher, which may be due to the difference in the type of the machine estimate. We elected to continue the study using the original stimulus set and visual representation of the machine estimate to compare the results and glean further information about whether the sliding bar representation yields better results than the percentage estimate.

We published the results of the first two experimental studies in a paper entitled "Decision Making Using Automated Estimates in the Classification of Novel Stimuli." This research was

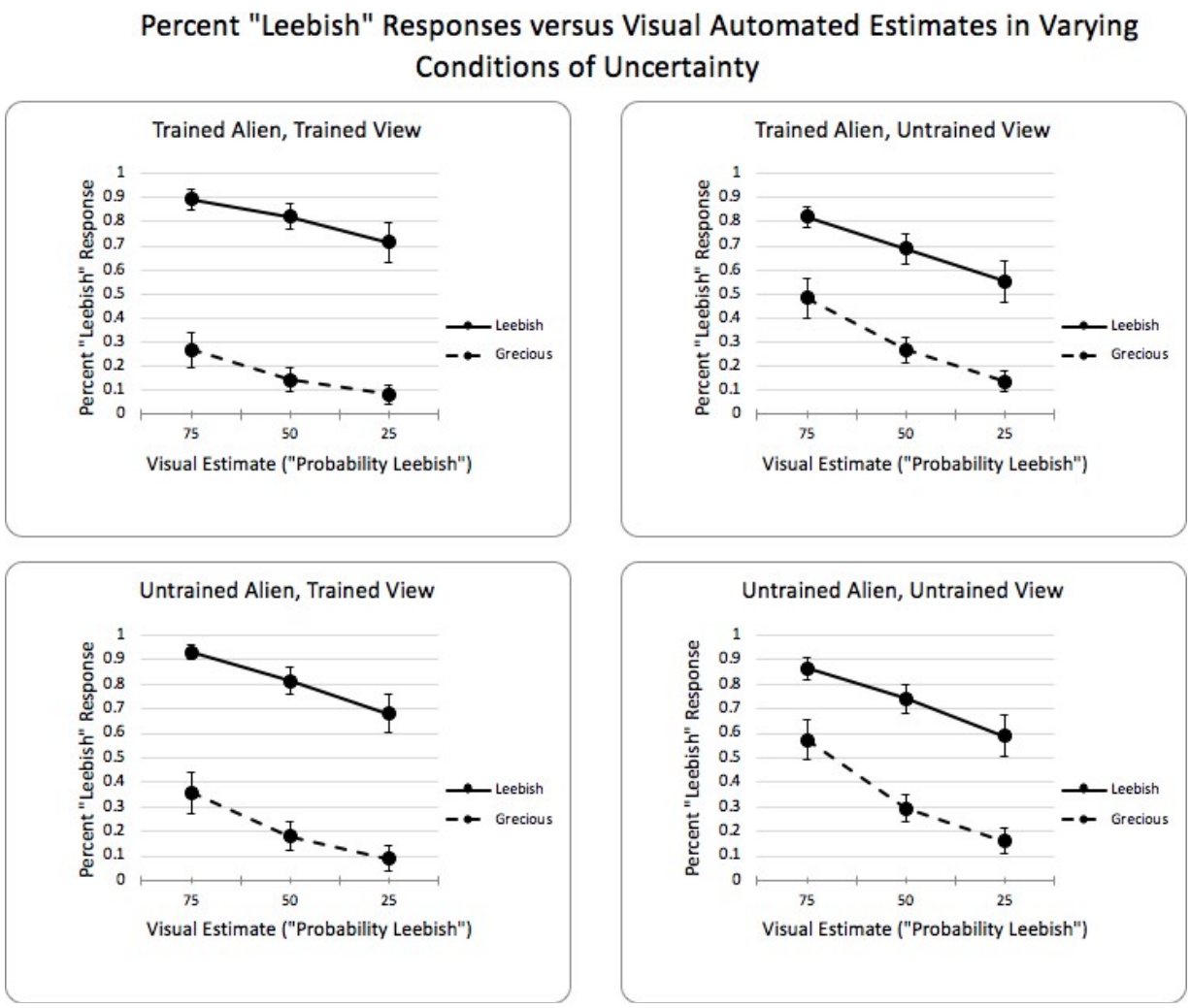


Figure 164: ent of stimuli identified as "leebish" versus the percent likelihood indicated by the machine estimate in the second experiment. The four different stimulus conditions represent varying levels of uncertainty. In the most uncertain condition of untrained stimulus/untrained view, the responses are closest to the percentage value given by the automated estimate.

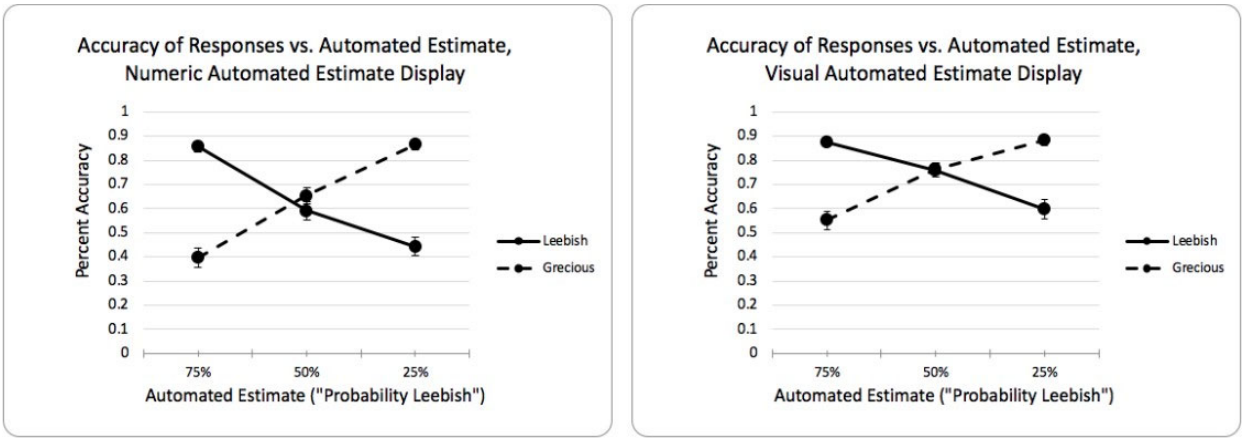


Figure 165: The mean accuracy of responses for each type of machine estimate in experiments 1 and 2.

presented at the 10th International Conference on Applied Human Factors and Ergonomics held in Washington, D.C.

After this, we performed a literature review to ascertain the best direction to go with our continuing research. We assessed the existing autonomous vehicle and LSASV interfaces, decision aids, flaws in existing interface designs, and the effects and visualization of uncertainty. We found that current models of remote operations provide far fewer sensory cues to operators than would be given in the location being represented. Remote operators typically do not receive kinesthetic, auditory, vestibular, and ambient visual cues, and most cues are instead given in a visual manner [174]. Uncertainty is increased by the fallible nature of unmanned vehicles; communication loss, sensor failure, and vehicle malfunctions can all cause problems in the data received by the remote operation interface [175]. This article recommends explicit consideration of uncertainty in decision making rather than simple reliance on designing machines with greater robustness to failures. This information shows the need for user interface design which provides operators with the necessary information while communicating the possibility of inaccuracy.

A study by [176] found that the optimal method of keeping the human in the loop (aware of what the unmanned vehicle system is doing and what its state is) is to use the management by consent technique. This research highlights the importance of human operators actively participating in decision making with automation. In addition, it supports the concept that the present experimental model correlates with the appropriate level of automation. Our research presents participants with information and options from which to choose, matching the management by consent model.

Effective communication of information about system states, situation, and points of interest are crucial in operations of swarms of unmanned vehicles. Deploying many types of sensors in large numbers allows for far greater information capture than a human could manage [177]. Distilling this information into a machine estimate and presenting this side by side with sensor output may be used in large-scale autonomous systems of vehicles (LSASV) operator interfaces to enable the human in the loop to make accurate, quick decisions while harnessing the computational power of distributed and varying types of sensors. The present research combines training and system design with the goal of improving operators' ability to maintain focus as recommended in [178]. The goals

of this type of training and interface also include decreasing mental fatigue, lowering response times, and prevention of loss of systems. Such a system will also take better care of its users; constant vigilance, particularly when using automated subsystems, is stressful and difficult mental work that causes increasing distress with greater task difficulty [179]. Research that promotes such changes is crucial for LSASV use.

Building on the model of the previous experiments, a new combination of the stimulus sets and machine estimate was used in a third experiment, including the second group of novel stimuli from the YUFO set with the percentage machine estimate. This research was necessary in order to determine that the effects observed in the first studies were not due to distinguishing visual characteristics of the stimuli. The results of this experiment are consistent with the previous research and validate the findings that presenting a visual stimulus with a machine estimate is a valid method of presenting information about a situation to humans tasked with classification of stimuli [180]. The results were similar to the previous findings, showing a pattern of increased reliance on the machine estimate in greater conditions of uncertainty. This is desirable in a supervisory classification task.

A multi-way repeated-measures ANOVA was performed on logit transformed data. The machine estimate, classification × training level of stimulus, and classification× training level of viewpoint all showed significant effects on the percent "leebish" classification response of the participants. The effect of the machine estimate is $F(2,16) = 5.652$, $p = .014$. The effect of the classification× training level of stimulus is $F(1,17) = 5.553$, $p = .031$, and the effect of classification × training level of viewpoint is $F(1,17) = 6.322$, $p = .022$.

This experiment showed less significant effects than the first two. This may be due to a smaller sample size ($n = 17$). However, the response patterns were observed in overlapping areas with the first two experiments, most importantly from the machine estimate, which verifies the influence of the decision aid. Another difference is in the accuracy of the participants' responses. The experiment which used a visual machine estimate remains the highest in terms of accuracy; in fact, Experiment Three which uses the same data set as Experiment Two with a numeric machine estimate scored dramatically lower in accuracy by 18.12 percent. These results encourage inclusion of a visual machine estimate in following experimental designs. The findings affirm the validity of the proposed model.

Experiment	Type of ME	Stimulus Set	Mean Response Accuracy	Standard Error
1	Numeric	1	63.40%	.0149
2	Visual	2	73.75%	.0129
3	Numeric	2	55.63%	.0182

After this, we researched the best way to modify the experiment to learn more about situation awareness and decision making in conditions of uncertainty. We evaluated existing user interfaces, the types of information they convey, and their designs to shape our continuing research. We identified a new method to test whether participants will rely increasingly on the decision aid given in previous experiments as the level of uncertainty of the target stimulus increases. We considered several different modalities for the research and concluded that adding a realistic interface may extend the experimental results by simulating the visual experience and heavy task load of operators of unmanned vehicles.

To develop the user interface, the authors explored existing interfaces and related challenges. Operators of swarms of unmanned vehicles must perform a large number of tasks for extended periods of time. They must give commands, monitor system health, diagnose problems, make

decisions about whether to continue or stop the mission, choose whether to make changes to the mission as planned, and maintain awareness of enemies or threats [181]. Current interfaces are difficult to use because of the large amount of information displayed on them. Each has a learning curve, and interfaces are usually designed to be mission-specific [182]. The sensor information and environment can vary with the types of vehicles in use, whether they are unmanned aerial, ground, surface, or underwater vehicles [183]. The information displayed on unmanned vehicle interfaces can be divided into categories of environment, task and control, and equipment. In addition, interfaces may be designed for different levels of automation and thus be configured for different tasks.

To continue the goal of researching ways to help remote operators of unmanned vehicles maintain situation awareness and make appropriate decisions, the researchers are developing their own model of an unmanned vehicle interface. They will combine this with their previous research to test its performance in a more realistic environment. Adding new monitoring tasks and providing much more information will require more cognitive resources and attention switching, thus mimicking the circumstances experienced in real remote operations of unmanned vehicles.

In the experiment, the participants will learn to classify unknown stimuli on a training screen, then be tested on their ability to identify the stimuli using the full simulated interface. In the testing section of this research, participants will be informed that they are monitoring a remotely operated autonomous vehicle on the planet. They will be tasked with classifying the aliens once again, this time with the help of a machine estimate designed to help them. They must also monitor the screen to detect warnings or faults. The participants will be given extra training to enable them to complete their tasks. The levels of uncertainty of the stimuli will be manipulated as in the previous experiments, showing trained and untrained stimuli and viewpoints, to evaluate whether the participants rely increasingly on the machine estimate in more uncertain conditions.

The unmanned vehicle remote operations interface will provide only visual information in order to avoid introducing too many variables at once (e.g. auditory or haptic outputs); a visual interface will already add an abundance of information to the experiment. The interface is divided into eight sections, each of which contains different mission information. The environment sections include a satellite view, large area view, a small area map with points of interest and a key, and the stimulus classification panel used in past experiments. The latter panel is placed in the center of the screen; it displays the stimulus (unknown entity in the environment), a machine estimate of classification likelihood, and reminders for the categorizations associated with each response key.

The top left panel displays mission information, warnings, and communications. The panel on the left center displays measures of unmanned vehicle health. The bottom left and bottom center panels show status information including bearing, altitude, airspeed, wind, command mode, landing gear status, and a picture of the vehicle. The bottom right and center panels show environmental information including temperature, visibility, fuel, cloud ceiling, altitude, elevation, and barometric pressure.

After the experiment was developed, social distancing measures were enacted by the university. For this reason, we redeveloped the experiment for online administration. We made extensive changes to the format of the experiment, set up a server, set up a JATOS account for experimental administration and data collection, and ran many tests.

After administering the online experiment, we analyzed the data and discovered significant changes from the results of our previous work. Rather than showing a pattern of increased reliance on the machine estimate as uncertainty increased, all conditions of uncertainty showed similar strong reliance on the machine estimate. This suggests that increased tasks and cognitive workload

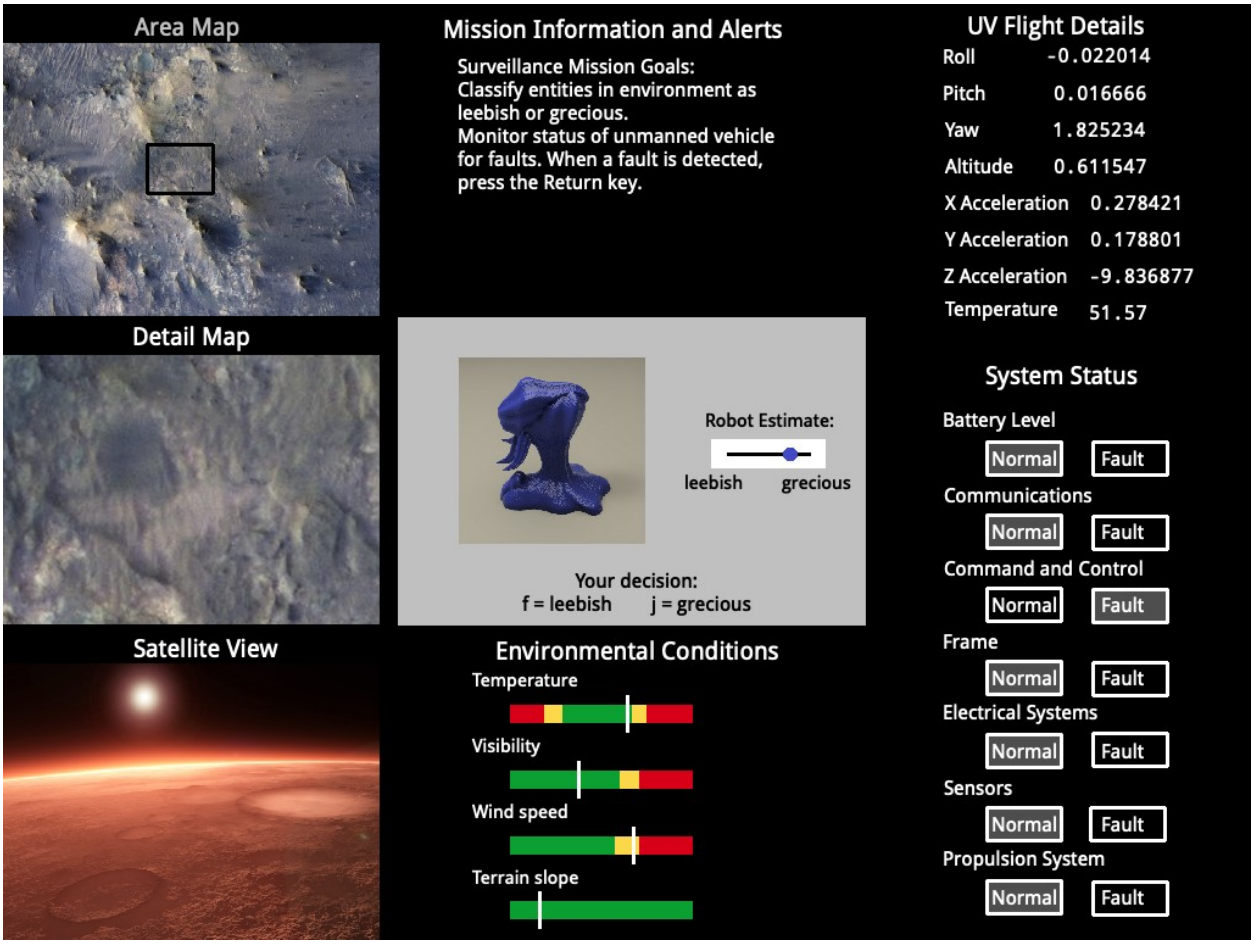


Figure 166: The display interface.

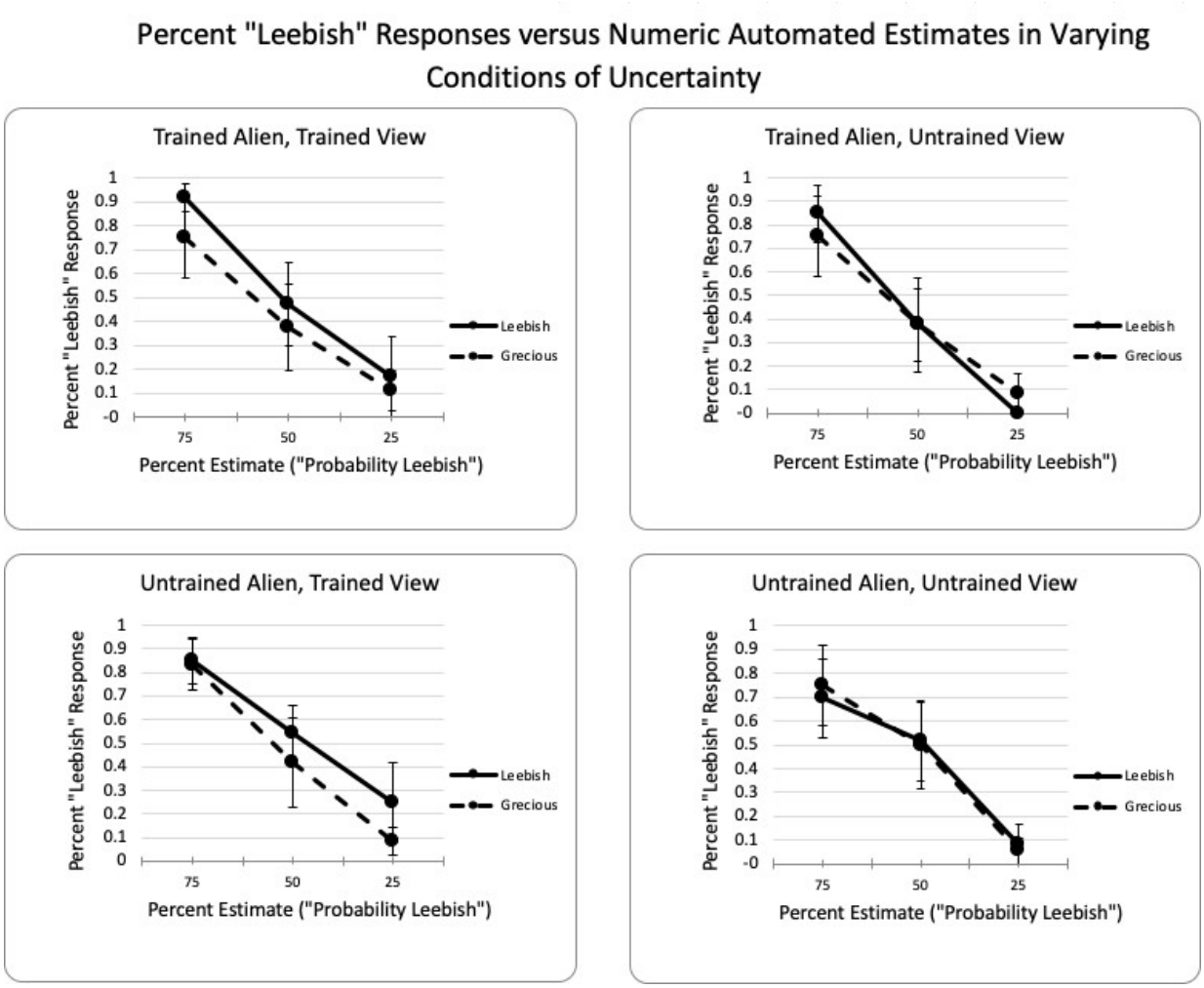


Figure 167: Reliance on the machine estimate in the present experiment. Participants' decisions closely reflect the machine estimates of 25%, 50%, and 75%.

may lead to overreliance on the machine estimate, which could lead to serious errors in real-world missions.

Another significant and surprising finding in this research is that all participants correctly identified all the fault states. Participants may have paid such close attention to the fault states of the system because they can be critical to the survival of the unmanned vehicle. Classification of items in the environment, on the other hand, may not seem as important offhand. This may change when the stakes are higher (for example, target identification of enemy unmanned vehicles in a military task). The participants spent significantly more time identifying the fault states based on the aver- age time spent on identifying them, 5.829 seconds; the mean time spent on identifying the type of visual stimuli displayed was only 1.5216 seconds. This is slightly lower than the response times in the previous experiments, which were 1.887 seconds, 1.756 seconds, and 1.874 seconds in the first, second, and third experiments respectively. On the other hand, time spent identifying fault states

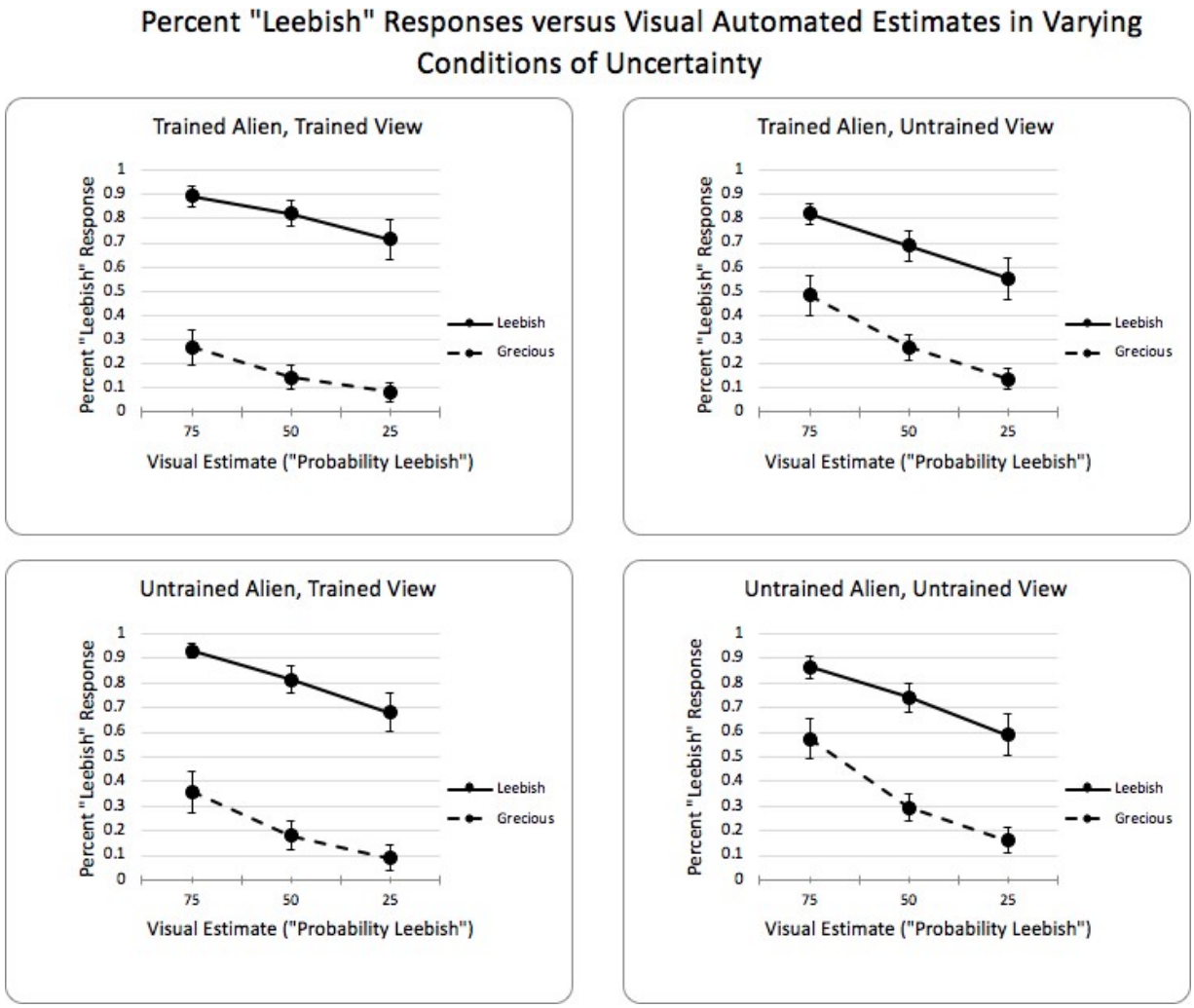


Figure 168: Reliance on the machine estimate in the second experiment. Participants' decisions show increased reliance on the machine estimates as the level of uncertainty of the stimuli increases.

was much higher. We hypothesize that part of the decreased time spent in identification of the stimuli is due to increased reliance on the machine estimate. This might indicate that participants spent less time observing and forming their opinions about the classification of the stimulus, relying on the machine estimate to make a decision more quickly.

Because of this, a weighting scheme may be useful to achieve the desired results. Moving the value of the machine estimate closer to 50% may help to indicate less certainty on the part of the estimate. This could encourage participants to rely less closely on the machine estimate. Specific instructions may be given that although fault state identification is key, target identification is of greatest importance. Another factor that may have contributed to the extra time spent on fault detection is the fact that it was clearly denoted. As long as the participant noticed the fault state indication on the screen, they knew the correct response, unlike the more ambiguous target classification task.

Our research suggests that as the number of tasks to do increases, so does the reliance on the machine estimate, with little change being evidenced by varying levels of uncertainty of the stimuli. This is not the desired effect of the machine estimate; it is intended to be a guide only when needed, when the operator is not sure of the classification of an entity. The increased visual complexity of the interface and the additional system status monitoring task may have increased the cognitive workload of the participants, perhaps contributing to the overreliance on the machine estimate. This is supported by the work of [184], which states that command and control situations, particularly those which require multitasking, can cause attention overload. One method which can be used to encourage appropriate reliance on identification decision aid systems is to inform users of the estimates' reliability [185]. A study by [186] asked pilots in a simulation to determine the miss distance between two approaching aircraft. They found that half of the pilots in the group with automation were able to rely on that automation to assist in making their estimates of the miss distance, improving their accuracy. This may be due in part to the higher reliability of the automation.

Another possible cause of the strong reliance on the machine estimate is that participants may have prioritized fault detection over target identification. Regardless, this overreliance can lead to incorrect decision making with negative consequences. However, this does not mean that a decision aid is not useful in the context of human-machine interfaces for unmanned vehicles. It simply indicates that this behavior may occur in such a context, so designers must plan with this in mind. A weighting scheme that indicates more uncertainty on the part of the machine estimate may help to compensate for participants' overreliance on the machine estimate. In the literature, factors that influence whether or not humans choose to use automation (such as the machine estimate in this experiment) include risk, fatigue, automation reliability, task complexity, trust in the automation, and learning about automation states [187]. Specific instructions or training on how to use the machine estimate accurately with feedback may be useful to assist participants in its appropriate use by helping them understand the automation. [187] also recommends training individual operators to become aware of their own biases toward automation which influence the ways in which they use it.

3.3.7.3 Progress Against Planned Objectives in Task T3-7

Each of the above objectives was completed. In this research area, we have successfully evaluated the ways in which human operators integrate information in uncertainty with a machine estimate

to make their decisions. This is critical knowledge to allow humans to make the best decisions possible in timely, urgent circumstances. Understanding the information being communicated on the interface, being able to successfully interpret it, and making quick accurate decisions is the foundation for successful interaction with the interface, thus promoting success of the mission.

3.3.7.4 Technical Accomplishments in Task T3-7

Two major technical accomplishments were achieved in this task. First, the researchers created an experimental model to evaluate human decision making with a decision aid in varying degrees of uncertainty. Second, we developed a simulated user interface. We tested it with human subjects to learn how the participants responded to different stimuli and extra tasks in a simulated autonomous vehicle monitoring experience. This research will inform development of LSASV interfaces, improving their ability to communicate information about the environment and system, using the proper level of automation, helping the operator maintain situation awareness, and assisting the human with making the best decisions.

3.3.8 Project Progress for Task T3-8 (Evaluate human trust and its calibration in human-machine interaction)

3.3.8.1 Period of Performance under

Task T3-8 Start Date: April 2014

Conclusion Date: April 2021

Faculty lead: Younho Seong. Industrial & Systems Engineering

3.3.8.2 General Description of Task T3-8

Studies in human trust have demonstrated a compelling relationship between decision-makers and the use of automated devices in making crucial judgments. The main objective of this task is to conduct simulated experiments and to develop empirical methods for calibrating trust metrics among human-automation interactions. The subordinate objectives include: a) Defining the necessary factors for calibrating trust in a two-dimensional domain of continuum and human perception when human and autonomous machine interact to perform a task, b) Developing a simulated environment to collect trust-related data in human-automation interaction systems, and c) Calibrating human trust in machine based on experimental data.

3.3.8.3 Objectives for Task T3-8 over the Course of Project

In contemporary society, workplaces have become more complex and dynamic because of the strong need to perform sophisticated tasks and use advanced technology. In these situations, trust can be an important factor of "decision aid" in dynamic and uncertain situations when complex systems are impossible to comprehend and when flexible behaviors are necessary to combat unexpected situations that arise, which are unavoidable. Measuring the human trust of a worker interacting with systems is vital in predicting his or her decision-making. This research examined human trust using an electroencephalogram (EEG) by recording, identifying and analyzing specific brainwaves. This research adopted a word elicitation study from previous research to evoke situations of trust

and mistrust and used a power spectrum analysis for the recorded brainwaves. The results identified that while alpha and beta waves were stronger in the trust situations, gamma waves were stronger in the mistrust situations. Therefore, based on the activities of the brain waves associated with trust, we concluded that trust can impact effective decision-making by increasing concentration and performance, which are related to alpha and beta waves. Contrarily, mistrust can interrupt effective decision-making by increasing stress and anxiety, which are related to beta waves.

3.3.8.4 Progress Against Planned Objectives in Task T3-8

3.3.8.5 Study 1 - Literature Review on Human Trust in Sociological Environment and its Potential Implication for Human Interaction with Autonomous Systems

As there have been some attempts investigating the role of human trust in human machine (specifically, human automation) interaction, reference materials have been collected accordingly. In addition, the issue of human trust in a sociological environment has been surveyed and the reference materials have been collected and summarized. The progress has been made according to the initial plan. Articles and manuscripts from various journals and sources were collected and summarized to provide a foundation to develop 1) theoretical framework to develop a model of human cognition with and trust in autonomous systems, and 2) an experimental testbed to investigate some identified factors of human cognition and trust on human autonomous system collaboration. In addition, we discovered some potential evidences in finding the correspondence between human trust and brain originated waves, leading us to envision an establishment of experimental protocols to be used later.

During the past decade, the role of human trust as a determinant of human interaction behavior with automated systems has been studied extensively. Borrowing several definitions from the sociological and more dyadic perspective, landmarks studies performed by several researchers found empirically that the levels of trust affected by the reliability of automated systems were a good predictor of the human use of such systems, and once this trust was broken, it was difficult to recover. These studies used many task domains such as an industrial process control, autopilots and flight management systems, and maritime navigation systems to investigate the role of trust in joint human-machine work systems.

Studies in trust have demonstrated a compelling relationship between human decision makers and the use of automated devices in making crucial judgment. In a study, trust among civil aviation pilots was strongly correlated with control actions based on the information displayed by automation. It was observed that trust is a true component of pilot activity. A study presented equally useful results in studies of air traffic controllers as there was a trust between the air traffic controllers and their reliance on the conflict probe automation designed for the aircraft separation. Hall [188] noted that trust in automation is similar to trust in human organizations involving supervisors and subordinates. Hall [188] therefore concluded that trust can be used to estimate how supervisors trust their subordinates regarding the use of key critical information in decision-making. Trust has been shown to serve as a metric that determines whether a human will use automation judiciously, misuse it, or even abuse it.

A close examination of trust characteristics in the literature appears to suggest that some characteristics can only be captured through analytical interaction, (e.g., effects of automation reliability on human trust of the automation). However, psycho-social factors such as fiduciary responsibility, usefulness, and familiarity are based on human judgment and intuition and must be subjectively

evaluated. Under these conditions, the theory of cognitive continuum is useful in evaluating human trust. In this theory, human cognition dynamically varies on a continuum ranging from analytic to intuitive, depending on the properties of task (e.g., graphics versus text). Thus, trust may be defined using the same continuum which is termed the trust continuum. In this context, we may be able to define human trust continuum, borrowing the fundamentals of cognitive continuum.

Another important issue with regards to human trust in autonomous systems is the concept of trust along with its calibration. Studies on human trust in automated systems have been dramatically increasing during the past decade. Since several studies provided an extensive review on trust in sociological perspective ([189]; [190]), this work will focus on the issue that has not been discussed elsewhere. First is the question of how humans build their trust in automated systems. In other words, what dimensions of trust in automated systems will humans examine to decide whether to trust or not to trust. [191] provided a list of dimensions or attributes of trust specifically for a human machine environment. These are reliability, robustness, validity, transparency, understandability, usefulness, and utility. While one dimension, reliability, has been extensively used by many landmark studies ([189]; [190]) to operationally and successfully manipulate human operators' trust in automated systems, the rest of Sheridan's dimensions were never discussed or used in an attempt to calibrate operator's trust. This is precisely the value of this study by visualizing other dimensions to provide an opportunity for an operator to calibrate trust. Therefore, brief examinations of these unused dimensions are necessary.

First is the reliability factor. Briefly, this refers to a system of reliable, predictable, and consistent functioning ([191]). Most of the prior definitions of trust addressed this attribute as the first step in developing trust, based on the premise that a person who behaves in a consistent manner will be trusted easily. In [189] experiments, the definition of reliability is broader than Sheridan's. While the latter indicates consistent functioning, the former includes the degree to which auto- matic controlled values are close to the target values as well, which includes the notion of validity, Sheridan's second dimension. Validity refers to the degree to which an automated system produces correct output. It seems intuitive that the automated system that produces more valid outputs to the human operators will be trusted more.

Third is robustness. Robustness supports expectations of future performances based on capabilities and knowledge not strictly associated with specific circumstances that have occurred before. If the automated system was designed to handle this situation whether it is (un)expected, it will still provide a useful way to control the situation. [192] performed an experiment investigating the effect of inferior decision aid on users' performance in the tax calculation domain representing highly structured environment. However, the decision aid was designed to mainly focus on the typical cases. That is, the decision aid was not robust to consider a variety of cases that can exist in the real world. Their results showed that participants' performance was significantly better with the decision aid than those without the decision aid, but only for those typical cases. For non-typical cases, participants without the decision aid significantly outperformed those with the decision aid. From the result, it appears that if the decision aid is not robust to handle many different cases, it is of no use in supporting human operators' judgment and decision-making activities.

Next is the issue of transparency. Transparency refers to the degree to which the inner workings

or logic of the automated systems are known to human operators to assist their understanding about the system. Somewhat related to transparency, understandability refers to the operators' understanding of the automation for the expectations that make the operators' trust and use the system appropriately. The operator's intervention will be better and more timely when one's trust is well calibrated to the actual trustworthiness of the system. In designing a machine to aid a human operator, understandability is affected by the degree of transparency of the system which the operator can see the underlying mechanisms through the interface.

Another way to increase the level of operators' understanding is to develop an appropriate display interface design. The interface design presumably matches the operators' concept of automated systems, so called mental model. Many studies (e.g., [193]) have claimed that an interface embedded with relationships among the components can enhance operators' understanding, which consequently contributes to increase in performance. On the other hands, [194] found no significant effect of different display types (configural display, bar graph, and alphanumeric) on operators' decision time to agree or disagree with the recommendations. In the case of a decision aid which produces the estimate of the environment, there must be a suitable presentation of the uncertainty regarding the system's inferences. This method of presentation becomes the window of opportunity for the operators to reduce the additional uncertainty created by having the automated decision aid.

The final characteristic is usefulness or utility. The usefulness of data or machines means responding in a useful way to create something of value for operators, eventually developing into trust. Conducting an experiment with two operators as a team within the tactical decision-making environment, [195] found that the operators tended to rate those parts of the decision-making support system modules that supported quick decision making, and thus were more useful, higher than other modules. However, it should be noted that these characteristics are difficult to be hard-coded into a program to manipulate the degree of system's trustworthiness due their subjective nature.

[191] list seems comprehensive in that it includes factors beyond those of [190], or of [196], that may have effect on operators' trust indirectly. For example, consider understandability and transparency. These factors may not be the ones that operators directly access when they evaluate the level of trustworthiness of an automated system. However, as operators understand the automated system better, they can grasp knowledge about the pros and cons of the system and when to rely on the system. Therefore, it seems that Sheridan's list of trust attributes provides a comprehensive understanding of operators' trust and systematic characteristics of trust that may cause operator's trust to develop.

Based on the research framework to study human trust in human-machine interaction setting, several studies performed experiments in a continuous processing domain. Among the prior research, two studies, [197], and [189], made attempts to identify the role of operator's trust in automated systems. Further, these two studies showed interesting results on the relationship between the operator's trust in automated systems and their reliance on the system measured by the use of the system. For example, [197] performed a process control simulation experiment and the results supported this perspective, showing a positive regression coefficient between operator's trust and the use of automation. They also concluded that as the level of operator's trust suffers, their use of automated system decreases. [189] conducted very similar experiments and showed that the operators

tended to use less of the automated system as the level of trust decreases.

A behavioral counterpart of operator's trust in an automated system is an index of reliance, use of automation. This is based on the assumption that the more operators trust, the more they use the automation. [198] performed an experiment using a simple computer game in which participants had to control various simple multiple tasks, part of which they could relinquish the control to automated system. Automation reliability (90%, and 50%) was controlled by setting the probability that the automation would perform a task correctly. The automation faults were distributed over time and the experimental conditions, which were different in the level of workload and in the level of uncertainty. The level of uncertainty in a task was controlled by introducing abnormalities so that the automated system could not perform the task appropriately. The level of workload was also controlled by making another separate task more complicated. The results showed that par- ticipants demonstrated a bias toward manual control, and that both college student and pilots did not delay turning on automation after a failure, and continued to rely on failed automation. Pilots showed greater tendency toward the use of automation under the lower risk. In a subsequent exper- iment focusing on the dynamics of trust, Riley controlled the level of information that participants were given about the automation prior to performing the experiment. These conditions differed in whether participants were provided with information regarding automation reliability, state uncer- tainty, or both to reveal the contributions of each element to their automation use decisions. He found that both state uncertainty and trust affect automation use decisions, but only early in the participants' experience with automation.

Although those research projects concentrated on human trust in automation based on the general understanding of supervisory control tasks, the importance of the trust concept seems applicable. It has been attempted to define the role of trust in other domains, such as computer supported cooperative work ([199]), decision-making in management ([200]), medical diagnosis expert system ([201]), computer security problems ([202]), trust in e-commerce ([203]).

Neurological approach to human trust in autonomous systems

In contemporary society, autonomous systems have become more complex because of the strong need to perform sophisticated tasks and the need to design complex control systems using advanced technology. In complex autonomous systems, the role of the human operator of controlling and monitoring is crucial to avoid failure and risk, and to prevent unpredictable situations that cannot be programmed in the system. Trust can affect the operator's degree of acceptance and reliance on the automated system, so measuring human operators level of trust is vital in predicting their strategies when interacting with the system. Previous studies investigated the relationship between social constructs (i.e., trust) and neural or physiological evidence and focused on brain regions using fMRI and efMRI, which engage with stimuli related to trust and distrust. This research measures and analyzes human brainwaves in situations involving trust and mistrust using an electroencephalogram (EEG). This experiment uses a word elicitation study taken from a survey used in previous research about words related to trust and mistrust. The international 10-20 system of electrode placement is used for recording brainwaves of participants in real time when they watch the selected words, which evoke trust and mistrust. After the experiment, the noise from the raw EEG data was removed using filters. Power spectrum analysis was used to analyze the EEG data collected. The results indicate that α and β waves are affected by trust situations while γ waves

are affected by mistrust situations. Therefore, trust can assist in effective decision-making by increasing concentration and performance, which are related to α and β waves, while mistrust can disrupt effective decision-making by increasing stress and anxiety, which are related to γ waves. The findings of this research will contribute to defining how trust in automation affects the human operator's decision-making and overall performance. In addition, it will contribute to monitoring the psychological state of the human operators in complex automations such as pilots of automated aircrafts, captains of automated ships or supervisors of automated factories.

An autonomous system is the use of various control systems for operating equipment with minimal or reduced human intervention. It has been actively used in many applications such as machinery, factories, ships, aircraft and vehicles to consistently improve quality, accuracy, productivity and efficiency. Automated systems have evolved very rapidly into complex sophisticated tasks and advanced technology. However, complex autonomous systems can increase unstable and uncertain situations in workplaces that eventually will lead to increased uncertainty, which will increase cognitive complexity [204]. In these dynamic and uncertain situations, trust can be a 'decision aid' when complex automated systems are impossible to comprehend and when flexible behaviors are necessary to combat unexpected situations that arise, which cannot be programmed in the system [204]. Trust also plays a crucial role in contributing to the cognitive complexity and increased uncertainty in sophisticated automation systems.

To understand trust between human and automation, it is necessary to study how to define trust in sociological settings. Trust has been defined related to expectation. [205] defined trust as confidence that one will find what is desires from another and [196] defined it as a generalized expectation related to the subjective probability to future events. A sociologist [206] defined trust with three specific expectations such as persistence, technical competence, and fiduciary responsibilities. By integrating the study of [206], and [196], [207] extended her study to develop trust in human-machine relationships and claimed that trust is developed by the human's ability to estimate the predictability of the machine's behaviors.

Trust between humans and machines is similar. Trust is developed over time but very hard to regain once gone. Humans (trustor) can develop trust in a machine (trustee) by evaluating the performances of the machine over time. With trust, then human (trustor) is willing to rely on the use of the machine (trustee). If the machine's outcome is unexpected and unstable involving the risk of failure or harm to humans (trustor), humans (trustor) will lose their trust and not rely on the machine in the future. Building trust is also influenced by previous experiences, the current state is to believe and the future expectation is to have faith. Therefore, the key here is maintaining a level of trust in the interaction between humans and machines.

Trust can affect the operator's degree of acceptance and reliance on the automated system. Human operator's trust in the complex automation plays an important role in decision making and solving problems during unexpected and uncertain situations. If the operator mistrust automation, he or she will tend to use manual operation in possible emergent situations [207]. Using manual operation can also increase the operator's workload needed to process all information without the assistance of automation, which will inevitably decrease quality, productivity and efficiency. If the operator overtrusts automation, he or she cannot correctly monitor and properly control the system.

Nor can the operator find the cause of the problem(s). Subsequently, these type situations become difficult to understand and to solve the problem that can cause a serious accident or failure. Hence, trust is an important factor in using automation properly. [208] states the degree of trust can influence system performance because trust can affect the operator's degree of acceptance and reliance on automation [191] and the operator's strategies regarding the use of automation [209]. Therefore, measuring the human operator's level of trust in automation becomes crucial in predicting their strategies while using automation. However, trust is a multidimensional construct influenced by various factors such as motives, intentions, and actions [210] [211], so it is challenging to measure trust. How can we measure human trust in the use of automation effectively? As a method to measure trust, questionnaires are mostly used to investigate trust in interpersonal relationships ([212] [196].) However, many questionnaires used in these studies are based on a theoretical ap- proach that does not have an objective measure and can vary by respective researchers' theoretical orientation.

To compliment a theoretical approach of questionnaires, there are researchers to use with a semi-automatic simulation of human and automation [209] and with an empirical approach ([213] [214]). [209]) used an experimental system such as a semi-automatic pasteurization plant to investigate the change in an operator's control strategies and trust during interaction with the automation. [213] used empirically developed trust questionnaires and they extensively investigated three types of trust, which are human-human trust, human-machine trust, and trust in general, with a three phased experiment of a word elicitation study, a questionnaire and a paired compari- son study. The use of questionnaires with simulations or empirical approaches has great potentials to measure trust effectively, but the questionnaires cannot be a direct method to measure trust. Trust is a multidimensional concept, so it is required to measure trust in various perspectives.

To supplement the questionnaires, this research proposes use a neurological measure, which can be a direct and objective method to detect and measure human trust. This research uses an empirically developed wordlist related to trust and mistrust from a word elicitation study of [213] to evaluate the subjective feeling of trust with the neurological measure. Some studies have investigated the correspondence of relationship between social constructs (i.e., trust) and neural or physiological evidence. [215] investigated the correlation between the degree of trust and human facial appearance using event-related functional magnetic resonance imaging (efMRI). They found that trustworthiness was correlated in a frontopolar cortex (frontal lobe) and untrustworthiness was correlated with right amygdala and right insula (temporal lobes). [216] examined interpersonal trust when two strangers interacted online with one another in a trust game and detected differ- ent brain regions based on conditional and unconditional trust in a partnership using hyper-fMRI. They discovered that conditional trust activated the ventral tegmental area (frontal lobe) while unconditional trust activated the septal area (temporal lobes). [217] had an experiment to find a correlation between brain regions and trust using fMRI by measuring response of two fictitious online sellers who have high and low reputation. They found that trust is related to the caudate nucleus, putamen, anterior paracingulate cortex, and orbitofrontal cortex (frontal lobe) while distrust is related to the amygdala and insular cortex (temporal lobes). These previous studies focused on brain regions using fMRI and efMRI, which engage with stimuli related to trust and distrust. However, the previous studies did not fully investigate brainwaves in the situations of trust and mistrust. fMRI and efMRI resolution provide excellent spatial resolution and good temporal without

no radiation and no side effects from the magnetic fields and radio waves, so they are very useful to detect brain regions with specific stimuli. However, both fMRI and efMRI are claustrophobic, noisy, and dangerous to any present of mental in the body, and very sensitive to moving objects [218]. In other words, fMRI and efMRI are not effective tools measure various cognitive tasks in a real time.

This research measures and analyzes human brainwaves using situations involving trust and mistrust using an electroencephalogram (EEG). Electroencephalography (EEG) is an electrophysiological monitoring method that records the electrical activity of the brain by placing electrodes on the scalp. EEG has several benefits for measuring brain activity. EEG has a wide range of intra-individual and inter-individual variability which can be divided in frequency bands (δ, θ, α, β and γ) [219]. EEG detects changes in the brain's electrical activity in response to external stimuli in various regions within a millisecond timeframe in real time. EEG can be sensitively changed to the external stimuli, so it is useful for experiments with a wide range of cognitive tasks, which requires to observe responses by external stimuli. In addition, EEG provides high resolution without exposure to radiation or magnetic fields and it is not expensive compared to other equipment types, portable, and is not claustrophobic.

Human Brain regions and EEG

Neuroimaging research is useful to investigate neural activities with a wide range of cognitive processes with attention, language and memory. Particularly, neuroimaging techniques contributes to the study of cognition. Neuroimaging techniques can assist with examining various cognitive processes including language, reasoning, problem solving, decision making and other complicated mental functions [220]. Neuroimaging techniques allow us to find out how neural activities relate to any particular cognitive task through detecting brain regions and brain waves. This part will investigate the cognitive process through examining the human brain, EEG history, EEG analysis, brain waves, brain regions, and brain imaging methodologies. Also, a clear connection to measure human trust using EEG will be articulated.

The human brain is the center of the central nervous system, which consists of the cerebrum, brainstem and cerebellum. The cerebral cortex is the outer layer of the neural tissue in the cerebrum that plays an important role in memory, attention, perception, awareness, thought, language and consciousness. The cerebral cortex is divided into four lobes - the frontal, parietal, temporal and occipital lobes. The frontal lobe involves with executive functions, the parietal lobe involves with sensory information, the temporal lobe involves with processing sensory input, and the occipital lobe involves with visual information through the visual cortex.

The cerebral cortex is relative to the cognitive process, namely, frontal lobe, parietal lobe, occipital lobe and temporal lobe (see Figure 157. The frontal lobe plays a crucial role in the brain as a control tower because of its association with memory, willpower, logical thinking, adjusting the activities of other regions of the brain, and controlling emotion and impulse [221]. In addition, the frontal lobe is associated with problem-solving, judgment, attention, organization, planning, and anticipation [222]. Particularly, the prefrontal cortex (PFC), which is the cerebral cortex that covers the front part of the frontal lobe, performs executive functions. This involves organizing and executing a series of actions for complex cognitive goals by predicting consequences of current activities, and making decisions by differentiating among conflicting thoughts and determining good and bad ([222], [223]). To conduct actions in a timely manner, the prefrontal cortex needs direct

access to sensory, motor and mnemonic information [222]. A damaged prefrontal cortex could result in antisocial, violent and psychopathic behaviors [223].

The parietal lobe of the brain is located above the occipital lobe and behind the frontal lobe and central sulcus. The parietal lobe deals with language, sensation, perception and integrating sensory inputs, especially the sense of touch, pain and temperature. The posterior parietal cortex is highly associated with visuomotor integration, spatial perception and spatial orientation, because of its location between the sensory and motor area that are related to both sensory-related and motor-related activity [224].

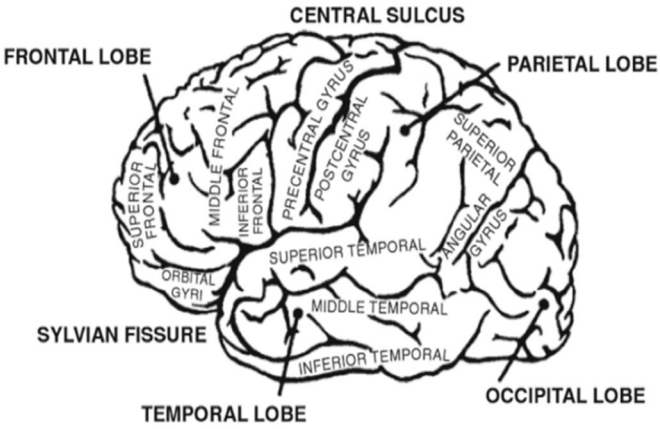


Figure 169: Brain Anatomy. Source: [225]

The occipital lobe is located under the occipital bone at the lower back of the skull and is divided into two areas, dorsal (superior) and ventral (inferior). As a visual processing center, it houses the visual cortex which functions visuospatial processing, color differentiation, and motion perception. Damage to the right hemisphere of the occipital lobe can cause serious problems with the visual perception [225].

The temporal lobe is interconnected with other cortical regions for executing complicated functions [225]. This lobe is associated with the perception of visual and auditory sensation related to speech and language processing, language comprehension, long term memory storage and facial and object recognition and emotion. A damaged temporal lobe can cause perseverative speech, paranoia, and rage. Amygdala, which is deep inside the temporal lobe is known for controlling major affective activities like friendship, love and affection and the expression of moods such as fear, rage, and aggression. The amygdala is the center for identifying danger that aids in self-protection, when activated brings fear and anxiety [226].

The human brain functions when neurons, electrically excitable cells process and transmit information, are electrically charged by membranes that transport proteins to pump ions across their membranes and pass electrical signals to other neurons through a synapse, a specialized connec- tion with other cells. Neurons make up the core components of the brain and the spinal cord (the central nervous system (CNS)), and the ganglia (the peripheral nervous system (PNS)). These elec- trical signals affect all the processes in the brain; therefore, understanding how the brain works is important (e.g., how neurons communication, how stimuli are maintained to function correctly, how attention is used selectively, and how memory is made and stored to understand how neurons

communicate). These electrical signals can be detected and recorded using electroencephalography (EEG). EEG is an electrophysiological monitoring method used to record the electrical activity of the brain. EEG measures voltage fluctuations that resulted from ionic current within the neurons of the brain detected by multiple electrodes placed on the scalp.

The use of EEGs to measure brain activity dates back more than a century ago. In 1875, Richard Caton (1842-1926), an English physician, reported electrical activities from the cerebral cortex of rabbits and monkeys in the British Medical Journal. German physiologist and psychiatrist Hans Berger (1873-1941) recorded the first electrical activities of human brain by inserting two platinum electrodes under the damaged cranial bone and called it electroencephalography (EEG) similar to an electrocardiogram (ECG) or electromyography (EMG).

3.3.8.5.1 EEG Band related to Cognitive Process

Brain waves consist of different frequency waves, which allows one to analyze the waves using a specific frequency. The Power Spectrum analysis allows for classifying the brain waves by frequency ranges: delta waves (0.2-4 Hz), theta waves (4-8 Hz), alpha waves (8-13 Hz), beta waves (13-30 Hz) and gamma waves (30-60 Hz). The classification of the beta waves is low beta (13-15 Hz, which is called as SMR), mid beta (15-20 Hz) and high beta (20-30 Hz).

Delta waves have a frequency between 0.2 Hz and 4 Hz. Usually, these waves are common in infants or in normal adults in a deep state of sleep. They tend to be the highest in amplitude and the slowest waves. When delta waves are strong, the body is healing itself by resetting its internal clock. Delta waves are common in people in a coma or have a defect of the cerebral cortex. Delta wave frequency and the frequency of the movement of the eyes or bodies to objects are similar. During an experiment conducted to record brain waves over time due to unavoidable noise, the increased power of the delta waves was not seriously considered for analysis.

Alpha waves have a frequency between 8 Hz and 13 Hz. The waves appear when people are awake but have a relaxed awareness and inattention without processing much information. When people are more comfortable and relaxed, the alpha waves increase. The waves generally appear as a regular wave, detected strongly over the occipital lobe, and in small amounts over the frontal lobe. They emerge with the closing of the eyes and with tension, and are suppressed during eye opening or mental exertion. These waves relate to recalling memory, relieving pain and reducing stress and anxiety. Therefore, stimulating alpha waves can improve the efficiency of learning and working.

Beta waves have a frequency between 13 Hz and 30 Hz. These waves are detected mostly in the front lobe and are associated with conscious activities. They can affect cognitive tasks such as attention, problem solving, judgment and decision-making. In sustained attention tasks, low beta (SMR) waves affect perceptual sensitivity and lower omission errors, while mid-beta waves reduce reaction times. Beta waves can be a guideline for measuring the cognitive process. High beta waves are associated with intensity, anxiety and hypervigilance. People that lack sufficient beta activity can experience mental or emotional disorders such as depression, Attention Deficit Disorder (ADD) and insomnia. Beta activity is important because it can enhance concentration, attention, emotional stability and energy levels.

Gamma waves have a frequency between 30 Hz and 60 Hz. These are the fastest waves, indicative of anxiety and processes advanced cognitive information such as reasoning and judgement. They are connected to the development of ideas, the use of language, processing memories and learning. They are detectable when people use their short-term memory to identify objects, sounds and tactile sensations.

Theta waves have a frequency between 4 Hz and 8 Hz. These waves occur in young children and are present in adults and teens during drowsiness. They are usually detected when people are falling asleep or a state of calming down. Theta waves are associated with access to unconscious mind, creative inspiration and deep meditation, but this has not been proven because of lack of standardized results. They can be used in hypnotherapy and self-hypnosis during a receptive mental state to accept affirmations and suggestions.

3.3.8.5.2 Experiment I: Human Trust and EEG

The word trust originated with trust, which means comfort in German. If I trust someone, it means that I am comforted by the belief that he or she will not do any action contrary to my ex- pectation. Many researchers have proposed to define trust between humans related to expectation. Barber [206] defined trust as "confidence that one will find what is desired from another rather than what is feared." Others summarized trust as "an actor's willingness to arrange and repose his or her activities on others because of the confidence that others will provide expected gratification." It was also defined the concept of interpersonal trust as "a generalized expectancy held by an individual that the world promise, oral or written statement of another individual or group can be relied on." Rempel et al. [196] explained trust as "a generalized expectation related to the subjective probabil- ity of an individual assign to the occurrence of some set of future events," and later, Rempel and Holmes [196] defined trust as "the degree of confidence one feels when thinking about a relation- ship." Barber [206] understood the multidimensional concept of trust and defined trust with three expectations of "the persistence of the natural and the moral social orders, technically competent role performance, and fiduciary obligations and responsibilities." Based on the definitions of trust by Barber [206] and Rempel and Holmes [196] , Muir [207] defined trust in the human-machine relationship as such: "Trust is the expectation, held by a member of a system, of persistence of the natural and moral social orders, and of technically competence performance, and of fiduciary responsibility, from a member of the system, and is related to, but not necessarily isomorphic with, objective measures of these qualities." Therefore, it is necessary to understand trust as a multidi- mensional concept. Trust is highly related to the idea of social influence because it would be much easier to influence or persuade someone who is trusting. Trust can also have a positive influence on a person's behaviors, perceptions and performances. Trust can predict the acceptance of behaviors by other individuals, social groups such as friends, communities, organizations, companies, nations and objects such as machines and automated systems. Trust can play the important role of "deci- sion aid" in dynamic and uncertain situations when complex automated systems are impossible to comprehend and when flexible behaviors are necessary to combat unexpected situations that arise, which are unavoidable.

How can we measure human trust effectively? Because trust is a multidimensional concept, mea- suring a worker's level of trust becomes crucial in predicting his or her decision-making while using the systems. Many researchers have used questionnaires that measure subjective feelings of trust. However, the questionnaires used in these studies are based on a theoretical approach, which can vary by the researcher's theoretical orientation. Lee and Moray [209] used an experimental system, such as a semi-automatic pasteurization plant, that posed limitations when being applied to other automated systems due to the systems having different tasks or levels of complexity. Because trust is a multidimensional concept, we needed to use both quantitative and qualitative methods to measure trust. As a method for this research, we used both a neurological imaging technique as a quantitative

method and a questionnaire as a qualitative method to measure human trust. Neurological imaging techniques can measure quantitative aspects of trust by investigating relations between neurological activities and human trust. Up to now, there have been some studies investigating trust and neural correspondence. Winston et al. [215] investigated the correlation between the degree of trust and human facial appearance using event-related functional magnetic resonance imaging (efMRI). Krueger et al. [216] examined interpersonal trust in an online context to detect different brain regions based on conditional and unconditional trust using hyper-fMRI. Another study defined a relationship between brain areas and psychological processes using fMRI that revealed a correlation between trust and mistrust with different brain areas. These studies focused on brain regions using fMRI and efMRI that engaged with stimuli related to trust and mistrust. However, the previous studies did not fully investigate to identify specific brainwaves in the situations of trust and mistrust.

The aim of this research is to understand human trust by measuring and analyzing human brainwaves captured with an electroencephalogram (EEG). The experiment consisted of a word-elicitation study used to identify specific brainwaves (i.e., delta, theta, alpha, beta and gamma) which can aid in understanding brain activities relative to trust and mistrust. The aim of this research is to understand human trust by measuring and analyzing human brainwaves captured with an electroencephalogram (EEG). Thus, the research investigated human trust and mistrust by analyzing the neurological activities (brainwaves) between human operators. Specifically, this research em- ploys two experiments. Experiment I consisted of a word elicitation study used to identify specific brainwaves and brain regions that resulted from trust and mistrust situations. Identifying specific brainwaves (i.e. delta, theta, alpha, beta and gamma) and brain regions (i.e. frontal, parietal, temporal, central and occipital lobes) can aid in understanding brain activities relative to trust and mistrust.

3.3.8.5.3 Experiment I with words

Experiment I is designed to measure neurological activities in the situations of trust and mistrust using a word list related to trust and mistrust. It adopts the framework from a previous study about words related to trust and mistrust. The previous study collected 96 words related to trust using a word elicitation study by rating an initial set of 138 words from dictionary definitions and thesauri and adding 38 words obtained from the written descriptions about trust from participants in Linguistics or English graduate students. Then they used the questionnaire study to 120 students in various majors identify the limited set of words related to trust and distrust using a seven-point scale. Through the word elicitation study and the questionnaire study, they had a figure, which contains sets of the top 5, 10 and 15 words related to trust and distrust. After the questionnaire study, they did factor analysis and cluster analysis to compare three types of trust, which are general trust, human-human trust, and human-machine trust. Finally, they confirmed that the three types of trust are very similar in sets of words related to trust. This experiment utilizes only the top 15 ranked words related to trust and mistrust from the study of Jian et al (2000) to measure general concept of trust.

3.3.8.5.4 Hypotheses

- In the situation of general concept of trust, specific brainwaves and brain regions will be

detected.

- In the situation of general concept of mistrust, specific brainwaves and brain regions will be detected.

3.3.8.5.5 Recruitment of participants

A total of 18 participants (male and female) were solicited to participate in the experiment I. All participants would encompass undergraduate and graduate students recruited on campus at North Carolina A&T State University.

3.3.8.5.6 Participants demographics

Each participant was over 18 years old, with normal or corrected to normal vision, was free of current or past neurological and psychiatric disorders and was able to read and comprehend the English language. There would be no preference on right-handed or left-handed user, but all participants must have the ability to use a keyboard and a mouse with their hands. There would be no discrimination upon participants based on age, race, gender, religion, or prior participation in other experiments. Each participant was in an isolated room under guided instruction. Conditions for all participant were same.

3.3.8.5.7 Instrumentation

The EEG data was recorded using the g.HIamp (256 multichannel amplifier), g. GAMMAsys (active electrode system with g. GAMMAcap) and g. Recorder (brain signal recording software) by g. tec medical engineering company. Per the International 10-20 system of electrode placement, 20 electrodes (Fp1, Fp2, Fpz, F7, F3, Fz, F4, F8, T7, C3, Cz, C4, T8, P7, P3, Pz, P4, T6, O1, O2) were recorded and measured (see Figure 2). The 10-20 system is based on the relationship between the location of an electrode and the underlying area of the cerebral cortex. The letters F, T, C, P, and O represent the frontal, temporal, central, parietal, and occipital lobes. Even numbers refer to the right hemisphere of the brain and odd numbers refer to the left hemisphere of the brain. The participants head is fitted with a cap of electrodes (g. GAMMAcap), and the selected 20 electrodes are filled with abrasive electrolyte gel using a syringe. The sampling frequency to record brainwaves is 256 Hz, high-frequency filter as 60 Hz, and low-frequency filter as 0.1 Hz. The g.Recorder (brain signal recording software) also included artifact removing filters for eye blink, jaw clenching and muscle movement. Before an experiment, participants artifacts were recorded, marked and removed from the raw data for the experiment.

3.3.8.5.8 EEG procedure

At the beginning of the experiment, each participant reads informed consent agreement. Afterwards, the researcher addresses any questions and concerns. All participants are required to provide demographic information such as age, gender, and contact information as well as a signed and dated consent form. After each participant signs the consent form, the participant is asked to put aside all personal items on a secure table. The participant is asked to turn off their cell phone to avoid any interruptions during the experiment. After the participant is seated, detail instructions are given for each experiment. The facilitator in the room addresses any questions or concerns related to the experiments. Then the participant is fitted with a g. GAMMA cap with electrodes that measures



Figure 170: Example of g. GAMMAcap & g.HIamp

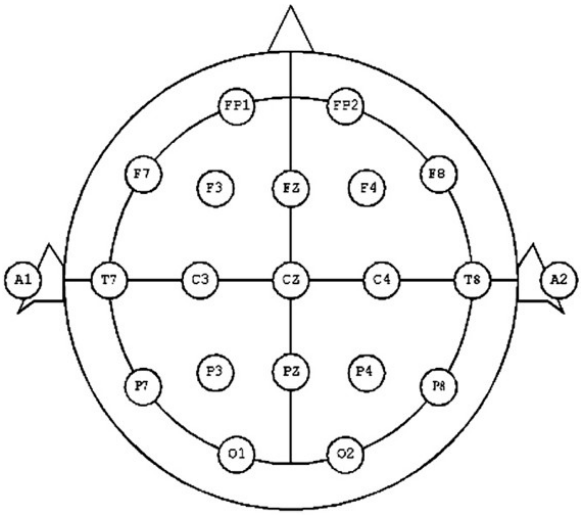


Figure 171: International 10-20 system for electrode placement

small electrical signals produced by the brain in a completely non-invasive manner. The 20 electrodes, per the international 10-20 system for electrode placement is filled with a gel using a syringe with a plastic tip. Before each experiment begins, participants are asked to record their brainwaves for marking and removing artifacts such as eye blink, jaw clenching and muscle movement. Then, as baseline data without any stimuli, participants brainwaves are recorded when they are relaxed, but they are asked not to move, not to talk and see the blank monitor for a minute. When an experiment begins, the participant is asked to focus their attention on the computer screen and use the keyboard or mouse if needed. They are asked not to talk and blink their eyes (if possible) to reduce recording artifacts in the brainwaves. After the experiment, each participant gently removes the electrode cap and any remaining gel from their head or hair using the tissues and/or towel provided.

3.3.8.5.9 Experimental design

Experiment I with words is designed to measure neurological activities in the situations of trust and mistrust using a word list related to trust and mistrust. It adopts the framework from a previous study about words related to trust and mistrust. The top rating of 15 words are selected from the previous study and presented in the experiment randomly. All 18 subjects participated in the experiment. The participants were expected to think about situations of trust and mistrust evoked by the selected words. Before the experiment began, the instruction was displayed for 10 seconds. Each word displayed for 7 seconds in a slideshow, which was enough time to read and understand the meaning of each word. The slideshow lasted for a total of 3 minutes and 30 seconds, which was enough time to evoke a situation of trust or mistrust. Each experiment consisted of two slideshows: 1) trust words and 2) mistrust words. The experiment I is designed to record trust first and mistrust later without counter-balancing design because it evokes emotional responses. Especially mistrust can evoke negative emotions such as stress and anxiety, which can last longer and affect other emotions. The first slideshow consisted of a slideshow with 30 random stimuli which was sufficient stimuli for recording brainwaves by displaying the 15 words related to trust twice in a random order. The participants brainwaves were recorded during the slideshow with words related to trust. The participant took a one-minute break in order to avoid fatigue and stress. Similarly, the second slideshow consisted of a slideshow with 30 random stimuli which was enough stimuli for recording brainwaves by displaying the 15 words related to mistrust twice in a random order. After the experiment, the participants completed a survey to evaluate the level of trust for trust and mistrust word sessions.

- Words related with trust: honesty, royalty, love, confidence, assurance, friendship, security, integrity, fidelity, familiarity, honor, reliability, trustworthy, entrust, promise.
- Words related with mistrust: cheat, betray, deception, steal, suspicion, distrust, sneaky, misleading, mistrust, phone, beware, harm, falsity, lie, cruel.

3.3.8.5.10 Analysis

This research used both power spectrum for identifying specific brainwaves and coherence analysis for detecting active brain regions. Analyzing brainwaves consisted of linear spectral measurements, nonlinear measurements of complexity, and interdependency measurements. The benefits of using linear spectral measurements (power spectrum and coherence analysis) are results of obtainable fast and easy; thus, most often used.

One linear spectral measurement termed the power spectrum measurement is a common method to quantify EEG raw data to illustrate the distribution of signal power by frequency. To identify specific brainwaves corresponding to the words for trust and mistrust situations, this involved analyzing the EEG data using the power spectrum through the EMSE Suite Data Editor by Cortech Solutions. Power spectrum analysis involved performing a Fast Fourier Transform (FFT) on defined data intervals. Then, the results are squared and averaged. The power spectrum data is displayed in units of Volts-square(V^2).

$1.0\mu V^2 = 1.0(\mu V^2) = 1.0e^{-0.12V^2}$

Power spectrum analysis is helpful in analyzing each channels frequency; however, this analysis has limitations when examining complex and comprehensive activities of the brain such as trust and mistrust situations. Therefore, this study used a coherence analysis to detect active brain regions and to examine communications among the active brain regions.

Using coherence analysis (based on the power spectrum) for functional connectivity in the different brain regions has gained frequent use. Coherence is a measurement of the amount of phase stability or phase jitter between two different time series. When the phase difference between two signals is constant, coherence =1; when the phase difference between signals is random, coherence = 0. To obtain coherence when two channels (a, b) are recorded from a specific range of the frequency (f); the square of cross-spectrum (Pab) needs to be divided by power spectrum Pa(f) and multiplied by power spectrum Pb(f).

$$Coh_{ab}(f) = \frac{P_(ab)(f)^2}{[P_a(f) \times P_b(f)]}$$

3.3.8.5.11 Results

This research recorded the participants' brainwaves as they responded to select words relative to trust and mistrust by the 10-20 systems of electrode placement. The total time for each session was 210 seconds. The brainwaves of the 17 participants were normal without any extreme brain activity, but the brainwaves of the one subject had abnormal brain activity, so the collected brainwaves of 17 subjects were usable for analysis.

First, the power spectrum analysis was used to analyze the raw data generated from the brainwaves to examine specific brainwaves for the trust and mistrust situations. The researcher used the power spectrum analysis to classify the frequency ranges by brain waves as follows: delta (δ, $0.2-4Hz$), *theta* (θ, $4-8Hz$), *alpha* (α, $8-13Hz$), *beta* (β, , $13-30Hz$) *andgamma* (γ, $30-60Hz$). Delta and theta waves were not relative to this experiment because delta waves occur in a deep sleep state and theta waves occur during drowsiness; thus, these were not applicable for this experiment. Therefore, the analysis includes the alpha, beta, gamma waves, intraindividual and average differences that are compared to the trust and mistrust situations. Further, the individual variability for the brainwaves is high, so the analysis includes the intraindividual differences, which shows individual difference with baseline and two stimuli of trust and mistrust situation, followed by the average differences. Second, the power spectrum and coherence analyses aided in investigating the active regions of the brain (frontal, temporal, central, parietal and occipital areas) for the trust and mistrust situations. The power spectrum analysis aided in investigating the stimulated region with the absolute power value. The regions of the brain were compared to the trust and mistrust

situations. The coherence analysis was used to compare trust and mistrust situations to the brain regions by investigating functional connectivity in each region based on a relative value scale of 0 to 1. According to the modified 10-20 system of electrode placement, each channel is assigned a location such as frontal lobe (F3, F4), temporal lobe (T3, T4), central lobe (C3, C4,), parietal lobe (P3, P4) and occipital lobe (O1, O2).

3.3.8.5.12 Comparison of Intraindividual and Average Differences

The frequency of alpha waves ranged from 8 Hz to 13 Hz. By comparing the intraindividual differences for the trust situations to the baseline (with no stimuli), the power of alpha waves of 16 of 17 participants significantly increased and the power of 1 participant (P16) decreased. By comparing the intraindividual differences for the mistrust situation to the baseline (with no stimuli), the power of the alpha waves for 14 of 17 participants slightly decreased but the power of 3 participants (P2, P4 and P5) increased.

The averages for the baseline, trust, and mistrust were 2.63E-10 μV^2, 4.15E-10 μV^2 and 2.08E-10μV^2 respectively. By comparing these averages, the power of alpha waves increased for the trust situation with a difference of 1.53E-10 μV^2 and decreased for the mistrust situation with a difference of 5.50E-11 μV^2.

The frequency of beta waves ranged from 13 Hz to 30 Hz. By comparing the intraindividual differences for the trust situation to the baseline (with no stimuli), the power of beta waves for 16 of 17 participants significantly increased, but the power of 1 participants (P4) decreased. By comparing the intraindividual differences for the mistrust situation to the baseline (with no stimuli), the power of the beta waves for 14 of 17 slightly increased, but the power of 3 participants (P1, P3 and P5) decreased.

The averages of the baseline, trust and mistrust were 1.58E-10 μV^2, 2.23E-10 $\mu V^2$and 1.65E-10 μV^2, respectively. A comparison of these averages showed the power of beta waves significantly increased for the trust situation to a difference of 6.50E-10 μV^2 and slightly increased for the mistrust situation to a difference of 7.00E-12 μV^2.

The frequency of gamma waves ranged from 30 Hz to 60 Hz. A comparison of the intraindividual differences for the trust situation to the baseline (with no stimuli) showed a slight decrease for the power of gamma waves for 14 of 17 participants, but not in 3 participants (P6, P9 and P12). A comparison of the intraindividual differences for the mistrust situation to the baseline (with no stimuli) showed a significant increase for the power of the gamma waves for 15 of 17 participants, but not in 2 participants (P2 and P9).

The averages of the baseline, trust, and mistrust were 2.48E-11 μV^2, 2.36E-11 $\mu V^2$and 3.38E-11 μV^2, respectively. A comparison of these averages showed a slight decrease for the power of gamma waves for the trust situation with a difference of 1.20 E-12 μV^2 and showed a significant increase for the mistrust situation with a difference of 9.00 E-12 μV^2.

Since the 4 participants brainwaves were normal without any extreme brain activity, the analysis was performed on the averaged data of all the brainwaves. The participants brainwaves were recorded as they responded to the word selections relative to trust and mistrust by the 10-20 system of electrode placement (22 channels) with a total time of 210 seconds for each session. First, the power spectrum was used to analyze the raw data generated from the brainwaves. Then, the average of the power spectrum results was used in the coherence analysis.

The power spectrum analysis categorized the brainwaves by frequencies: alpha (8-13 Hz), beta (13-30 Hz), and gamma (30-60 Hz). For a more detailed analysis in the future, the classifications

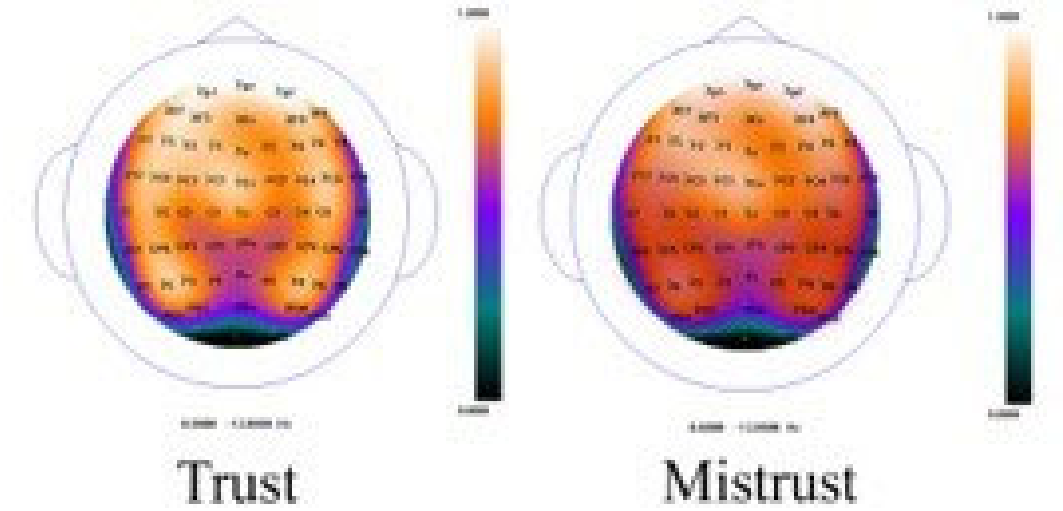


Figure 172: Coherence topography map (alpha waves)

of beta waves are as follows: low beta (13-15 Hz, called Sensorimotor Rhythm), mid beta (15-20 Hz) and high beta (20-30 Hz). The study examined the experimental results for the trust and mis- trust situations by using the frequencies of the active brain regions (alpha, beta, and gamma waves).

In the trust situations, the alpha and beta waves were stronger than mistrust (see Figure 161 & 162). The simulation of the frontal and parietal lobes also showed activity in the trust situations in the alpha and beta wave frequencies. By contrast, in the mistrust situations, the gamma waves were stronger than trust. Also, the temporal and occipital lobes showed engagement in the mistrust situations in the gamma frequency waves (see Figure 163).

3.3.8.5.13 Comparisons of trust and distrust

By comparing the alpha waves, beta waves and gamma waves in the trust and mistrust situation, the alpha and beta waves are associated with the trust situation and the gamma waves are associated with the mistrust situation, so they need to analyze with more details. By comparing the trust differences to the alpha waves when the baseline is 0, 94.11% of participants increases in the trust situation and the standard deviation is 2.25E-10. By comparing the trust differences to the beta waves when the baseline is 0, 94.11% of participants increases in the trust situation and the standard deviation is 7.96E-10) . By comparing the mistrust differences to the gamma waves when the baseline is 0, 88.23% of participants increases in the mistrust situation and the standard deviation is 1.09E-11)

3.3.8.5.14 Comparison of stimulated brain regions to the baseline, trust, and mis-trust situations

Comparing the stimulated brain regions of the alpha waves to the baseline, trust, and mistrust situations, it revealed that the frontal lobe (3.71E-09 uV2) for the trust situation was highly stimulated compared to the other regions. According to the topography map with the average data, the alpha waves in the frontal lobes showed active connectivity for the trust situations, but the right

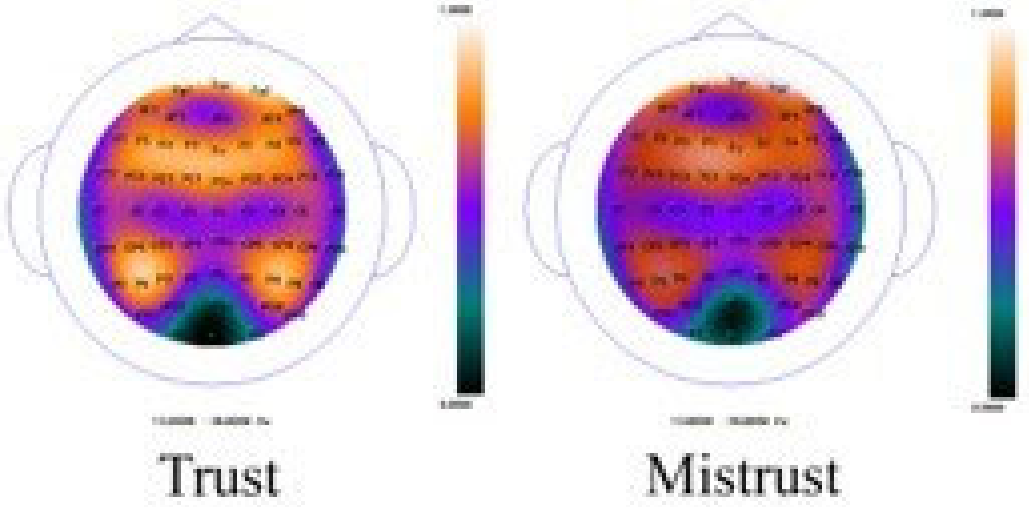


Figure 173: Coherence topography map (beta waves)

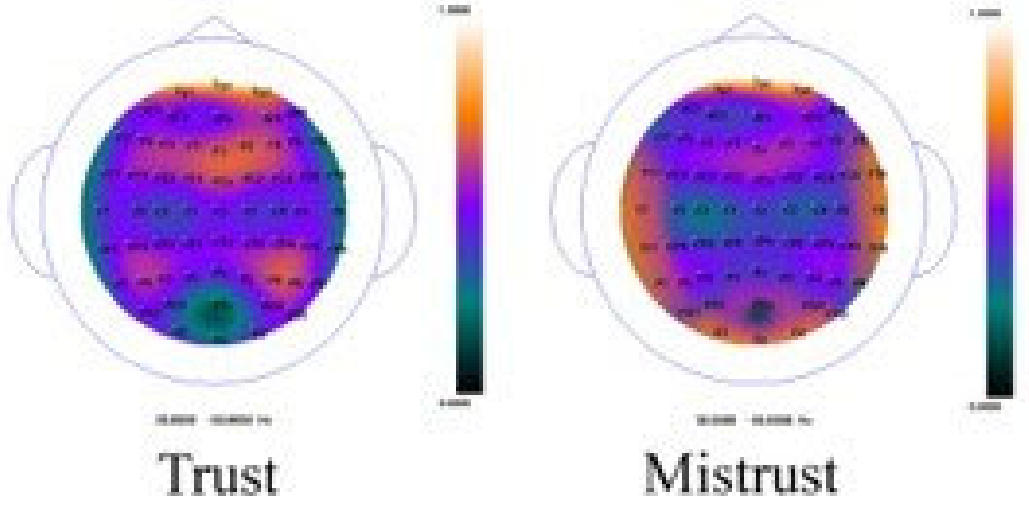


Figure 174: Coherence topography map (gamma waves)

central and temporal lobes showed little connectivity for the mistrust situations.

Comparing the stimulated brain regions for the beta waves to the baseline, trust, mistrust situations, it also revealed that the frontal lobe (1.01E-09 uV2) in the trust situations was highly stimulated compared to the other regions. According to the topography map with the average data, the beta waves in the frontal lobes also showed active connectivity for the trust situations. However, the temporal lobe showed little connectivity for the mistrust situations.

Comparing the gamma waves for the stimulated brain regions for trust, mistrust, and the baseline. The temporal lobe (3.15E-10 uV2) for mistrust was highly stimulated compared to the other regions. According to the topography map with the average data, the gamma waves in the temporal lobes showed active connectivity for the mistrust situations, but there is not active connectivity for the trust situation.

3.3.8.5.15 Detailed comparison of average and exceptional topography maps in the trust and mistrust situations

By comparing average and exceptional topography maps of the alpha waves in the trust situation, the average topography map showed the active connectivity in the frontal lobe, but the other typography map showed the active connectivity in the parietal lobes. This exceptional topography map is from P16, which is the participant who decreased alpha waves while 16 of 17 participants increased.

By comparing average and exceptional topography maps of the beta waves in the trust situa- tion, the average topography map showed the active connectivity in the frontal lobe, but the other typography map showed the active connectivity in the parietal lobes. This exceptional topography map is from P4, which is the participant who decreased beta waves while 16 of 17 participants increased in the trust situation.

By comparing average and exceptional topography maps of the gamma waves in the mistrust situation, the average topography map showed the active connectivity in the temporal lobe, but the other typography map showed the active connectivity in the occipital lobes. This exceptional topography map is from P2, which is the participant who decreased gamma waves while 15 of 17 participants increased in the mistrust situation.

3.3.8.5.16 Conclusion

Based on the power spectrum analysis, a comparison of trust and mistrust situations by α, β, andγwaves is shown. While alpha and beta waves were stronger in the trust situations, gamma waves were stronger in the mistrust situations. Alpha waves represent a calm state without any stress or tension and beta waves correspond cognitive tasks such as attention, problem-solving, judgment, and decision-making. Gamma waves denote complicated cognitive processes with significant stress and anxiety. By examining alpha, beta and gamma waves, trust situations can help calm down or execute normal cognitive tasks. However, mistrust situations can interrupt mental activity because of increased stress and anxiety.

The coherence analysis consisted of a simulation of specific brain regions derived from the trust and mistrust situations. While the frontal and parietal lobes showed active communication for

the trust situations, the temporal and occipital lobes showed active engagement for the mistrust situations (See Figure 161-163). To investigate the simulation of the trust and mistrust situations, an understanding of the function of the specific brain region related to cognitive process are necessary.

Trust situations seem to affect the frontal and parietal lobes, which proves that trust is highly related and important in the cognitive process such as processing information from senses, controlling emotions, solving problems and making decisions. The frontal lobe plays a crucial role in the brain as a control tower because of its association with memory, willpower and logical thinking, and adjusting the activities of other regions of the brain and controlling emotion and impulse ([221]). In addition, it is related to problem-solving, judgment, attention, organization, planning, and anticipation ([222]). The parietal lobe of the brain deals with sensation, perception and integrating sensory inputs, especially the sense of touch.

In this study, mistrust situations activated the occipital and temporal lobe; therefore, proving that mistrust situation increases the workload for perception from visual and auditory sensation and evokes intensive emotions such as fear and anxiety. The occipital lobe is in the lower back of the brain. This is the visual processing center that house the visual cortex that functions visuospatial processing, color differentiation, and motion perception. The temporal lobe deals with sensation such as visual and auditory inputs, language comprehension, memory, and emotion. Especially, a damaged temporal lobe can cause perseverative speech, paranoia, and rage. Amygdala, which is deep inside the temporal lobe, which is known for controlling major affective activities like friendship, love and affection and the expression of moods such as fear, rage, and aggression. The amygdala is the center for identifying danger for self-protection, when activated brings fear and anxiety ([226]).

3.3.8.5.17 Discussion

As denoted by the findings of this research, trust can assist in effective decision-making by increasing concentration and performance. However, mistrust can impede the effectiveness of decision-making by increasing stress and anxiety. Even though there were limited samples of brainwaves, this research presents the differences between specific brainwaves, in response to the words related with trust and mistrust, and the differences between the brain regions. Further research with more samples of brainwaves and brain regions will aid in further investigation of complex cognitive processes between trust and mistrust; eventually, to define how trust can affect performance and decision-making.

While considering a human-machine system, especially if they consist of any kind of automation, it is important to interpret the human brain signal correctly in a quantitative way. Over the years, studies have been conducted to understand human decision-making behavior and trust in different contexts. Along with studying different somatic markers, neural signals have been analyzed using EEG, fNIR, etc.

One of the common challenges while studying neurological signals, it is hard to identify the specific mental function that is associated with the neural signal in the instance. While we are interpreting signals to understand 'decision making', other mental influence can be involved, such as emotion, analysis, stress, social context, etc. Similarly, while analyzing neural signals associated with trust/mistrust, there might be other influences like emotion, experience, training, etc. In Brain-Computer Interface and human-automation teaming, a very frequently faced issue is decision making based on trust/mistrust. But it is hard to measure them together. Several studies have been associated with decision-making based on trust in the machine. An experiment has been conducted before in this project to study human decision-making based on trust/mistrust involving automation

using a simulated driving car, where the participant i.e. the human operator is to choose between manual and automatic control. In our study, to understand the brain function more profoundly, we aimed to study neurological function from two individual aspects- decision making and trust.

3.3.8.5.18 Decision Making Decision making is a complex process and cannot be described in a 'simple' form. This single mental task can be divided into three distinct steps: (1) the assessment of options and formation of preferences (2) the selection and execution of an action, and (3) the experience or evaluation of an outcome. And several dimensions can affect decision making- 1) uncertainty, 2)reward or punishment effect, 3) Experience, 4) group decision vs individual decision, 5) Contextual characteristics. Some studies found that decision making can consist of two parallel processes may come with conflicting outcomes, one is evolutionarily older, faster, kind of automatic and thus not very accessible to conscious awareness and control; and the other process is based on rules and algorithms, formal logic slower. Using neuroimaging studies or brain lesion patients and decision-making paradigms, several investigators have shown that structures important for the processing of reward, punishment, positive and negative moods, as well as self-relevant processing are also active in decision-making situations. Decision making has an association with cognitive operations for logical reasoning, cognitive control. Under risk situation, decision making neural association has been found with parietal, occipital, and temporal regions with an fMRI study.

We plan to approach decision making as a 'single' task and as free as possible other cognitive tasks. So we aim to study binary situations, like- 'go'/'no-go', 'yes'/'no', 'right'/'left' based on several stimuli set-ups. Different 'go'/'no-go' studies have been conducted to understand the neural correlates, but most of them focused on some specific context, such as- with children, social drinkers, ERP analysis with complex tasks.

3.3.8.5.19 Trust and Decision Making

For trust has several dimensions, we aim to understand neurological activities regarding basic trust, mistrust situation. Like decision making, trust is also related to other mental factors, like- emotion, stress, confidence.

Few studies have been done on Trust and Mistrust from a neuroscience perspective. A study performed power spectrum analysis with EEG data and observed frequency ranges. This study found that Trust evokes higher alpha and beta waves whereas mistrust evokes higher gamma waves. This can be interpreted as trust that can cause calmness and a relaxing state of the human brain and mistrust can cause stress and anxiety. This study also shows that there can be intraindividual differences in power spectrum density regarding trust and mistrust. A study has been conducted to observe different trust scenarios by games among human-human partners and human-robot partners, there was no overall difference in initial trust to human and robots; but their trust can vary depending upon their partners, observing their behavior.

Trust is an important factor for automation reliance decisions and affects human-automation performance. Trust also affects the human operators' adaptation to automation and self-confidence. In different automation system settings, the wrong trust may lead to different kinds of complications. Under-trust can cause demotivation to rely on the system while over-trust may cause users less cautious. While some robot or automation is involved, there should not be any presence of "trust betrayal", so humans sometimes choose to depend on machine decision over 'other human's' decision. And hence comes the question of over-trust. In a study on maritime navigation using autopilot, it has been found that human operators make more error when they trust the autopilot. Automated

vehicle drivers' behavior also to be found affected by trust. Neural studies have found that trust can be associated with many other factors including very individual factors like emotion, stress, anxiety feelings.

A study found that human trust decreases with the increasing rate of error, thus choosing manual control over automation. The findings are aligned with the above discussed findings, alpha and beta waves are associated with trust and active in the frontal lobe, where gamma wave is associated with mistrust and active in the temporal lobe.

3.3.8.5.20 Experiment II with Simulation For investigating human trust in automa- tion, the experiment II is designed to measure neurological activities of decision-making between automatic or manual control and to record the use of automatic control. This experiment is developed from a simulation model of previous study. We used an experimental system such as a semi-automatic pasteurization plant to investigate the change in an operatorÕs control strategies and trust during interaction with the automation. The operator had the two goals of performance and safety, and made decisions by using manual control, automatic control or a combination of the two. They evaluated trust based on data collected from subjective but specific questions about the operatorÕs trust in automation and the self-confidence in specific parts of automation. The study used a well-designed experiment of 60 trials with a training trial and the occurrence of fault that examined various aspects of human-machine trust that could influence the operatorÕs strategies.

3.3.8.5.21 Hypotheses This study investigates human trust through EEG recordings, identify and analyze specific brainwaves and brain regions, and explore how human trust affect human operatorÕs use of automation. Based on the findings of current literature and the rationale described, the followings hypotheses will be investigated:

- When a human operator trusts automation, alpha and beta waves and the frontal lobe will be detected, and human operatorÕs strategy the use of automation will increase.
- When a human operator mistrusts automation, gamma brainwaves and the temporal lobe will be detected, and human operatorÕs strategy on the use of automation will decrease.

3.3.8.5.22 Instrumentation The EEG data was recorded using the g.HIamp (256 multi-channel amplifier), g. GAMMAsys (active electrode system with g. GAMMAcap) and g. Recorder (brain signal recording software) by g. tec medical engineering company. Per the International 10-20 system of electrode placement, 20 electrodes (Fp1, Fp2, Fpz, F7, F3, Fz, F4, F8, T7, C3, Cz, C4, T8, P7, P3, Pz, P4, T6, O1, O2) were recorded and measured. The 10-20 system is based on the relationship between the location of an electrode and the underlying area of the cerebral cortex. The letters F, T, C, P, and O represent the frontal, temporal, central, parietal, and occipital lobes. Even numbers refer to the right hemisphere of the brain and odd numbers refer to the left hemisphere of the brain. The participantÕs head is fitted with a cap of electrodes (g. GAMMAcap), and the selected 20 electrodes are filled with abrasive electrolyte gel using a syringe. The sampling frequency to record brainwaves is 256 Hz, high-frequency filter as 60 Hz, and low-frequency filter as 0.1 Hz. The g.Recorder (brain signal recording software) also included artifact removing filters for eye blink, jaw clenching and muscle movement. Before an experiment, participantsÕ artifacts were recorded, marked and removed from raw data for the experiment.

3.3.8.5.23 EEG procedure At the beginning of the experiment, each participant reads informed consent agreement. Afterwards, the researcher addresses any questions and concerns. All participants are required to provide demographic information such as age, gender, and contact information as well as a signed and dated consent form. After each participant signs the consent form, the participant is asked to put aside all personal items on a secure table. The participant is asked to turn off their cell phone to avoid any interruptions during the experiment. After the participant is seated, detail instructions are given for each experiment. The facilitator in the room addresses any questions or concerns related to the experiments. Then the participant is fitted with a g. GAMMA cap with electrodes that measures small electrical signals produced by the brain in a completely non-invasive manner. The 20 electrodes, per the international 10-20 system for electrode placement is filled with a gel using a syringe with a plastic tip. Before each experiment begins, participants are asked to record their brainwaves for marking and removing artifacts such as eye blink, jaw clenching and muscle movement. Then, as baseline data without any stimuli, participantsÕ brainwaves are recorded when they are relaxed, but they are asked not to move, not to talk and see the blank monitor for a minute. When an experiment begins, the participant is asked to focus their attention on the computer screen and use the keyboard or mouse if needed. They are asked not to talk and blink their eyes (if possible) to reduce recording artifacts in the brainwaves. After the experiment, each participant gently removes the electrode cap and any remaining gel from their head or hair using the tissues and/or towel provided.

3.3.8.5.24 EEG Recording

The EEG data are recorded using the g.HIamp (256 multichannel amplifier), g. GAMMAsys (active electrode system with g. GAMMAcap) and g. Recorder (brain signal recording software) by g. tec medical engineering company. Per the international 10-20 system of electrode placement, 20 electrodes (Fp1, Fp2, Fpz, F7, F3, Fz, F4, F8, T7, C3, Cz, C4, T8, P7, P3, Pz, P4, T6, O1, O2) are recorded and measured (see Figure 163). The 10-20 system is based on the relationship between the location of an electrode and the underlying area of the cerebral cortex. The letters F, T, C, P, and O represent the frontal, temporal, central, parietal, and occipital lobes. Even numbers refer to the right hemisphere of the brain, and odd numbers refer to the left hemisphere of the brain. The participants head is fitted with a cap of electrodes (g. GAMMAcap) and the selected 20 electrodes are filled with abrasive electrolyte gel using a syringe. The sampling frequency to record brainwaves is 256 Hz, high frequency filter as 50 Hz and low frequency filter as -50 Hz.

3.3.8.5.25 Results

3.3.8.5.26 Rate of Automatic Control and Trust Level After 28 participants com- pleted 10 trials, they finished the survey to evaluate the level of trust between people and automa- tion with 7 scales. The level of trust can be defined using a subjective trust questionnaire with a trust scale. This survey follows a 7-point rating scale that ranged from '1' being 'not at all' (which is close to mistrust) to '7' being 'extremely'(which is close to trust). For trials 1 to 6, the level of trust increased from 57% to 94% whereas using the automatic control increased from 79% to 96% (see Figure 165. On the other hand, when an error of automatic control with 32% performance rate in trial 6 occurred, trial 7 showed a substantial decrease in the level of trust from 94% to 30%, and a decrease in the automatic control from 96% to 21% (see Figure 4). These results imply that when the trust level in automation increases, the use of automatic control will increase. Trust is

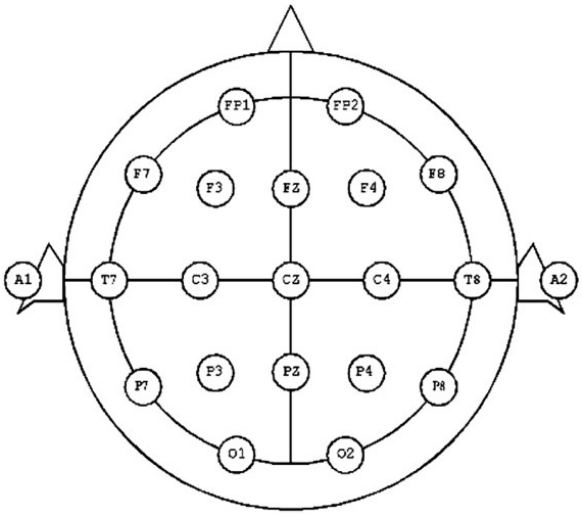
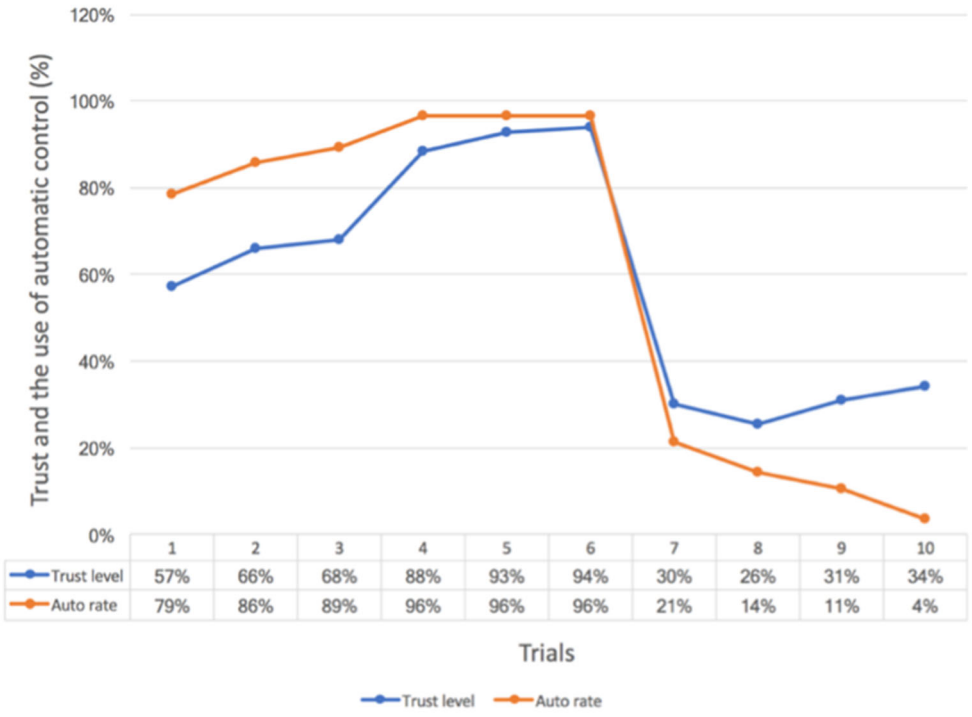


Figure 175: International 10-20 system for electrode placement

a multidimensional construct influenced by various factors, so it is important to find which trial evokes trust or mistrust through the level of trust. It is important to understand that previous performance affected the next decision. For example, the low performance (32%) of automation of trial 6 affected the decision of trial 7. Trial 5 shows 93% and trial 6 shows 94%, so they show a high level of trust. The brainwaves of trial 5 and 6 were analyzed for trust situations. Trial 7 shows 30% and trial 8 shows 26%, so they show a low level of trust, which is mistrust. The brainwaves of trial 7 and 8 were analyzed for mistrust situation.

3.3.8.5.27 Comparison of Intraindividual and Average Differences

The individual variability for the brainwaves is high because of individual characteristics, so the analysis of this paper focuses on the intraindividual differences, which shows the individual difference with baseline (standard) and two stimuli of trust and mistrust situation in this paper. This paper paid attention to the change in how trust and mistrust situations affect individual brainwaves. It didn't compare the female and male results yet, but it can be investigated in the future. In the trust situation, the power of alpha waves of 27 of 28 participants highly increased, and the power of 1 participant (P16) decreased, but in the mistrust situation, the power of the alpha waves of 26 from 28 participants slightly decreased, but the power of 2 participants (P5 and P24) increased (see a in Figure 166). In the trust situation, the power of beta waves for all 28 participants increased considerably, but in the mistrust situation, the power of the beta waves for 26 of 28 participants increased slightly, but the power of 2 participants (P3 and P16) decreased (see b in Figure 166). In the trust situation, the power of the gamma waves decreased slightly for 22 of 28 participants, but 5 participants (P6, P14, P16, P26, and P27) increased and 1 participant (P4) didn't change, but in the mistrust situation, the power of the gamma waves increased significantly for the 27 of 28 participants, but not for 1 participant (P23) (see c in Figure 166). There was 1 subject (P16) who chose more manual control than other subjects regardless of the performance of automation. In the trust situation, the alpha and beta wave of the subject (P16) is decreased and different than other



	1	2	3	4	5	6	7	8	9	10
Trust level	57%	66%	68%	88%	93%	94%	30%	26%	31%	34%
Auto rate	79%	86%	89%	96%	96%	96%	21%	14%	11%	4%

Figure 176: Comparison of the trust level and the use of an automatic control

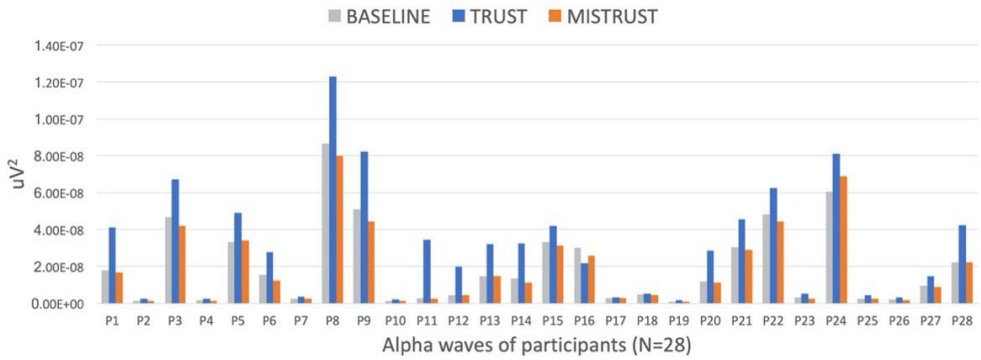
participants. In the mistrust situation, the gamma waves of the subject (P16) are increased and similar to other participants. Some people have over-confidence in themselves or strong mistrust of machines, so it needs to research further in the future.

3.3.8.5.28 Comparison of Trust and Distrust differences According to the results of intraindividual differences, the alpha and beta waves are associated with the trust situation, but the gamma waves are associated with the mistrust situation, so they need to analyze further using a standard deviation. By comparing the trust differences to the alpha waves when the baseline is 0, 96.42% of participants increases in the trust situation, and the standard deviation is 1.13E-08E (see in Figure 167). By comparing the trust differences to the beta waves when the baseline is 0, 100% of participants increases in the trust situation, and the standard deviation is 1.18E-08E (see b in Figure 167). By comparing the mistrust differences to the gamma waves when the baseline is 0, 96.42% of participants increases in the mistrust situation, and the standard deviation is 4.08E-10E (see c in Figure 167. According to the comparisons of trust and mistrust differences, the alpha and beta waves are increased in the trust situation, while the gamma waves are increased in the mistrust situation.

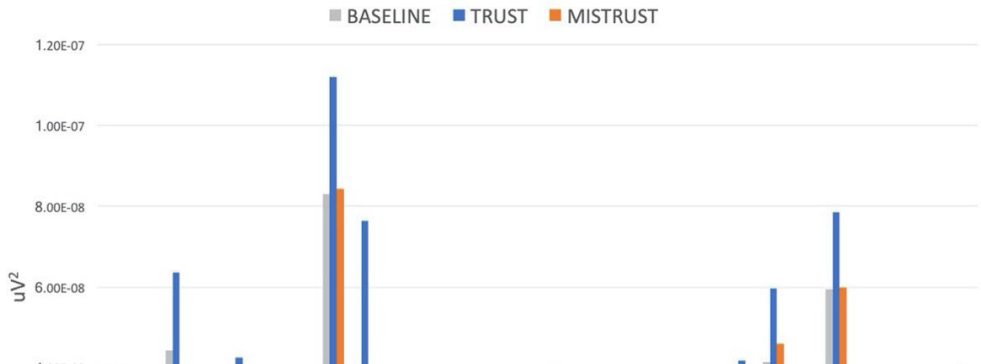
3.3.8.5.29 Conclusion This research investigated human trust in automation using EEG to identify specific brainwaves within the trust and mistrust situations. According to the results of the experiment, the power of alpha and beta waves was stronger for the trust situation; whereas, the power of gamma waves was stronger for the mistrust situation. This section investigates the neurological relationships that are grounded within the neuroscience literature, which is critical to understand the relation of alpha, beta, and gamma waves to trust. Alpha waves are related to meditation and reducing stress and anxiety. Beta waves are associated with conscious activities and can be a guideline for measuring the cognitive process. Beta activity is important because it can enhance concentration, attention, emotional stability, and energy levels. As the fastest brainwaves, Gamma waves can indicate anxiety, and advanced cognitive information, such as reasoning and judgment. By understanding the brain waves, alpha, and beta waves in trust situations were related to the normal cognitive process. However, gamma waves in mistrust situations are related to the complex cognitive process by increased stress and anxiety. Previous studies dealt with factors such as risk and feedback that affect human trust and decision-making using EEG and support the results of this research. Using EEG, Cohen found increased feedback processing affected lower the power delta and theta waves, and higher the power of alpha and beta waves through a competitive decision-making game. Findings from this research will contribute to utilizing a neurological technology to measure the human operator's level of trust which may affect decision- making and overall performances of automation in industries. This research can be valuable in designing automated systems to develop a user-friendly interface and effective training, which can increase the human operator's trust and decrease workload. Further, this research can be applied to monitor the psychological state of human operators in complex automation such as pilots operating automated aircraft or captains operating automated ships.

3.3.8.6 Technical Accomplishments in Task T3-8

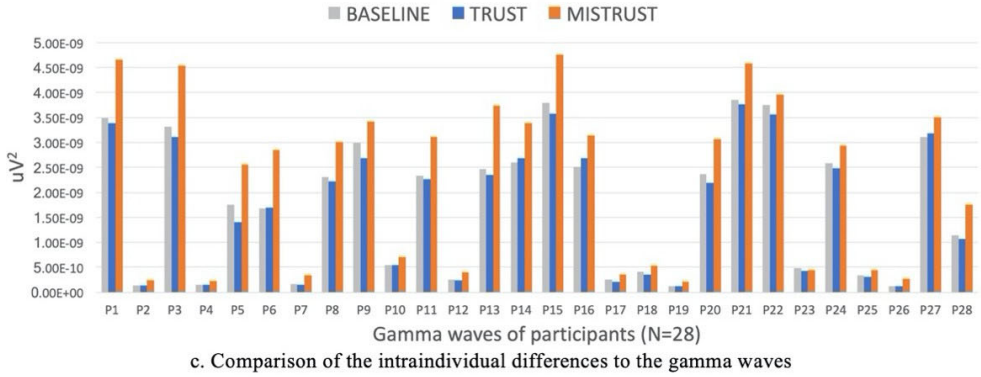
This task is finished.



a. Comparison of the intraindividual differences to the alpha waves



b. Comparison of the intraindividual differences to the beta waves



c. Comparison of the intraindividual differences to the gamma waves

Figure 177: Comparisons of the intraindividual differences to the alpha (a), beta (b), and gamma

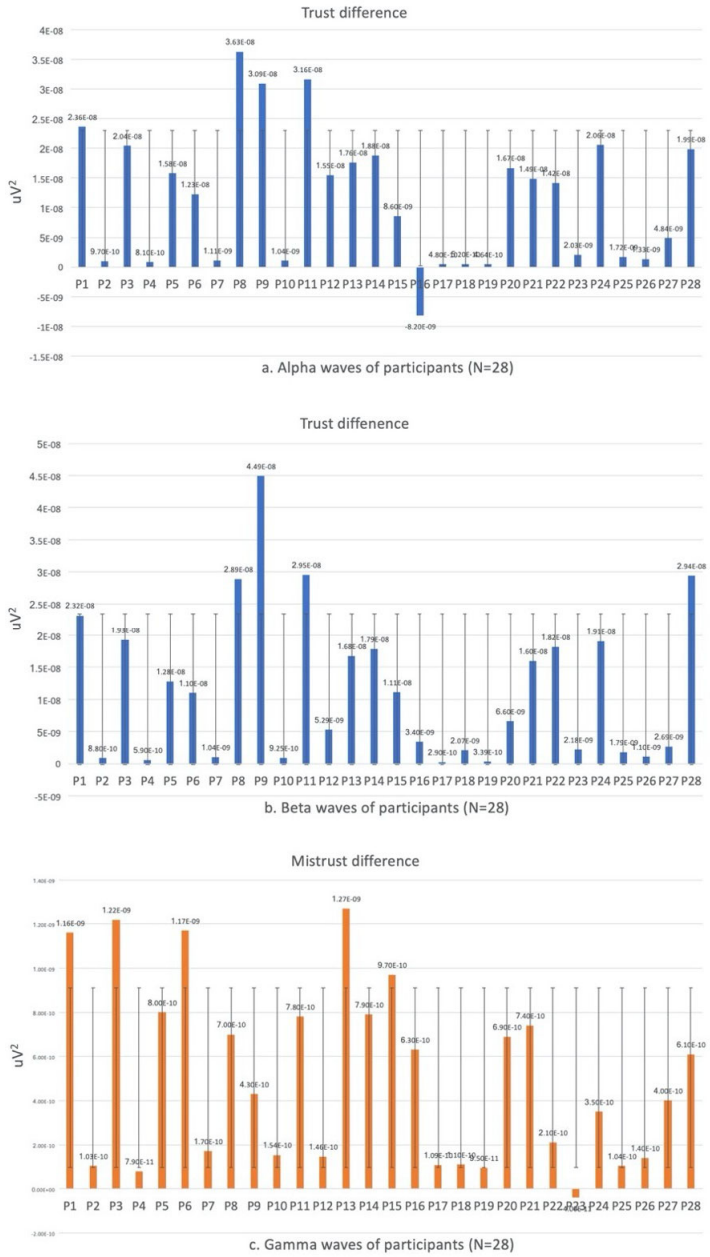


Figure 178: Comparisons of trust and mistrust differences to the alpha (a), beta (b), and gamma

3.3.9 Project Progress for Task T3-9 (Evaluation of visualization of uncertainty dynamically and intuitively)

3.3.9.1 Period of Performance under

Task T3-9 Start Date: April 2014

Conclusion Date: April 2021

Faculty lead: Younho Seong. Industrial and Systems Engineering

3.3.9.2 General Description of Task T3-9

In addition to error reduction and improvement of trust in the human-machine interaction, a further means for coping with uncertainty in semi-autonomous systems would be to enable human awareness of uncertainty itself. This task proposes to develop and test a prototype that represents uncertainty dynamically in order to support operators' judgment and sense-making. An experimental framework to examine the efficacy of the developed prototype for tactical visualization of uncertainty through many dimensions and morphing mechanisms will be provided to collect objective measures of performance. Studies into display factors influencing decision-making under uncertainty are planned. Specifically, indications from past research show that a more concrete representation may lead people to believe that information is more certain than it actually is, while a degraded or less concrete representation may result in an understanding as less certain. The proposed research will utilize a multi-dimensional framework to systematically vary display factors to simulate conditions of degraded information corresponding to uncertainty present. For example, the information could be represented through the use of numeric ranges or graphical areas rather than point estimates. Experiments will evaluate and compare how these various visualizations influence human judgments in operating simulated LSASVs.

3.3.9.3 Objectives for Task T3-9 over the Course of Project

There is a gap between the high-level references made to cognitive fit and the low-level ability to identify and measure it during human interaction with visualizations. We bridge this gap by using an electroencephalography metric derived from frontal midline theta power and parietal alpha power, known as the task load index, to determine if mental effort, measured at the level of cortical activity, is less when cognitive fit is present compared to when cognitive fit is not. We found that when there is cognitive fit between the type of problem to be solved and the information displayed by a system, the task load index is lower compared to when cognitive fit is not present. Furthermore, we support this finding with subjective (NASA task load index) and performance (reaction time and accuracy) measures. Our approach using electroencephalography provides supplemental information to self-report and performance measures. Findings from this study are important because they (1) provide more validity to the cognitive fit using a neurophysiological measure, and (2) use the task load index metric as a means to assess cognitive workload and effort in general.

3.3.9.4 Progress Against Planned Objectives in Task T3-9

3.3.9.5 Introduction

Technology allows information to be presented in any way for any task system, providing system designers with many choices for displaying information about the task system. However, task representation formats are seldom designed with a clear understanding of the cognitive activity they will induce in the human operator. Indeed, relationship between task representation formats, task types, and information processing is one of the least understood areas in research on how information visualization influence problem solving.

3.3.9.5.1 Cognition and Task Types Cognition refers to the mechanisms by which hu- mans organize and understand the world through perception of stimuli, and selection of an option from a set of available alternatives. Dual process theories can be used as organizing framework to explain human cognition. The theories posit that cognition can either be intuitive or analytical. Intuitive cognition is relatively fast and requires less cognitive effort, while analytical cognition is relatively slow and requires more cognitive effort.

According to Payne, "... decision-making ... is highly contingent on the demands of the task." For instance, a task that represents information in numerical or symbolic format and allows enough time for computation offers greater support for analytical cognition. Whereas, a task that presents the same information in graphical format and allows a short time for judgment offers greater support for intuitive cognition.

Task information may be represented in graphical or table format. Graphs are spatial problem representations that facilitate viewing presented information without attending to elements analyt- ically. They accentuate relationships in the data. Tables, on the other hand, are symbolic problem representations that facilitate analytical processing of presented information. They do not represent data relationships openly. However, they facilitate extracting data values.

3.3.9.5.2 Cognitive Fit Theory The cognitive fit theory (CFT) is a suitable theoretical framework for understanding the relationship between visualizations and task types. It was devel- oped to explicate how the fit between information presentation format and decision-making task can influence the human operator's problem-solving performance. The CFT (see Figure) suggests that when there is a match between information presentation format (visualization) and task type, human operators can form the right mental representations and use cognitive strategies to fit the presented data. This fit minimizes their cognitive effort leading to better decision–making perfor- mance.

In CFT, problem representation and problem-solving task are viewed independently. Task types are classified as either symbolic or spatial. On the one hand, symbolic tasks involve "extracting discrete and therefore, precise, data values" and "are best accomplished by analytical processes". Symbolic tasks are akin to analysis-inducing tasks. On the other hand, in spatial tasks decision makers evaluate "the problem as a whole...and therefore requires making associations or perceiving relationships in the data". Spatial tasks are akin to intuition-inducing tasks.

There is cognitive fit when graphical representations support spatial (intuition-inducing) tasks, and when numerical (or table) representations support symbolic (analysis-inducing) tasks. CFT posits that when there is a cognitive fit, decision makers exert minimal cognitive effort.

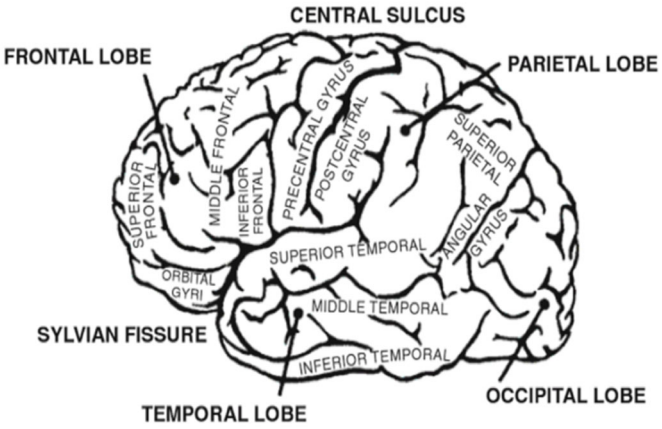


Figure 179: Cogntive Fit Theory

It was considered clinical decision making as a set of problem-solving activities in which (1) subjective responses from a patient and objective data from the patient encounter represent the problem and (2) assessment of a patient's condition based on consolidated subjective and objective information, and treatment planning represent the task. They posited that clinical decision support systems (CDSSs) that align with the problem representation and task representation in order to match clinicians' mental schemas should lead to reduced cognitive effort and enhanced performance. Based on this, they developed a stroke CDSS. It was reported that physicians performed better and reported lower perceived cognitive effort when they used the developed stroke CDSS compared to a baseline stroke CDSS. Also it was investigated cognitive fit between quality control tasks and how these tasks are visually represented. They found that specific information visualizations can improve both accuracy of detection and response time of identifying out-of-control conditions. Their results suggest that CFT can be used to evaluate the effectiveness of information visualization technologies. Another claim compared relative effectiveness of different social network representations for aiding human problem–solving. They found that participants performed best when symbolic problem–solving tasks were represented in table format. Likewise, they found that participants performed best when spatial problem–solving tasks were represented in graphic (node–link diagrams) format. Their results suggest that problem–solving is enhanced when there is a match between task type and network representation.

3.3.9.5.3 Cognitive Effort Cognitive effort is the amount of cognitive resources –including perception, memory, and judgment– required to cope with the demands of a task or complete a task. It is a subjective, psychological construct. Self-reports, psychological or physiological measures can be used as proxies of cognitive effort.

3.3.9.5.4 Subjective measures of cognitive effort Previous CFT research has used a variety of subjective measures as proxies for cognitive effort. For example, it was employed the NASA Task Load Index (NASA-TLX) as a measure of perceived cognitive effort. The NASA-TLX is considered one of the most widely used, extensively validated and effective measures of perceived mental workload. The NASA TLX instrument can be administered using a paper and pencil version

or by a computer-based application during or following an experimental trial.

3.3.9.5.5 Physiological measures of cognitive effort Traditional means of assessing cognitive effort are intrusive and do not produce enough information on the cognitive state of the human operator. Subjective measures are prone to bias from tasks that have been previously completed. That is, cognitive workload ratings, in particular, have been shown to be influenced by how demanding a preceding task was. Moreover, task performance measures are limited since they are overt and neglect the underlying biological content of behavior. These measures are unable to explain the variability of human performance under conditions of stress and high workload. The same level of performance may be accompanied by multiple degrees of cognitive workload. Indeed, human performance should be analyzed as a summation of task performance and physiological cost of maintaining that performance. Physiological measures can be used as proxies of cognitive effort. They include pupil dilation, heart rate and brain activity.

Neuroergonomics, defined as the study of brain and behavior, allows researchers to study and develop new frameworks about humans and work than an approach based solely on the measurement of subjective perceptions or overt performance of human operators. Functional brain imaging techniques employed in neuroergonomics enable the monitoring and/or capturing of brain activities while participants perform various perceptual, motor, and/or cognitive tasks. They include electroencephalography (EEG), functional near-infrared spectroscopy (fNIRS), and functional magnetic resonance imaging (fMRI). Despite its relatively low spatial resolution, EEG's high temporal resolution makes it very suitable for assessing cognitive states.

3.3.9.5.6 EEG Spectral (frequency) analysis of EEG data is commonly used to understand neural mechanisms underlying perception and decision making. EEG frequency bands include delta (less than 4Hz), theta (4-8Hz), alpha (8-12Hz), beta (12-30Hz), and gamma (30-80Hz). Several researchers have demonstrated that EEG can be used as an unobtrusive measure of cognitive effort. Highly related to the present study compared the effectiveness of different visualizations of box plots in terms of the cognitive effort required by participants to interpret them. They focused on EEG theta power and alpha power derived from electrodes placed around the prefrontal cortex. The study found simultaneous alpha power decreases and theta power increases were associated with visualizations that were difficult to interpret. Their findings are consistent with prior studies which found increased cognitive effort to be associated with increased frontal-midline theta band activity and decreased parietal-occipital alpha band activity. Task load index (TLI), a composite metric, is defined as the ratio of theta band activity at frontal midline sites to alpha band activity at parietal sites - ratio between Frontal theta power and Parietal Alpha power. TLI increases with increasing cognitive effort and can be used to measure cognitive effort.

3.3.9.5.7 Aim of study Neuroergonomics holds promise for measuring cognitive effort in visualization and CFT research. We did not find the use of neurophysiological measures to estimate cognitive effort in the published CFT literature. Indeed, the traditional approach of evaluating the best visualization from a set of visualizations for a particular data set is not enough as it discounts the neurophysiological cost of interpreting visualizations. This study evaluates visualizations by using EEG to measure the cognitive effort exerted by participants as they interact with visualizations. Particularly, we use subjective (NASA TLX) and objective (TLI) measures to investigate the claim

that congruence between visualization format and nature of a task minimizes cognitive effort and leads to better performance.

3.3.9.5.8 Hypotheses Representing an inherently spatial (intuition-inducing) task in two presentation formats-graphical and numerical (see Figure 2), we hypothesize that when there is a match between task type and task representation:

Hypothesis 1: Perceived cognitive effort, measured by NASA TLX, will be lower,

Hypothesis 2: Cognitive effort, measured by TLI, will be lower,

Hypothesis 3: Accuracy will be higher; response time will be faster.

Task Type	Presentation Format	
	Graphical Display	Numerical Display
Spatial Task	1.Cognitive fit results in less cognitive effort 2.Better performance than when there is no cognitive fit	1. No cognitive fit results in more cognitive effort 2.Lower performance than when there is cognitive fit

Figure 180: Hypothese development

3.3.9.6 Methods

3.3.9.6.1 Description of Experimental Task Online water quality monitoring enables drinking water utilities to detect, in real-time, changes in water quality conditions in their distribution systems. Water quality parameters that can be monitored include chlorine, fluoride, pH, total dissolved solids (TDS), nitrate, and turbidity. The US Environmental Protection Agency has water quality standards that must be met by these drinking water utilities. Some contaminants and their maximum contaminant levels are shown in Figure 159. Human operators visualizing the data must ensure no contaminant exceeds its maximum contaminant level (MCL).

Contaminant	Maximum Contaminant Level
Chloride	250 mg/L
Fluoride	2.0 mg/L
pH	6.5–8.5
Total Dissolved Solids	500 mg/L
Nitrate	10 mg/L
Turbidity	0.1–1 NTU

Figure 181: Contaminants and the maximum contaminant levels

In the present study we simulated a fictional company, XYZ Aqua Inc, with an automated sampling station. The station contained 6 different sensors that measured 6 different water qual- ity parameters. We randomly generated parameter values in Python version 2.7 using the trun- cated normal distribution function, a normal distribution with finite interval, implemented as the scipy.stats.truncnorm module. Figure 4 shows distribution characteristics of 120 generated parameter values.

3.3.9.6.2 Participants Fifteen right-handed participants (25.9 +/-1.9 years, 10 men) from a university in the south east of the United States participated in the study. Each participant read

Parameter	Mean	Standard Deviation	Minimum	Maximum	Range
Chlorine	168.25	92.73	9.00	344.50	335.50
Fluoride	1.55	0.45	0.60	2.80	2.20
pH	7.40	0.79	6.40	9.70	3.30
Total Dissolved Solids	292.39	79.83	129.70	506.60	376.90
Nitrate	4.67	2.66	1.00	12.30	11.30
Turbidity	1.19	0.60	0.10	2.70	2.60

Figure 182: Distribution characteristics of generated parameter values (n = 120).

and signed the informed consent form. Participants had no history of neuropsychiatric disorders and were not taking psychoactive medications.

3.3.9.6.3 Experimental Design We designed a within-subject experiment in which participants were subjected to two conditions – graphical (intuition-inducing) and numerical (analysis-inducing). Participants had to decide if water quality parameters were within acceptable limits using either a graphical display or numerical format. We presumed cognitive fit would occur with this inherently spatial task using graphical display because it allowed participants to make infer- ences about relationships within data, make comparisons, and detect patterns or deviations from patterns. Figure 5 below shows 2 stimuli (1 for each condition) representing 6 separate simulated sensor readings for water samples collected at time intervals. Each condition had 4 blocks, with each block containing 30 trials making it 120 trials per condition. A trial, in each condition, begun with a black screen for 1s, followed by presentation of the stimuli for 60s. The trial ended when participants clicked the mouse button before or at the stipulated duration of 60s. A cross fixation of variable duration (2 4s) was presented between trials. (see Figure 163.) We inserted markers to record the onset of each trial, the presentation of each visualization, and the response of the partici- pants. We instructed participants to press the left mouse button to indicate bad water quality when any of the parameters exceeded its MCL. Otherwise, they were to press the right mouse button to indicate good water quality.

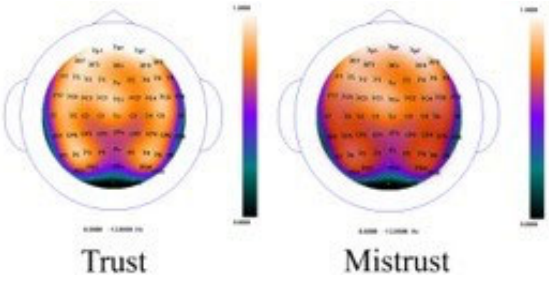


Figure 183: Stimuli presented in graphical format

3.3.9.6.4 EEG We fitted participants with an EEG cap, g.GAMMAcap (g.tec, Austria), which uses the International 10-20 system for EEG electrode placement locations, and ensured that the cap was correctly positioned with Cz showing the same distance to the naison and to the inion. All channels used the right ear lobe as common reference (Ref), and AFz as ground (Gnd).

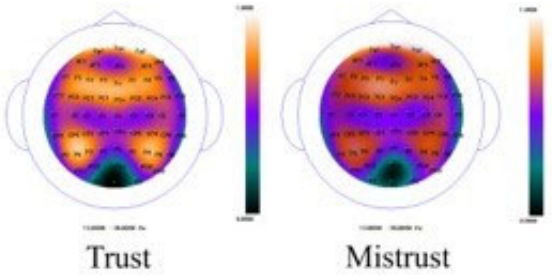


Figure 184: Stimuli presented in numerical format.

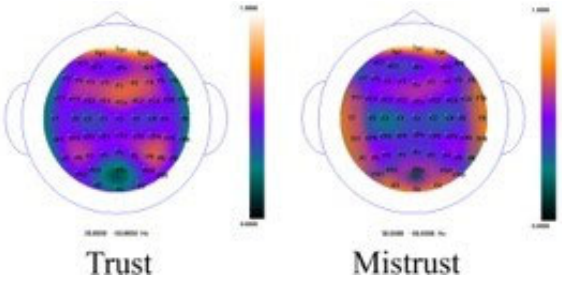


Figure 185: Study block design. Conditions were randomized. Depending on the condition, stimuli were presented in graphical or numerical format.

Before recording participants' EEG, we ensured that electrode impedances were less than 10k . Electrodes were connected and amplified with a g.HIamp amplifier. We recorded continuous raw EEG data (see Figure 165) at a sampling rate of 512 Hz using a Butterworth filter (0.01Hz high pass -100Hz low pass). We used a notch filter with 60 Hz cutoff frequency to remove line noise. We stored data to disk and analyzed it offline.

3.3.9.6.5 EEG Processing We imported the continuous raw EEG data into EEGLAB. We downsampled the data from 512 Hz to 256 Hz to compress its size and reduce computational requirements. We used Artifact Subspace Reconstruction and implemented in EEGLAB, to automatically remove artifacts including eye blinks.

In this study, we time-locked EEG epochs to stimuli presentation. Epochs started 1s after a trial (see Figure 163). Because of the variable response time for each trial, we set a 1s window around participant response to mark the end of an epoch. Thus, an epoch started 1s after a trial and ended when the participant clicked the mouse button. We performed all EEG data analysis offline with MATLAB, EEGLAB, and custom code. We used the nonparametric Welch method implemented in MATLAB as pwelch to estimate the power spectrum in the frequency bands of interest. To minimize spectral leakage and smearing, we used a Hamming window to segment preprocessed EEG data into 1s windows with a 50% overlap of the previous segment and 50 % of the next. We calculated log spectral power in each window for theta (4-8Hz) and alpha band (8-12Hz) and mean over all data segments for each participant for the two experimental conditions. Finally, for each experimental condition, we computed each participant's TLI as ratio of mean theta power at Fz to mean theta power at Pz.

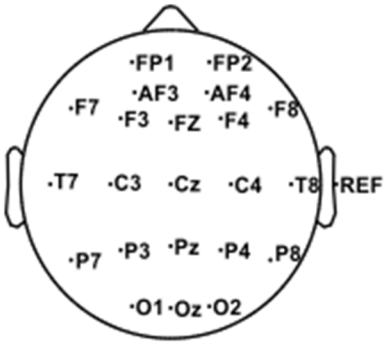


Figure 186: 22-channel electrode montage

3.3.9.6.6 Procedure After completing an informed consent form and a demographic questionnaire, participants were given a brief introduction to the tasks. Participants sat on a chair approximately 56 cm from a 21 inch LCD monitor with a resolution of 1024 x 768. Stimuli were presented using the stimulus delivery software, Presentation® software (Version 18.0, Neurobehavioral Systems, Inc., Berkeley, CA) at an approximate size of 15 x 10.5 of visual angle. Each stimulus contained sensor readings and corresponding MCLs for 6 water quality parameters. We used outcome feedback to familiarize participants with tasks before the main experiment. The familiarization session consisted of 30 trials per condition. We did not record participants' EEG during this session. After the familiarization session, we fitted participants with an EEG cap (see Figure 8) and recorded EEG data as they carried out the main experiment. We did not use outcome feedback in the main experiment. Participants carried out the experiment as show in Figure 6 above. To avoid order effects, we presented the block of trials, and the trials in each block in pseudorandom order. We recorded participants' accuracy and response times. We administered the NASA-TLX questionnaire at the end of the last block for each condition. Finally, we debriefed and thanked participants.

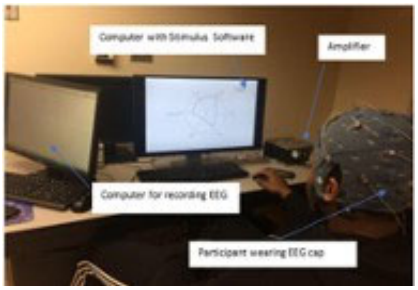


Figure 187: Participant undertaking experiment

3.3.9.6.7 Data analysis We conducted all statistical analyses in R version 3.4.1. Before testing each hypothesis, we tested for normality using Shapiro-Wilk test. Where normality was not violated, we used two-tailed paired t-tests to test for significant differences between conditions. Otherwise, we used two-tailed Wilcoxon signed-rank tests. We used p-value of 0.05 as criterion for statistical significance. We use Hedge's grm to estimate effect sizes for differences between

dependent means.

3.3.9.7 Results

3.3.9.7.1 Hypothesis 1 First, we considered each subscale of the NASA TLX questionnaire. Results (see Fig. 9) showed the conditions significantly differed with respect to EF ($t(14) = 3.84$, $p = .02$, grm = 0.86), FR ($t(14) = 3.20$, $p < .001$, grm = 0.77), MD ($t(14) = 4.67$, $p < .001$, grm = 1.72), PE ($t(14) = 2.47$, $p = 0.03$, grm = 0.63), and TD ($t(14) = 5.78$, $p < .001$, grm = 1.19). It is worth noting that scores for the PE subscale are low for ratings of high performance. There was no significant difference between the two conditions in terms of physical demand (PD); $t(14) = 0.21$, $p = .83$, grm = 0.31.

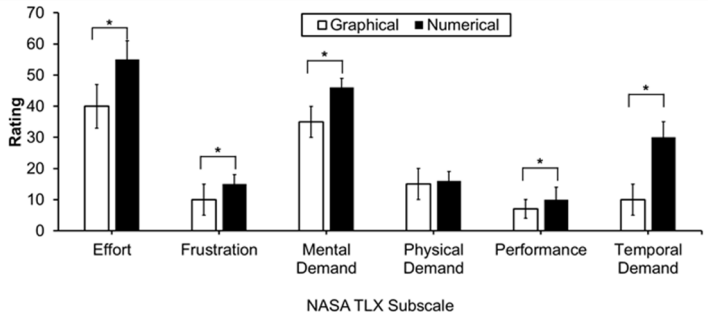


Figure 188: Subjective ratings by NASA TLX subscale for the graphical and numerical conditions Error bars are standard errors. The asterisk indicates a significant difference (p 0.05).

Second, we determined the global workload rating in each condition. Using the unweighted scoring procedure, we determined workload scores on each of the six subscales. A paired t-test revealed that mean perceived workload of the graphical format condition (M = 19.73, SD = 8.75) was significantly lower than that perceived of the numerical format condition (M = 26.27, SD = 6.53) ; $t(14) = 8.32$, $p = .02$, grm = 1.79.

3.3.9.7.2 Hypothesis 2 Figure 167 presents data for each participant and all experimental conditions, including the mean theta power at Fz and mean alpha power at Pz, as well resulting TLI values. For all participants mean theta power was higher in the numerical condition than in the graphical condition. Conversely, mean alpha power was higher in the graphical condition than in the numerical condition.

Figure 168 shows mean TLI in each task condition for each participant. For all 15 participants, task load indices were substantially higher in the numerical condition than in the graphical condition. TLI of the numerical condition (M = 0.49, SD = 0.13) was significantly greater than TLI of the graphical condition (M = 0.35, SD = 0.16); $t(14) = 6.43$, $p = .03$, grm = 0.86.

3.3.9.7.3 Hypothesis 3 Accuracy of the graphical condition (M = 94.72, SD = 4.36) was significantly higher than accuracy of the numerical condition (M = 91.28, SD = 6.68); $t(14) = 3.37$, p <02, grm = 0.88. Response time was significantly higher in the numerical condition (M = 44.60, SD = 11.62) than in the graphical condition (M = 13.72, SD =11.62); $t(14) = 11.75$, p .00<1, grm = 2.89.

Participant	Condition					
	Graphical			Numerical		
	Mean theta power at F_z (μV^2)	Mean alpha power at P_z (μV^2)	Mean TLI	Mean theta power at F_z (μV^2)	Mean alpha power at P_z (μV^2)	Mean TLI
1	38.12	26.04	1.46	46.49	12.20	3.81
2	41.36	19.37	2.14	41.37	11.92	3.47
3	56.31	10.16	5.54	58.13	10.05	5.78
4	33.35	13.76	2.42	34.32	9.43	3.64
5	39.49	14.46	2.73	46.20	9.32	4.95
6	42.40	13.64	3.11	51.57	10.94	4.71
7	47.62	18.94	2.52	49.15	14.09	3.49
8	56.33	29.13	1.93	59.76	21.27	2.81
9	51.59	11.04	4.67	47.52	7.10	6.69
10	54.18	9.03	5.60	55.15	8.95	6.16
11	35.09	10.14	3.46	48.14	9.27	5.19
12	49.70	12.96	3.84	54.86	11.53	4.76
13	33.81	14.43	2.34	53.86	9.83	5.48
14	43.21	10.10	4.29	46.60	8.96	5.20
15	52.11	8.13	6.41	53.51	7.52	7.12

Figure 189: Mean theta, mean alpha power and mean TLI for each participant and all experimental conditions.

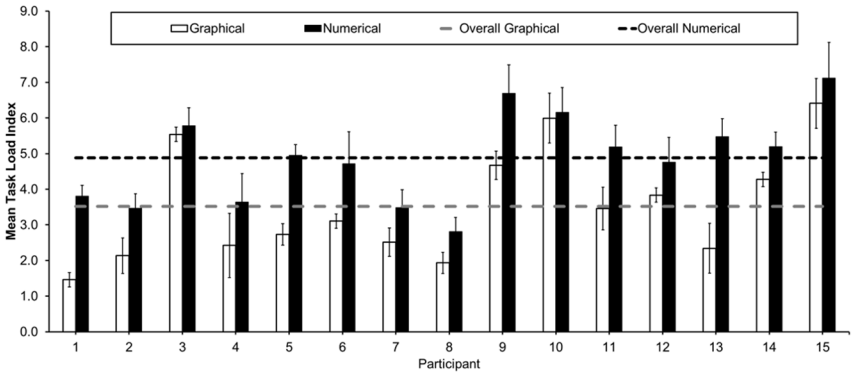


Figure 190: Mean EEG TLI in each task condition for each participant, and overall graphical vs. numerical group means. Error bars are standard errors.

3.3.9.8 Discussion

In the present study, we adopted a neuroergonomic approach to evaluate the effectiveness of information visualizations using the CFT. This approach afforded us the opportunity to employ neurophysiological measures to validate the claim that cognitive fit leads to minimal cognitive effort and enhanced performance. We found that when there is a match between task type and task representation, perceived cognitive effort, measured by NASA TLX, is lower; cognitive effort, measured by TLI, is lower; accuracy is higher; response time is lower. Furthermore, we found that found that the TLI can be used to evaluate visualizations in the context of the CFT, suggesting that the it can complement the strengths of behavioral and subjective methods used in traditional CFT research. Previous CFT research has used subjective methods to estimate cognitive effort, and accuracy and reaction time to determine decision making performance. Though traditional subjective methods for estimating cognitive effort, including NASA TLX, are applicable to general user studies, they are mostly administered post-task, making them susceptible to recall bias. Furthermore, because

these measures only provide information following task completion, there is no way to assess cognitive effort during actual task performance. It is worth noting that traditional behavioral measures disregard underlying biological content of behavior and therefore are unable to explain the burden that different visualizations of the same data place on participants' cognitive resources. By using neurophysiological measures, as adopted in the present study, the challenges listed above may be overcome. Variations in task demand and/or task difficulty have been shown to lead to changes in the spectral composition of EEG. Increasing task demand elicits lower parietal alpha power and higher frontal theta power. TLI, a composite measure of parietal alpha power and frontal theta power, is known to be very good at discriminating different cognitive task demand levels, with higher TLI values indicating higher cognitive effort. The two conditions in the present study are akin to a task at two different task demand levels. We used a radar chart to represent the task in the graphical condition because it allows for the viewing of high dimensional quantitative data, is more compact and helps visualize a norm violation. Considering the required cognitive effort in the graphical condition as different from that in the numerical condition, our results are consistent with our previous findings, which found mental effort measured using TLI higher for analysis-inducing tasks higher than for intuition-inducing tasks, which found increased TLI with increased task diffi- culty, and found simultaneous theta increase and alpha decrease with increased task difficulty. We found that mean EEG task load indices across participants increased with increasing task demand, suggesting that cognitive effort is significantly lower when there is a fit between task type and task representation.

The mixed methods approach adopted in the present study provided us an opportunity to validate the TLI using a subjective measure, the NASA-TLX. Previous studies have used the NASA-TLX to validate EEG measures. Recall that cognitive effort is low when cognitive fit is present. In the present study, we found significant differences between the two conditions with respect to mental demand and temporal demand (see Figure 168). These findings suggest that participants perceived the numerical condition to be more demanding in terms of mental activity and time required. Furthermore, nonsignificant difference between the two conditions regarding physical demand suggests that participants perceived the two presentation formats as having the same physical demand. We found that perceived cognitive effort measured in this study by the global score of NASA TLX questionnaire was significantly higher when there was a mismatch between task type and task representation. Cognitive fit is associated with enhanced decision-making performance. A review of previous literature revealed that decision-making performance is mostly measured by reaction time (or task completion time) and accuracy. In the present study reaction time, the per- formance measure with the highest effect size, was used to estimate how long it took participants to process presented stimuli. It took longer time for participants to initiate motor responses for stimuli in the numerical condition than in the graphical condition, suggesting that stimulus that must be recognized requires longer time to initiate motor response than stimulus that must be detected. Additionally, accuracy was higher in the graphical condition than in the numerical condition. We found that cognitive fit between task presentation and task type leads to better decision-making performance.

3.3.9.8.1 Practical Implications There is a gap between the high-level references made to cognitive fit and the low-level ability to identify and measure it during human interaction with visualizations. This study bridges that gap. Findings from this study are important to human factors and ergonomics professionals because they (1) provide more validity to the CFT using a

physiological measure, and (2) use the TLI ratio as a means to assess cognitive workload and effort in general. Given that visualizations mediate human system interactions in many operational environments and that cognitive overload is a source of performance errors during these interactions, system developers should examine and minimize the overall cognitive load associated with visualizations. The neuroergonomic approach adopted in this study can be applied to the evaluation of different visualizations. By combining a physiological measure, TLI, with traditional behavioral and subjective measures, system designers can determine when there is a cognitive fit between the type of problem to be solved and the information displayed by a system.

3.3.9.8.2 Study Limitations and Future Work This study has some limitations. First, while the study sample size was small, this was a completely within-subject design that exhibited medium to big effect sizes. Future work that considers a larger sample size is warranted. Second, although the task in this study might not generalize well to other operational settings, the TLI metric employed in the study does generalize across multiple domains. Third, EEG is difficult to apply in practice, considering the setup time, electrodes with gel, etc. Despite these limitations, EEG has an added value when combined with standard self-reports and performance measures. Future work will explore other neuroimaging techniques like fNIRS. In contrast to EEG, fNIRS systems noninvasively measure changes in cortical deoxygenated and oxygenated hemoglobin concentrations and are relatively resistance to motion artifacts. Finally, integrating EEG and fNIRS systems provide better insight into mental workload than either system alone.

3.3.9.9 Conclusion

To the best of our knowledge this is the first study to use both subjective (NASA TLX) and objective (TLI) measures to investigate the claim that congruence be-tween visualization format and nature of the task minimizes cognitive effort and leads to better performance. Our results suggest that visualization for an inherently spatial (intuitive) task reduces cognitive effort and enhances performance when it is in spatial or graphical format rather than when it is in symbolic or numerical format. We found TLI to be sensitive to task demand changes elicited by different visualizations, demonstrating its utility in visualization evaluation. Furthermore, our results demonstrate that TLI could complement the strengths of behavioral and subjective methods used in traditional CFT research.

3.3.9.10 Technical Accomplishments in Task T3-9

We are able to establish concrete research and it associated activities in neuroergonomics field. This is the only laboratory in HBCUs in the nation.

4 Deliverables

4.1 Publications

32 journal and 103 conference papers, have 2 patents and 4 under consideration and 4 book chapters were published:

- **Title:** Impact of Time Delays on Networked Control of Autonomous Systems

Author(s): Kolar, Prasanna, Nicholas Gamez, and Mo Jamshidi

Publication date: Feb 29, 2020

Publication venue: Beyond Traditional Probabilistic Data Processing Techniques: Interval, Fuzzy etc. Methods and Their Applications. Springer, Cham

Most Relevant Task: Task T1-1

Other Relevant Tasks: Task T1-3

- **Title:** Development of robot operating system (ROS) compatible open source quadcopter flight controller and interface

  Author(s): Abhijit Majumdar, Nicholas Gamez, Patrick Benavidez, Mo Jamshidi

  Publication date: June 18, 2017

  Publication venue: 2017 12th System of Systems Engineering Conference (SoSE)

  Publication keywords: Quadcopter, Robot Operating System, Unmanned Aerial Vehicle, Flight controller, interface

  Most Relevant Task: Task T1-1

  Other Relevant Tasks: Task T1-3

- **Title:** Utilizing Transfer Learning and Learning without Forgetting Methods to Train A 3D Object Detection System for Autonomous Vehicles Based on LiDAR, RADAR, And Image Data

  Author(s): Ramin Sahba, Amin Sahba, Mo Jamshidi, Paul Rad

  Publication date: *under review*

  Publication venue: World Automation Congress 2021

  Publication keywords: Transfer Learning, Learning without Forgetting, 3D Object Detection, Autonomous Vehicles, LiDAR, RADAR

  Most Relevant Task: Task T1-2

  Other Relevant Tasks: NA

- **Title:** Optimized IoT Based Decision Making for Autonomous Vehicles in Intersections

  Author(s): Amin Sahba, Ramin Sahba, Paul Rad, Mo Jamshidi

  Publication date: October 2019

  Publication venue: 2019 IEEE 10th Annual Ubiquitous Computing, Electronics & Mobile Communication Conference

  Publication keywords: IoT, Autonomous Vehicles, Decision Making, Intersection Management, Optimization, Traffic Network

  Most Relevant Task: Task T1-2

  Other Relevant Tasks: NA tasks.

- **Title:** 3D Object Detection Based on LiDAR Data

  Author(s): Ramin Sahba, Amin Sahba, Mo Jamshidi, Paul Rad

Publication date: October 2019

Publication venue: 2019 IEEE 10th Annual Ubiquitous Computing, Electronics & Mobile Communication Conference

Publication keywords: 3D Object Detection, Encoder, Lidar, Dataset, Point Cloud

Most Relevant Task: Task T1-2

Other Relevant Tasks: NA tasks.

- **Title:** Formation Control Implementation Using KobukiTurtleBots and Parrot Bebop Drone

  Author(s): Nicolas Gallardo, Karthik Pai, Berat A. Erol, Patrick Benavidez, Benjamin Champion, Nicholas Gamez and Mo Jamshidi

  Publication date: 06 October 2016

  Publication venue: 2016 World Automation Congress (WAC)

  Publication keywords: formation control, virtual leader, robot, drone,UGV, UAV, turtle-bot, parrot bebop

  Most Relevant Task: Task T1-3

  Other Relevant Tasks: NA

- **Title:** Autonomous Decision Making for a Driver-less Car

  Author(s): Nicolas Gallardo, Nicholas Gamez, Paul Rad, and Mo Jamshidi

  Publication date: 31 July 2017

  Publication venue: 2017 12th System of Systems Engineering Conference (SoSE)

  Publication keywords: Autonomous Driving, Deep Learning, MachineLearning, Neural-Network, TensorFlow, Driver-less Cars

  Most Relevant Task: Task T1-3

  Other Relevant Tasks: NA

- **Title:** Time Delay Based Dynamic System of Networked Autonomous Vehicles

  Author(s): Nicholas Gamez, Prasanna Kolar, Ahmad Taha and Mo Jamshidi

  Publication date: 31 July 2017

  Publication venue: 2017 12th System of Systems Engineering Conference (SoSE)

  Publication keywords: NA

  Most Relevant Task: Task T1-3

  Other Relevant Tasks: NA

- **Title:** Stereo Camera Based Formation Control for Unmanned Aerial Vehicles

  Author(s): Jonathan Lwowski, Abhijit Majumdar, Patrick Benavidez, John **J.** Prevost, Mo Jamshidi

  Publication date: 09 August 2018

  Publication venue: 2018 World Automation Congress (WAC)

Publication keywords: Formation control, Leader-follower, control, stereo camera, UAV

Most Relevant Task: Task T1-3

Other Relevant Tasks: NA

- **Title:** A New Combination Rule Based on the AverageBelief Function

  Author(s): Gabriel Awogbami, Norbert Agana, Shabnam Nazmi, Abdollah Homaifar

  Publication date: 04 October 2018

  Publication venue: SoutheastCon 2018

  Publication keywords: Uncertainty, Dempster Shafer rule of combination, belief function.

  Most Relevant Task: Task T1-3

  Other Relevant Tasks: NA

- **Title:** A Symbolic Approach for Multi-target Dynamic Reach-avoid Problem

  Author(s): Laya Shamgah, Tadewos G. Tadewos, Ali Karimoddini, Abdollah Homaifar

  Publication date: 23 August 2018

  Publication venue: 2018 IEEE 14th International Conference on Control and Automation (ICCA)

  Publication keywords: reach-avoid, symbolic planning, motion planning, hybrid control, reactive synthesis, multi-target

  Most Relevant Task: Task T1-3

  Other Relevant Tasks: NA

- **Title:** Multi-Attribute Decision Fusion for Pattern Classification

  Author(s): Gabriel Awogbami, Norbert Agana, Abdollah Homaifar

  Publication date: 04 October 2018

  Publication venue: SoutheastCon 2018

  Publication keywords: Multi-attribute decision fusion, multi-criteria decision analysis, pattern classification.

  Most Relevant Task: Task T1-3

  Other Relevant Tasks: NA

- **Title:** Implementation of Reinforcement Learning Simulated Model on Physical UGV Using Robot Operating System for Continual Learning

  Author(s): Edgar M. Perez, Abhijit Majumdar, Patrick Benavidez, Mo Jamshidi

  Publication date: 4 July 2019

  Publication venue: 2019 14th Annual Conference System of Systems Engineering (SoSE)

  Publication keywords: Unmanned ground Vehicle, Reinforced Learning, Robotic Operating System (ROS), Continual Learning

  Most Relevant Task: Task T1-3

Other Relevant Tasks: NA

- **Title:** Accuracy of the HTC Vive Tracker for Indoor Localization

  Author(s): Jonathan Lwowski, Abhijit Majumdat, Patrick Benavidez, John J. Prevost, Mo Jamshidi

  Publication date: 21 October 2020

  Publication venue: IEEE Systems, Man, and Cybernetics Magazine (Volume: 6, Issue: 4, Oct. 2020)

  Publication keywords: Localization, formation control, infrared, sensor calibration, virtual reality hardware

  Most Relevant Task: Task T1-3

  Other Relevant Tasks: NA

- **Title:** Virtual Testing and Policy Deployment Framework for Autonomous Navigation of an Unmanned Ground Vehicle Using Reinforcement Learning

  Author(s): Tyrell Lewis, Patrick Benavidez, Mo Jamshidi

  Publication date: 08 October 2021

  Publication venue: 2021 World Automation Congress (WAC)

  Publication keywords: Reinforcement Learning, Virtual Reality, Au-tonomous Car, Simulation, Multi-Agent Systems, Exploration

  Most Relevant Task: Task T1-3

  Other Relevant Tasks: NA

- **Title:** Voice Activation and Control to Improve Human Robot Interactions with IoT Perspectives

  Author(s): Berat A. Erol, Conor Wallace, Patrick J. Benavidez, PhD Mo Jamshidi, PhD.

  Publication date: 2018

  Publication venue: World Automation Congress 2018

  Publication keywords: Human robot interactions, humanoid robot, assistive robotics, Amazon Echo, Echo controls robot, Internet of robotic things, smart home

  Most Relevant Task: Task T1-4

  Other Relevant Tasks: NA

- **Title:** Multi-Agent Exploration for Faster and ReliableDeep Q-Learning Convergence in Reinforcement Learning

  Author(s): Abhijit Majumdar, Patrick J. Benavidez, PhD and Mo Jamshidi, PhD.

  Publication date: 2018

  Publication venue: World Automation Congress 2018

  Publication keywords: Reinforcement Learning, Q-learning, DQN, au-tonomous car, simulation, multi-agent, exploration

Most Relevant Task: Task T1-4

Other Relevant Tasks: Task T1-3

- **Title:** Application of a Scaled Ground Vehicle in a Testbed of Heterogeneous Autonomous Systems

  Author(s): Alexander Ibarra, Patrick Benavidez and Mo Jamshidi

  Publication date: 2021

  Publication venue: 2021 World Automation Congress

  Publication keywords: Robot Operating System (ROS), Unmanned Ground Vehicle, Hector Slam, Autonomous Navigation, Discrete Controllers, Computer Vision

  Most Relevant Task: Task T1-4

  Other Relevant Tasks: Task T1-3

- **Title:** Survey of Datafusion Techniques for Laser and Vision Based Sensor Integration for Autonomous Navigation

  Author(s): Prasanna Kolar, Patrick Benavidez, Mo Jamshidi

  Publication date: April 2020

  Publication venue: MDPI Sensors Journal

  Publication keywords: datafusion; data fusion; multimodal; fusion; information fusion; survey; review; RGB; SLAM; localization; obstacle detection; obstacle avoidance; navigation; deep learning; neural networks; mapping; LiDAR; optical; vision; stereo vision; autonomous systems; data integration; data alignment; robot; mobile robot

  Most Relevant Task: Task T1-4

  Other Relevant Tasks: Task T1-3

- **Title:** Low-Cost Heterogeneous Unmanned Ground Vehicle (UGV) Testbed for Systems of Autonomous Vehicles Research

  Author(s): Aldo Jaimes, Javier Gonzalez, Alexander Ibarra, Patrick Benavidez, and Mo Jamshidi

  Publication date: June 2019

  Publication venue: 2019 Systems of Systems Engineering Conference (SoSE 2019)

  Publication keywords: Localization, Robot Operating System, Unmanned Ground Vehicles

  Most Relevant Task: Task T1-4

  Other Relevant Tasks: Task T1-3

- **Title:** Accuracy of the HTC Vive Tracker for Indoor Localization

  Author(s): Jonathan Lwowski, Abhijit Majumdar, Patrick Benavidez, Jeff Prevost, Mo Jamshidi

  Publication date: 2021

Publication venue: SMC Magazine

Publication keywords: Localization, formation control, infrared, sensor, calibration

Most Relevant Task: Task T1-4

Other Relevant Tasks: Task T1-3

- **Title:** Toward Artificial Emotional Intelligence for Cooperative Social Human-Machine Interaction

  Author(s): Berat Erol, Abhijit Majumdar, Patrick Benavidez, Paul Rad, Raymond Choo, Mo Jamshidi

  Publication date: June 2019

  Publication venue: IEEE Transactions on Computational Social Systems

  Publication keywords: Assistive robotics, human-machine interactions, humanoid robot, Internet of robotic things, smart home, supervisory control

  Most Relevant Task: Task T1-4

  Other Relevant Tasks: Task T1-3

- **Title:** Voice Activation and Control to ImproveHuman Robot Interactions with IoT Perspectives

  Author(s): Berat A. Erol, Conor Wallace, Patrick J. Benavidez, PhD Mo Jamshidi, PhD.

  Publication date: 2018

  Publication venue: IEEE World Automation Congress 2018

  Publication keywords: Human robot interactions, humanoid robot, assistive robotics, Amazon Echo, Echo controls robot, Internet of robotic things, smart home

  Most Relevant Task: Task T1-4

  Other Relevant Tasks: Task T1-3

- **Title:** Multi-Agent Exploration for Faster and ReliableDeep Q-Learning Convergence in Reinforcement Learning

  Author(s): Abhijit Majumdar, Patrick J. Benavidez, PhD and Mo Jamshidi, PhD.

  Publication date: 2018

  Publication venue: IEEE World Automation Congress 2018

  Publication keywords: Reinforcement Learning, Q-learning, DQN, au-tonomous car, simulation, multi-agent, exploration

  Most Relevant Task: Task T1-4

  Other Relevant Tasks: Task T1-3

- **Title:** Improved Route Optimization for Autonomous Ground Vehicle Navigation

  Author(s): Ibrahim Mohammed, Berat A. Erol, Ikram Hussain Mohammed, Patrick J. Benavidez, Mo Jamshidi.

  Publication date: 2017.

Publication venue: 13th Annual IEEE System of Systems Engineering Conference July 18th, 2017.

Publication keywords: Autonomous Navigation, Path Planning, Robot Operating System, Differential drive robot.

Most Relevant Task: Task T1-4

Other Relevant Tasks: Task T1-3

- **Title:** A Deep Vision Landmark Framework for Robot Navigation.

  Author(s): Abhijith Ravikumar Puthussery, Karthik Pai Haradi, Berat A. Erol, Patrick Benavidez, Paul Rad, Mo Jamshidi.

  Publication date: 2017.

  Publication venue: 13th Annual IEEE System of Systems Engineering Conference July 18th, 2017.

  Publication keywords: Autonomous Navigation, Robot Operating System, Machine learning, TensorFlow, Convolutional Neural Network, Differential drive robot.

  Most Relevant Task: Task T1-4

  Other Relevant Tasks: Task T1-3

- **Title:** Development of Robot Operating System (ROS) Compatible Open Source Quadcopter Flight Controller and Interface.

  Author(s): Abhijit Majumdar, Nicholas Gamez, Patrick J Benavidez, Mo Jamshidi.

  Publication date: 2017.

  Publication venue: 13th Annual IEEE System of Systems Engineering Conference July 18th, 2017.

  Publication keywords: Quadcopter; Robot Operating System; Unmanned Aerial Vehicle; Flight controller; interface.

  Most Relevant Task: Task T1-4

  Other Relevant Tasks: Task T1-3

Title: Cloud-based Control and vSLAM through Cooperative Mapping and Localization

Author(s): Berat A. Erol, Patrick Benavidez, Satish Vaishnav, Joaquin Labrado, Mo Jamshidi

Publication date: 2016.

Publication venue: WAC 2016 World Automation Congress Rio Grande, PR, USA

Publication keywords: Cooperative SLAM; vSLAM; RGB-D; Cloud Computing; Robotics

Most Relevant Task: Task T1-4

Other Relevant Tasks: Task T1-3

- **Title:** 3D Printed Underwater Housing

  Author(s): Benjamin Champion, Mo Jamshidi, Matthew Joordens

  Publication date: 2016.

Publication venue: WAC 2016 World Automation Congress Rio Grande, PR, USA

Publication keywords: 3D Printing, Additive Manufacturing, Underwater Box, AUV

Most Relevant Task: Task T1-4

Other Relevant Tasks: Task T1-3

- **Title:** Autonomous Mobile Robot Platform with Multi-Variant ask-Specific End-Effector and Voice Activation

  Author(s): Jonathan Tapia, Eric Wineman, Patrick Benavidez, Aldo Jaimes, Ethan Cobb, John Parsi, Dan Clifton, Mo Jamshidi and Benjamin Champion.

  Publication date: 2016.

  Publication venue: WAC 2016 World Automation Congress Rio Grande, PR, USA

  Publication keywords: Polylactic Acid (PLA), Acrylonitrie Butadiene Styrene (ABS), Mecanum wheel, Solid Works, Voice Recognition, Fused Filament Fabrication (FFF), 3D printing

  Most Relevant Task: Task T1-4

  Other Relevant Tasks: Task T1-3

- **Title:** Cloud-based Control and vSLAM through Cooperative Mapping and Localization

  Author(s): Berat A. Erol, Satish Vaishnav, Joaquin D. Labrado, Patrick Benavidez, and Mo Jamshidi

  Publication date: 2016

  Publication venue: World Automation Congress 2016

  Publication keywords: Cooperative SLAM, vSLAM, Robotics, RGB-D, Cloud Computing

  Most Relevant Task: Task T1-4

  Other Relevant Tasks: Task T1-3

- **Title:** A Symbolic Approach for Multi-target Dynamic Reach-avoid Problem

  Author(s): Laya Shamgah, Tadewos G. Tadewos, Ali Karimoddini, Abdollah Homaifar

  Publication date: 2018

  Publication venue: Proc. of IEEE 14th International Conference on Control and Automation (ICCA)

  Publication keywords: reach-avoid, symbolic planning, motion planning, hybrid control, reactive synthesis, multi-target

  Most Relevant Task: Task T1-5

  Other Relevant Tasks: NA

- **Title:** A Symbolic Motion Planning Approach for the Reach-avoid Problem

  Author(s): Laya Shamgah, Ali Karimoddini, Abdollah Homaifar

  Publication date: 2016

  Publication venue: Proc. of 2016 IEEE International Conference on Systems, Man, and Cybernetics (SMC), Budapest

Publication keywords: reach-avoid, symbolic planning, motion planning, hybrid control, reactive synthesis

Most Relevant Task: Task T1-5

Other Relevant Tasks: NA

- **Title:** Path Planning and Control of Autonomous Vehicles in Dynamic Reach-Avoid Scenarios

  Author(s): Laya Shamgah, Tadewos G. Tadewos, Ali Karimoddini, and Abdollah Homaifar

  Publication date: 2018

  Publication venue: Proc. IEEE Conference on Control Technology and Applications (CCTA), Copenhagen

  Publication keywords: reach-avoid, symbolic planning, motion planning, hybrid control, smooth control

  Most Relevant Task: Task T1-5

  Other Relevant Tasks: NA or list the tasks that are relevant to this publication it is related to multiple tasks.

- **Title:** On the Design of Smooth Hybrid Controllers for a Class of Non-linear Systems

  Author(s): Laya Shamgah, Tadewos G. Tadewos, Ali Karimoddini

  Publication date: Dec 2020

  Publication venue: IET Control Theory & Applications

  Publication keywords: Provide Publication keywords

  Most Relevant Task: Task T1-5

  Other Relevant Tasks: NA

- **Title:** Reactive Symbolic Planning and Control in Dynamic Adversarial Environments

  Author(s): Laya Shamgah, Tadewos G. Tadewos, Abdullah Al Redwan Newaz, Ali Karimoddini, Albert C. Esterline

  Publication date: Under review

  Publication venue: IEEE Transactions on Automatic Control

  Publication keywords: reach-avoid, symbolic planning, motion planning, hybrid control, smooth control

  Most Relevant Task: Task T1-5

  Other Relevant Tasks: NA

- **Title:** Cloud Robotics: A Software Architecture For Heterogeneous Large-Scale Autonomous Robots

  Author(s): Ali Miraftabzadeh, Nicolas Gallardo, Nicolas Gamez, Karthik Haradi, Paul Rad, Mo Jamshidi

  Publication date: October 2016

  Publication venue: 2016 World Automation Congress

Publication keywords: cloud robotics; heterogeneous agents; large-scale autonomous; Hadoop-MapReduce

- **Title:** Formation Control Implementation Using Kobuki TurtleBots and Parrot Bebop Drone

  Author(s): Nicolas Gallardo, Karthik Haradi, Berat A. Erol, Patrick Benavidez Mo Jamshidi

  Publication date: July 2016

  Publication venue: World Automation Conference 2016, Rio Grande, Puerto Rico, USA

  Publication keywords: formation control, virtual leader, robot, drone, UGV, UAV, turtle-bot, parrot bebop

- **Title:** Task Allocation Using Parallelized Clustering and Auctioning Algorithms for Heterogeneous Robotic Swarms Operating on a Cloud Network

  Author(s): Jonathan Lwowski, Patrick Benavidez, John J. Prevost, and Mo Jamshidi

  Publication date: March 2017

  Publication venue: Springer-Verlag Autonomy and Artificial Intelligence

- **Title:** Pedestrian Detection System Using Deep Convolutional Neural Networks.

  Author(s): Jonathan Lwowski, Prasanna Kolar, Patrick J Benavidez, Paul Rad, John J Prevost, Mo Jamshidi.

  Publication date: July 2017

  Publication venue: 2017 IEEE System of System Engineering Conference.

- **Title:** Lightweight Multi Car Dynamic Simulator forReinforcement Learning

  Author(s): Abhijit Majumdar, Patrick Benavidez and Mo Jamshidi

  Publication date: 2018

  Publication venue: IEEE World Automation Congress 2018

  Publication keywords: multi agent, simulation, reinforcement learning,multiple instances, multi environment

- **Title:** Towards the Optimal Placement of Containerized Applications on a Cloud-Edge Network

  Author(s): James Nelson, Jonathan Lwowski, Patrick Benavidez, Jeff Prevost, Mo Jamshidi

  Publication date: September 2019

  Publication venue: 2020 IEEE International Systems Conference (SysCon)

  Publication keywords: cloud computing; edge computing; fuzzy control; expert systems; containerization

- **Title:** Down Sampling using Simulated Annealing

  Author(s): Sean Ackels, Patrick Benavidez and Mo Jamshidi

  Publication date: June 2020

  Publication venue: 2020 Systems of Systems Engineering Conference (SoSE 2020)

Publication keywords: Optimization; Simulated Annealing; Radial Basis Function; LASSO; Data Reduction

- **Title:** Comparison of Cloud Architectures for Mobile Sensor Optimization Problem Implementation

  Author(s): Sean Ackels, Patrick Benavidez and Mo Jamshidi

  Publication date: June 2020

  Publication venue: 2020 Systems of Systems Engineering Conference (SoSE 2020)

  Publication keywords: Cloud Computing, Lambda Architecture, Kappa Architecture, Radial Basis Function, Mobile Sensor Optimization Problem, Swarm Robotics

- **Title:** A Survey of Modern Roadside Unit Deployment Research

  Author(s): Sean Ackels, Patrick Benavidez, Mo Jamshidi

  Publication date: August 2021

  Publication venue: 2021 World Automation Congress

  Publication keywords: Vehicular ad hoc Networks (VANET), Automated Vehicles, Smart Cities, Roadside Unit (RSU), RSU Deployment

- **Title:** Solving the Roadside Unit Deployment Problem: A Survey

  Author(s): Sean Ackels, Patrick Benavidez, Mo Jamshidi, Asad Madni

  Publication date: Submitted

  Publication venue: IEEE Transactions on Intelligent Transportation Systems

  Publication keywords: Intelligent Transportation Systems, Roadside Unit, RSU Placement Problem

  Most Relevant Task: Task T1-6

  Other Relevant Tasks: NA

- **Title:** A Learning-based Approach for Diagnosis and Disagnosability of Unknown Discrete Event Systems

  Author(s): Ira Wendell Bates II, Ali Karimoddini, and Mohammad Karimadini

  Publication date: (*under review*)

  Publication venue: IEEE International Conference on Systems, Man, and Cybernetics (SMC)

  Publication keywords: Active Learning, Fault diagnosis, Automata, Systematics, Discrete Event Systems

  Most Relevant Task: Task T2-1

  Other Relevant Tasks: NA

- **Title:** Fault Diagnosis of Discrete Event Systems Under Unknown Initial Conditions

  Author(s): Alejandro White; Ali Karimoddini; Rong Su

  Publication date: 2019

Publication venue: IEEE Transactions on Automatic Control

Publication keywords: Fault diagnosis, Discrete-event systems, Estimation, Systems operation, Automata, Systematics, Analytical models

Most Relevant Task: Task T2-1

Other Relevant Tasks: NA

- **Title:** Asynchronous fault diagnosis of Discrete Event Systems

  Author(s): Alejandro White; Ali Karimoddini

  Publication date: 2017

  Publication venue: American Control Conference

  Publication keywords: Fault diagnosis, Discrete-event systems, Estimation, Monitoring, Automata, Systematics, Tools, Complex Systems

  Most Relevant Task: Task T2-1

  Other Relevant Tasks: NA

- **Title:** Semi-asynchronous fault diagnosis of Discrete Event Systems

  Author(s): Alejandro White; Ali Karimoddini

  Publication date: 2016

  Publication venue: IEEE International Conference on Systems, Man, and Cybernetics (SMC)

  Publication keywords: Fault diagnosis, Discrete-event systems, Estimation, Cybernetics, Systematics, Systems operation

  Most Relevant Task: Task T2-1

  Other Relevant Tasks: NA

- **Title:** Event-based Fault Diagnosis for an Unknown Plant

  Author(s): Mohammad Mahdi Karimi; Ali Karimoddini; Alejandro P White; Ira Wendell Bates

  Publication date: 2016

  Publication venue: IEEE 55th Conference on Decision and Control (CDC)

  Publication keywords: Automata, Fault diagnosis, Discrete-event systems, Unmanned aerial vehicles, Monitoring, Fuels

  Most Relevant Task: Task T2-1

  Other Relevant Tasks: NA

- **Title:** Achieving Fault-tolerance and Safety of Discrete-event Systems through Learning

  Authors: Jin Dai, Ali Karimoddini, Hai Lin

  Publication date: 2016

  Publication venue: The 2016 IEEE American Control Conference

Publication keywords: Fault-tolerance, Discrete Event Systems

Most Relevant Task: Task T2-2

Other Relevant Tasks: NA

- **Title:** Application of Big Data Analytics via Cloud Computing

  Authors: Yunus Yetis, Ruthvik Sara, Berat Erol, Halid Kaplan, Abdurrahman Akuzum, Mo Jamshidi

  Publication date: 2016

  Publication venue: 2016 World Automation Congress

  Publication keywords: Cloud Computing, Data Analytics, MapReduce

  Most Relevant Task: Task T2-2

  Other Relevant Tasks: NA

- **Title:** Data-driven Fault Detection of Un-manned Aerial Vehicles Using Supervised Learning Over Cloud Networks

  Authors: Parsa Yousefi, Hamid Fekriazgomi, John J. Prevost, Mo Jamshidi

  Publication date: 2018

  Publication venue: IEEE World Automation Congress 2018

  Publication keywords: Un-manned Aerial Vehicles, Signal-based Fault Detection, Machine Learning, Supervised Learning, Data Prediction, Linear Discriminant Analysis, Logistic Regression

  Most Relevant Task: Task T2-2

  Other Relevant Tasks: NA

- **Title:** Predicting Fault Behaviors of Networked Control Systems Using Deep Learning for Mobile Robots

  Authors: Conor Wallace, Patrick Benavidez, Mo Jamshidi

  Publication date: 2019

  Publication venue: 2019 Systems of Systems Engineering Conference (SoSE 2019)

  Publication keywords: Deep Learning, Fuzzy Logic, Long Short-Term Memory Units, Recurrent Neural Networks, Robot Operating System, Unmanned Ground Vehicles

  Most Relevant Task: Task T2-2

  Other Relevant Tasks: NA

- **Title:** Real-Time Distributed Ensemble Learning for Fault Detection of an Unmanned Ground Vehicle

  Authors: C. Wallace, S. Ackels, P. Benavidez, M. Jamshidi

  Publication date: 2020

  Publication venue: 2020 15th Annual International Systems of System Engineering Conference (SoSE)

Publication keywords: Cloud Computing, System of Systems, Fault Diagnostics, LSTM, Ensemble Learning

Most Relevant Task: Task T2-2

Other Relevant Tasks: NA

- **Title:** Deep Learning and Spectral Clustering for Fault Tolerant Control of an Unmanned Ground Vehicle with Dislocated Actuator Fault

  Authors: C. Wallace, S. Ackels, P. Benavidez, M. Jamshidi

  Publication date: 2020

  Publication venue: IEEE Systems Journal

  Publication keywords: Fault Tolerant Control, Spectral Clustering, Mobile Robots, Neural Networks, Control Systems, Evolutionary Computation

  Most Relevant Task: Task T2-2

  Other Relevant Tasks: NA

- **Title:** Provide Title

  Author(s): Provide Author(s)

  Publication date: Provide Publication date

  Publication venue: Provide Meeting Location

  Publication keywords: Provide Publication keywords

  Most Relevant Task: Task T2-4

  Other Relevant Tasks: NA or list the tasks that are relevant to this publication it is related to multiple tasks.

- **Title:** Provide Title

  Author(s): Provide Author(s)

  Publication date: Provide Publication date

  Publication venue: Provide Meeting Location

  Publication keywords: Provide Publication keywords

  Most Relevant Task: Task T2-4

  Other Relevant Tasks: NA or list the tasks that are relevant to this publication it is related to multiple tasks.

- **Title:** Enabling Content-Centric Device-to-Device Communication in the Millimeter-Wave Band

  Author(s): Niloofar Bahadori, Mahmoud Nabil, Brian Kelley, Abdollah Homaifar

  Publication date: April 4, 2021

  Publication venue: IEEE Transactions on Mobile Computing

  Publication keywords: Content-centric network, information-centric network, device-to-device, mmWave, beamwidth selection, peer association

Most Relevant Task: Task T2-5

Other Relevant Tasks: NA

- **Title:** Antenna Beamwidth Optimization in Directional Device-to-Device Communication Using Multi-Agent Deep Reinforcement Learning

  Author(s): Niloofar Bahadori, Mahmoud Nabil, Abdollah Homaifar

  Publication date: Under-review

  Publication venue: IEEE ACCESS

  Publication keywords: D2D, mmWave, Deep reinforcement learning

  Most Relevant Task: Task T2-5

  Other Relevant Tasks: NA

- **Title:** Device-to-Device Communications in the Millimeter Wave Band: A Novel Distributed Mechanism

  Author(s): Niloofar Bahadori, Nima Namvar, Brian Kelley, Abdollah Homaifar

  Publication date: April 2019

  Publication venue: IEEE Wireless Telecommunication Symposium

  Publication keywords: D2D, mmWave, beam alignment, blockage, stochastic geometry

  Most Relevant Task: Task T2-5

  Other Relevant Tasks: NA

- **Title:** Device-to-Device Communications in Millimeter Wave Band: Impact of Beam Alignment Error

  Author(s): Niloofar Bahadori, Nima Namvar, Brian Kelley, Abdollah Homaifar

  Publication date: April 2018

  Publication venue: IEEE Wireless Telecommunication Symposium

  Publication keywords: D2D, mmWave, Stochastic geometry

  Most Relevant Task: Task T2-5

  Other Relevant Tasks: NA

- **Title:** Context-Aware D2D Peer Selection for Load Distribution in LTE Networks

  Author(s): Nima Namvar, Niloofar Bahadori, Fatemeh Afghah

  Publication date: November 2015

  Publication venue: IEEE Asilomar Conference on Signals, Systems and Computers

  Publication keywords: Device to Device communication, Context information, Link selection, Matching game

  Most Relevant Task: Task T2-5

  Other Relevant Tasks: NA

- **Title:** Jamming in the Internet of Things: AGame-Theoretic Perspective

Author(s): Nima Namvar, Walid Saad, Niloofar Bahadori, Brian Kelley

Publication date: December 2016

Publication venue: IEEE Global Communications Conference (GLOBECOM)

Publication keywords: Internet of Things; Jamming; Security; Game theory

Most Relevant Task: Task T2-5

Other Relevant Tasks: NA

- **Title:** IoT-Enabled Autonomous System Collaboration for Disaster-Area Management

  Author(s): Abenezer Girma, Niloofar Bahadori, Mrinmoy Sarkar, Tadewos G Tadewos, Mohammad R Behnia, M Nabil Mahmoud, Ali Karimoddini, Abdollah Homaifar

  Publication date: July 2020

  Publication venue: IEEE/CAA Journal of Automatica Sinica

  Publication keywords: Architectural analysis and design language (AADL) and cloud computing, disaster area management, internet of things (IoT), message queuing telemetry transport (MQTT), unmanned aerial vehicle (UAV), unmanned ground vehicle (UGV

  Most Relevant Task: Task T2-5

  Other Relevant Tasks: NA

- **Title:** Adaptive and Neural Network Based Control of Pitch of Unmanned Aerial Vehicles

  Author(s): M. Matthews; S. Yi

  Publication date: 2019

  Publication venue: Journal of Mechatronics and Robotics

  Publication keywords: UAV, PID, 2DOF-PID, Model Reference Adaptive Control (MRAC), Neural Network (NN).

  Most Relevant Task: Task T2-6

  Other Relevant Tasks: NA

- **Title:** Analysis of Time Delays in Quadrotor Systems and Design of Control

  Author(s): Armah, Stephen and Yi, S.

  Publication date: 2017

  Publication venue: Time Delay Systems: Theory, Numerics, Applications, and Experiments

  Publication keywords: Time Delay, networked systems, uncertainties, teleoperation, UAV.

  Most Relevant Task: Task T2-6

  Other Relevant Tasks: NA

- **Title:** Design of feedback control for quadrotors considering signal transmission delays

  Author(s): Armah, Stephen K. and Yi, Sun and Choi, Wonchang

  Publication date: 2016

Publication venue: International Journal of Control, Automation and Systems

Publication keywords: Transmission delay, estimation, feedback control, robust control, UAV.

Most Relevant Task: Task T2-6

Other Relevant Tasks: NA

- **Title:** Analysis of the Effects of Communication Trust and Delay on Consensus of Multi-Agent Systems

  Author(s): J. Sumpter; C. Thomas; S. Yi; A. Kruger

  Publication date: 2019

  Publication venue: 2019 IEEE Connected and Automated Vehicles Symposium

  Publication keywords: Delay, estimation, internet, Smith predictor, teleoperation, UAV.

  Most Relevant Task: Task T2-6

  Other Relevant Tasks: NA

- **Title:** Analysis of the effects of communication delays for consensus of networked multi-agent systems

  Author(s): Allen-Prince, Myrielle; Thomas, Christopher;Yi, Sun;

  Publication date: 2017

  Publication venue: International Journal of Control, Automation and Systems

  Publication keywords: Consensus, delay, multiple-agent systems, sensitivity, stability, topology.

  Most Relevant Task: Task T2-6

  Other Relevant Tasks: NA

- **Title:** An IoT self organizing network for 5G dense network interference alignment

  Author(s): Anil Kumar Yerrapragada, Brian Kelley

  Publication date: 6/18/2017

  Publication venue: 2017 12th System of Systems Engineering Conference (SoSE)

  Publication keywords: Receivers, 5G mobile communication, MIMO, Transmitters, Interference cancellation, Radio access networks

  Most Relevant Task: Task T2-7

  Other Relevant Tasks: NA

- **Title:** Design of K-user massive MIMO networks

  Author(s): Anil Kumar Yerrapragada, Brian Kelley

  Publication date: 10/19/2017

  Publication venue: 2017 IEEE 8th Annual Ubiquitous Computing, Electronics and Mobile Communication Conference (UEMCON)

Publication keywords: 5G mobile communication, channel estimation, densification, hypercube networks, interference suppression, MIMO communication, radiofrequency interference, receiving antennas

Most Relevant Task: Task T2-7

Other Relevant Tasks: NA

- **Title:** On the application of mimo space-time coding to physical layer security in sub-6 ghz 5g

  Author(s): Patrick Ormond, Anil Kumar Yerrapragada, Brian Kelley

  Publication date: 5/19/2019

  Publication venue: 2019 14th Annual Conference System of Systems Engineering (SoSE)

  Publication keywords: wireless communication links, enhanced PRP-STBC scheme, 1-bit codebook, dense 5G cellular geometries, PRP key based communication framework, performance improvement, transmission rate, MIMO space time block coding, precoder matrix index based scheme, private random precoding, key-based physical layer security, key bit error rate performance

  Most Relevant Task: Task T2-7

  Other Relevant Tasks: NA

- **Title:** On the Application of Key-Based Physical Layer Security in 5G Heterogeneous Networks

  Author(s): Anil Kumar Yerrapragada, Patrick Ormond, Brian Kelley

  Publication date: 11/12/2019

  Publication venue: MILCOM 2019-2019 IEEE Military Communications Conference (MILCOM)

  Publication keywords: 5G mobile communication, Physical layer, Network security, Base stations, Geometry, Heterogeneous networks

  Most Relevant Task: Task T2-7

  Other Relevant Tasks: NA

- **Title:** Very High Throughput Internet of Things Networks with access points and devices

  Author(s): Anil Kumar Yerrapragada, Brian Kelley

  Publication date: 11/12/2019

  Publication venue: MILCOM 2019-2019 IEEE Military Communications Conference (MILCOM)

  Publication keywords: MIMO communication, Receivers, Transmitters, Mathematical model, Interference cancellation, Throughput

  Most Relevant Task: Task T2-7

  Other Relevant Tasks: NA

- **Title:** Error rate analysis of physical layer security for sub-6 ghz 5g network planning

Author(s): Patrick Ormond, Anil Kumar Yerrapragada, Brian Kelley

Publication date: 5/19/2019

Publication venue: 2019 14th Annual Conference System of Systems Engineering (SoSE)

Publication keywords: 5G mobile communication, error statistics, MIMO communication, precoding, probability, Rayleigh channels, telecommunication security

Most Relevant Task: Task T2-7

Other Relevant Tasks: NA

- **Title:** On the Application of K-User MIMO for 6G Enhanced Mobile Broadband

  Author(s): Anil Kumar Yerrapragada, Brian Kelley

  Publication date: 1/2020

  Publication venue: Sensors

  Publication keywords: beyond-5G, 6G, MIMO, interference alignment, K-User MIMO, OFDM

  Most Relevant Task: Task T2-7

  Other Relevant Tasks: NA

- **Title:** Design of a Smooth Landing Trajectory Tracking System for a Fixed-wing Aircraft

  Author(s): Solomon Gudeta and Ali Karimoddini

  Publication date: 29 August 2019

  Publication venue: Philadelphia, PA, USA

  Most Relevant Task: Task T3-1

  Other Relevant Tasks: NA

- **Title:** Robust Dynamic Average Consensus for a Network of Agents with Time-varying Reference Signals

  Author(s): Solomon Gudeta, Ali Karimoddini, and Mohammadreza Davoodi

  Publication date: 14 December 2020

  Publication venue: Toronto, ON, Canada

  Publication keywords: Robust dynamic average consensus, Timevarying signals, Mutli- agent systems, Zero steady-state error, Chattering effect, Switching topology, Networked control systems

  Most Relevant Task: Task T3-1

  Other Relevant Tasks: NA

- **Title:** Leaderless Swarm Formation Control: From Global Specifications to Local Control Laws

  Author(s): Solomon Gudeta, Ali Karimoddini, Mohammadreza Davoodi, and Ioannis Raptis

  Publication date: 28 July 2021

Publication venue: New Orleans, LA, USA
Publication keywords: Distributed control, Cooperative control, Robotic swarms
Most Relevant Task: Task T3-1
Other Relevant Tasks: NA

- **Title:** Flexible Coordinated Collective Motion of Robotic Swarms
  Author(s): Solomon Gudeta, Ali Karimoddini, and Ioannis Raptis
  Publication date: under review
  Publication venue: IEEE Transactions on Cybernetics
  Most Relevant Task: Task T3-1
  Other Relevant Tasks: NA

- **Title:** Robust collective navigation of swarm of quadrotor
  Author(s): Solomon Gudeta and Ali Karimoddini
  Publication date: under preparation
  Publication venue: Conference on Control Technology and Applications
  Most Relevant Task: Task T3-1
  Other Relevant Tasks: NA

- **Title:** Distributed Swarm Time-Varying Formation Tracking With Collision Avoidance Subjected to Switching Topologies
  Author(s): Solomon Gudeta and Ali Karimoddini
  Publication date: under preparation
  Publication venue: IEEE Robotics and Automation Letters
  Most Relevant Task: Task T3-1
  Other Relevant Tasks: NA

- **Title:** Distributed swarm assignment and pattern formation with collision avoidance
  Author(s): Solomon Gudeta and Ali Karimoddini
  Publication date: under preparation
  Publication venue: IEEE Systems Journal
  Most Relevant Task: Task T3-1
  Other Relevant Tasks: NA

- **Title:** On-the-fly decentralized tasking of autonomous vehicles
  Author(s): Tadewos G. Tadewos, L Shamgah, A Karimoddini
  Publication date: 2019
  Publication venue: IEEE 58th Conference on Decision and Control (CDC)

Publication keywords: Intelligent Vehicles, Behavior Tree, Automatic Synthesis, Multi-Agent

Most Relevant Task: Task T3-2

Other Relevant Tasks: NA

- **Title:** Automatic safe behaviour tree synthesis for autonomous agents

  Author(s): Tadewos G. Tadewos, L Shamgah, A Karimoddini

  Publication date: 2019

  Publication venue: IEEE 58th Conference on Decision and Control (CDC)

  Publication keywords: Intelligent Vehicles, Behavior Tree, Automatic Synthesis, Safety

  Most Relevant Task: Task T3-2

  Other Relevant Tasks: NA

- **Title:** Specification-guided Behavior Tree Synthesis and Execution

  Author(s): Tadewos G. Tadewos, Abdullah Al Redwan Newaz, A Karimoddini

  Publication date: 2022

  Publication venue: Expert Systems with Applications

  Publication keywords: Intelligent Vehicles, Behavior Tree, Automatic Synthesis, Safety

  Most Relevant Task: Task T3-2

  Other Relevant Tasks: NA

- **Title:** An Effective Model for Human Cognitive Performance within a Human-Robot Collaboration Framework

  Author(s): Md Khurram Monir Rabby, Mubbashar Khan, Ali Karimoddini, and Steven Xiaochun Jiang

  Publication date: 2019

  Publication venue: IEEE 2019 International Conference on Systems, Man, and Cybernetics (SMC)

  Publication keywords: Human-Robot Cooperation, Physical performance, Cognitive performance, Dynamic model

  Most Relevant Task: Task T3-3

  Other Relevant Tasks: NA

- **Title:** Modeling of Trust Within a Human-Robot Collaboration Framework

  Author(s): Md Khurram Monir Rabby, Mubbashar Khan, Ali Karimoddini, and Steven Xiaochun Jiang

  Publication date: 2020

  Publication venue: IEEE 2020 International Conference on Systems, Man, and Cybernetics (SMC)

Publication keywords: Human-Robot Collaboration (HRC), Human-machine teaming, Trust, Physical performance, Cognitive performance, Dynamic model

Most Relevant Task: Task T3-3

Other Relevant Tasks: NA

- **Title:** Performance-Aware Trust Modeling within a Human-Multi-Robot Collaboration Setting

  Author(s): Md Khurram Monir Rabby, Mubbashar Khan, Ali Karimoddini, and Steven Xiaochun Jiang

  Publication date: Under review

  Publication venue: ACM Transaction on Human-Robot Collaboration

  Publication keywords: Human-Robot Collaboration (HRC), Trust, Human Performance, Multi-Robot Performance

  Most Relevant Task: Task T3-3

  Other Relevant Tasks: NA

- **Title:** A Novel Clustering Algorithm Based on Fitness Proportionate Sharing

  Author(s): Xuyang Yan, Abdollah Homaifar, Shabnam Nazmi and Mohammad Razeghi-Jahromi

  Publication date: 2017

  Publication venue: 2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC2017)

  Publication keywords: clustering, unsupervised learning.

  Most Relevant Task: Task T3-4

  Other Relevant Tasks: NA

- **Title:** Unsupervised feature selection through fitness proportionate sharing clustering

  Author(s): Xuyang Yan, Abdollah Homaifar, Gabriel Awogbami, Abenezer Girma

  Publication date: 2018

  Publication venue: 2018 IEEE International Conference on Systems, Man, and Cybernetics (SMC2018)

  Publication keywords: Feature selection, dimension reduction.

  Most Relevant Task: Task T3-4

  Other Relevant Tasks: NA

- **Title:** Multi-label classification using genetic-based machine learning

  Author(s): Shabnam Nazmi, Xuyang Yan, Abdollah Homaifar

  Publication date: 2018

  Publication venue: 2018 IEEE International Conference on Systems, Man, and Cybernetics (SMC2018)

Publication keywords: Multi-label classification, genetic algorithm.

Most Relevant Task: Task T3-4

Other Relevant Tasks: NA

- **Title:** Possibility rule-based classification using function approximation

  Author(s): Shabnam Nazmi, Abdollah Homaifar

  Publication date: 2018

  Publication venue: 2018 IEEE International Conference on Systems, Man, and Cybernetics (SMC2018)

  Publication keywords: Uncertain label classification, possibility theory, learning classifier systems

  Most Relevant Task: Task T3-4

  Other Relevant Tasks: NA

- **Title:** A novel streaming data clustering algorithm based on fitness proportionate sharing

  Author(s): Xuyang Yan, Mohammad Razeghi-Jahromi, Abdollah Homaifar, Berat A Erol, Abenezer Girma, Edward Tunstel

  Publication date: 2019

  Publication venue: IEEE Access

  Publication keywords: Data stream, clustering, density-based clustering.

  Most Relevant Task: Task T3-4

  Other Relevant Tasks: NA

- **Title:** Driver identification based on vehicle telematics data using LSTM-recurrent neural network

  Author(s): Abenezer Girma, Xuyang Yan, Abdollah Homaifar

  Publication date: 2019

  Publication venue: 2019 IEEE 31st International Conference on Tools with Artificial Intelligence (ICTAI)

  Publication keywords: Driver identification,deep learning,LSTM

  Most Relevant Task: Task T3-4

  Other Relevant Tasks: NA

- **Title:** An efficient unsupervised feature selection procedure through feature clustering

  Author(s): Xuyang Yan, Shabnam Nazmi, Berat A Erol, Abdollah Homaifar, Biniam Gebru, Edward Tunstel

  Publication date: 2020

  Publication venue: Pattern Recognition Letter

  Publication keywords: Unsupervised feature selection, feature clustering

  Most Relevant Task: Task T3-4

Other Relevant Tasks: NA

- **Title:** IoT-enabled autonomous system collaboration for disaster-area management

  Author(s): Abenezer Girma, Niloofar Bahadori, Mrinmoy Sarkar, Tadewos G Tadewos, Mohammad R Behnia, M Nabil Mahmoud, Ali Karimoddini, Abdollah Homaifar

  Publication date: 2020

  Publication venue: IEEE/CAA Journal of Automatica Sinica

  Publication keywords: IoT-enabled autonomous system collaboration, disaster-area management.

  Most Relevant Task: Task T3-4

  Other Relevant Tasks: Task T3-3, Task T2-5, Task T3-6

- **Title:** Evolving multi-label classification rules by exploiting high-order label correlations

  Author(s): Shabnam Nazmi, Xuyang Yan, Abdollah Homaifar, Emily Doucette

  Publication date: 2020

  Publication venue: Neurocomputing

  Publication keywords: Multi-label classification, Label correlation, Learning classifier systems

  Most Relevant Task: Task T3-4

  Other Relevant Tasks: NA

- **Title:** A clustering-based framework for classifying data streams

  Author(s): Xuyang Yan, Abdollah Homaifar, Mrinmoy Sarkar, Abenezer Girma, Edward Tunstel

  Publication date: 2021

  Publication venue: The 30th International Joint Conferences on Artificial Intelligence

  Publication keywords: Data stream classification, active learning

  Most Relevant Task: Task T3-4

  Other Relevant Tasks: NA

- **Title:** A supervised feature selection method for mixed-type data using density-based feature clustering

  Author(s): Xuyang Yan, Mrinmoy Sarkar, Biniam Gebru, Shabnam Nazmi, Abdollah Homaifar

  Publication date: 2021

  Publication venue: 2021 IEEE International Conference on Systems, Man, and Cybernetics (SMC)

  Publication keywords: Supervised feature selection, density-based clustering

  Most Relevant Task: Task T3-4

  Other Relevant Tasks: NA

- **Title:** DA2-Net : Diverse & Adaptive Attention Convolutional Neural Network

  Author(s): Abenezer Girma, Abdollah Homaifar, M Nabil Mahmoud, Xuyang Yan, Mrinmoy Sarkar

  Publication date: 2021

  Publication venue: 2021 IEEE International Conference on Systems, Man, and Cybernetics (SMC)

  Publication keywords: Diverse features, diverse and adaptive attention convolutional neural network

  Most Relevant Task: Task T3-4

  Other Relevant Tasks: NA

- **Title:** Multi-label classification with local pairwise and high-order label correlations using graph partitioning

  Author(s): Shabnam Nazmi, Xuyang Yan, Abdollah Homaifar, Mohd Anwar

  Publication date: 2021

  Publication venue: Knowledge-Based Systems

  Publication keywords: Multi-label classification, graph partitioning, density-based clustering

  Most Relevant Task: Task T3-4

  Other Relevant Tasks: NA

- **Title:** Mitigating shortage of labeled data using clustering-based active learning with diversity exploration

  Author(s): Xuyang Yan, Shabnam Nazmi, Biniam Gebru, Mohd Anwar, Abdollah Homaifar, Mrinmoy Sarkar, Kishor Datta Gupta

  Publication date: 2022

  Publication venue: 2022 International Conference on Machine Learning (ICML) Workshop on Adpative Experimental Design and Active Learning in the Real World

  Publication keywords: Active learning, clustering

  Most Relevant Task: Task T3-4

  Other Relevant Tasks: NA

- **Title:** An Online Unsupervised Streaming Features Selection Through Dynamic Feature Clustering

  Author(s): Xuyang Yan, Abdollah Homaifar, Mrinmoy Sarkar, Benjamin Lartey, Kishor Datta Gupta

  Publication date: 2022

  Publication venue: IEEE Transactions on Artificial Intelligence

  Publication keywords: Streaming feature, dynamic feature clustering

  Most Relevant Task: Task T3-4

Other Relevant Tasks: NA

- **Title:** A clustering-based active learning method to query informative and representative samples

  Author(s): Xuyang Yan, Shabnam Nazmi, Biniam Gebru, Mohd Anwar, Abdollah Homaifar, Mrinmoy Sarkar, Kishor Datta Gupta

  Publication date: 2022

  Publication venue: Applied Intelligence

  Publication keywords: Active learning, bi-cluster boundary selection

  Most Relevant Task: Task T3-4

  Other Relevant Tasks: NA

- **Title:** Multi-Attribute Decision Fusion for Pattern Classification

  Author(s): Gabriel Awogbami, Norbert Agana, Abdollah Homaifar

  Publication date: 2018

  Publication venue: IEEE, South East Conference

  Publication keywords: Multi attribute decision fusion, multi-criteria decision analysis, pattern classification

  Most Relevant Task: Task T3-5

  Other Relevant Tasks: NA

- **Title:** A new combination rule based on the average belief function

  Author(s): Gabriel Awogbami, Norbert Agana, Shabnam Nazmi, Abdollah Homaifar

  Publication date: 2018

  Publication venue: IEEE, South East Conference

  Publication keywords: Uncertainty, Dempster-Schafer rule if combination, belief function

  Most Relevant Task: Task T3-5

  Other Relevant Tasks: NA

- **Title:** An evidence theory based multi sensor data fusion for multiclass classification

  Author(s): Gabriel Awogbami, Norbert Agana, Shabnam Nazmi, Xuyang Yan, Abdollah Homaifar

  Publication date: 2018

  Publication venue: IEEE, SMC

  Publication keywords: Multi-sensor data fusion, uncertainty, Dempster-Schafer theory of evidence, multi-class classification

  Most Relevant Task: Task T3-5

  Other Relevant Tasks: NA

- **Title:** A reliability-based multisensor data fusion with application in target classification

Author(s): Gabriel Awogbami, Abdollah Homaifar

Publication date: 2020

Publication venue: Sensors

Publication keywords: multisensor; reliability; classification; belief function; evidence theory; data fusion

Most Relevant Task: Task T3-5

Other Relevant Tasks: NA

- **Title:** A Review on Human–Machine Trust Evaluation: Human-Centric and Machine-Centric Perspectives

  Author(s): Biniam Gebru, Lydia Zeleke, Daniel Blankson, Mahmoud Nabil, Shamila Nateghi, Abdollah Homaifar, Edward Tunstel

  Publication date: 2022

  Publication venue: IEEE Transactions on Human-Machine Systems

  Publication keywords: human-machine trust, trust measurement, trust calibration, machine trustworthiness

  Most Relevant Task: Task T3-5

  Other Relevant Tasks: NA

- **Title:** PIE: A Tool for Data-driven Autonomous UAV Flight Testing

  Author(s): Mrinmoy Sarkar, Abdollah Homaifar, Berat A Erol, Mohammadreza Behniapoor, and Edward Tunstel

  Publication date: 04 September 2019

  Publication venue: Journal of Intelligent & Robotic Systems

  Publication keywords: UAV; Decision tree; Test & evaluation;Data-driven autonomous flight and behavior analysis;AR drone;3DR IRIS;SOLO;Intel RTF drone;ROS;Gazebo;VICON

  Most Relevant Task: Task T3-6

  Other Relevant Tasks: NA

- **Title:** A Novel Search and Survey Technique for Unmanned Aerial Systems in Detecting and Estimating the Area for Wildfires

  Author(s): Mrinmoy Sarkar, Xuyang Yan, Berat A. Erol, Ioannis Raptis, and Abdollah Homaifar

  Publication date: 01 November 2021

  Publication venue: Robotics and Autonomous Systems

  Publication keywords: UAV; Multi-agent autonomous system; AMASE; Search & Survey; Collaborative operation; Robotics;

  Most Relevant Task: Task T3-6

  Other Relevant Tasks: NA

- **Title:** A Framework for Testing and Evaluation of Operational Performance of Multi-UAV Systems

  Author(s): Mrinmoy Sarkar, Xuyang Yan, Shamila Nateghi, Bruce Holmes, Kyriakos Vamvoudakis, and Abdollah Homaifar

  Publication date: 27 September 2022

  Publication venue: Intelligent Systems Conference (IntelliSys) 2021

  Publication keywords: Test and Evaluation; Multi-UAV Testing; Autonomous Behavioral Testing; Cognitive Systems; Physical Flight Testing; Perception; Deep Learning; Bi- LSTM; ROS;

  Most Relevant Task: Task T3-6

  Other Relevant Tasks: NA

- **Title:** Modeling age differences in effects of pair repetition and proactive interference using a single parameter

  Authors: Stephens, J.D.W., Overman, A.A.

  Publication date: 2018

  Publication venue: Psychology and Aging

  Publication keywords: associative deficit, proactive interference, age-related memory impairment, REM model

  Most Relevant Task: Task T3-7

  Other Relevant Tasks: NA

- **Title:** Decision making using automated estimates in the classification of novel stimuli

  Authors: Hoenig, A., & Stephens, J. D. W.

  Publication date: 2020

  Publication venue: AHFE 2019: Advances in Human Factors in Robots and Unmanned Systems

  Book Series : Advances in Intelligent Systems and Computing, vol. 962.

  Editor : Chen, J.

  Publication keywords: Human factors, human-systems integration, unmanned vehicles, decision making, representation of uncertainty

  Most Relevant Task: Task T3-7

  Other Relevant Tasks: NA

- **Title:** Neural Investigation of Human Trust using Electroencephalogram

  Author(s): Provide Oh, S. & Seong, Y

  Publication date: 2019 July

  Publication venue: International Advanced Research Journal in Science, Engineering and Technology

Publication keywords: human trust in automation, brain computer interface, eeg

Most Relevant Task: Task T3-8

Other Relevant Tasks: Task T3-9

- **Title:** Neurological measurement of human trust in automation using electroencephalogram

  Author(s): Oh, S., Seong, Y., Yi, S., & Park, S.

  Publication date: Provide Publication date

  Publication venue: International Journal of Fuzzy Logic and Intelligent Systems

  Publication keywords: EEG, human trust

  Most Relevant Task: Task T3-8

  Other Relevant Tasks: Task T3-9

- **Title:** Preliminary study on neural correspondence of human trust

  Author(s): Oh, S., Seong, Y., & Yi, S.

  Publication date: 2017

  Publication venue: Human Factors & Ergonomics Society Annual Meeting

  Publication keywords: human trust, eeg

  Most Relevant Task: Task T3-8

  Other Relevant Tasks: Task T3-9

- **Title:** Provide Title

  Author(s): Provide Author(s)

  Publication date: Provide Publication date

  Publication venue: Provide Meeting Location

  Publication keywords: Provide Publication keywords

  Most Relevant Task: Task T3-9

  Other Relevant Tasks: NA or list the tasks that are relevant to this publication it is related to multiple tasks.

- **Title:** Provide Title

  Author(s): Provide Author(s)

  Publication date: Provide Publication date

  Publication venue: Provide Meeting Location

  Publication keywords: Provide Publication keywords

  Most Relevant Task: Task T3-9

  Other Relevant Tasks: NA or list the tasks that are relevant to this publication it is related to multiple tasks.

4.2 Meetings and Presentations

16 journal and conference papers (1 accepted journal paper, 3 accepted conference papers, 2 submitted journal papers, and 10 submitted conference papers) were published and submitted as listed below:

- **Meeting Name:** Provide meeting name

  Meeting Purpose: Provide Meeting Purpose

  Meeting Start and End Dates: Provide Meeting Start and End Dates

  Meeting Location: Provide Meeting Location

  Meeting Attendees from this project: Provide Meeting Attendees from this project

  Presentations Made: Provide Yes/No

  Most Relevant Task: Task T1-5

  Other Relevant Tasks: NA or list the tasks that are relevant to this presentation in case it is related to multiple tasks.

- **Meeting Name:** Provide meeting name

  Meeting Purpose: Provide Meeting Purpose

  Meeting Start and End Dates: Provide Meeting Start and End Dates

  Meeting Location: Provide Meeting Location

  Meeting Attendees from this project: Provide Meeting Attendees from this project

  Presentations Made: Provide Yes/No

  Most Relevant Task: Task T1-5

  Other Relevant Tasks: NA or list the tasks that are relevant to this presentation in case it is related to multiple tasks.

- **Meeting Name:** Georgia Tech 2019 Southeast Controls Conference

  Meeting Purpose: Technological Information Sharing

  Meeting Start and End Dates: November 14-15, 2019

  Meeting Location: Atlanta, Georgia - Tech Square Research Building, Georgia Tech

  Meeting Attendees from this project: Faculty and students from Southeastern Universities

  Presentations Made: Yes - Ira Wendell Bates, II

  Most Relevant Task: Task T2-1

  Other Relevant Tasks: NA.

- **Meeting Name:** 2016 IEEE American Control Conference

  Meeting Purpose: Presenting the paper “Achieving Fault-tolerance and Safety of Discrete-event Systems through Learning”

  Meeting Start and End Dates: July 6 through 8, 2016

Meeting Location: Boston, US

Meeting Attendees from this project: Dr. Karimoddini

Presentations Made: Yes

Most Relevant Task: Task T2-2

Other Relevant Tasks: NA

- **Meeting Name:** 61st IEEE International Midwest Symposium on Circuits and Systems

  Meeting Purpose: Presenting the paper "A Survey on Fault-Tolerant Supervisory Control"

  Meeting Start and End Dates Aug4-8, 2018

  Meeting Location: Windsor, ON, Canada

  Meeting Attendees from this project: Drs. Karimoddini and Homaifar

  Presentations Made: Yes

  Most Relevant Task: Task T2-2

  Other Relevant Tasks: NA

- **Meeting Name:** Provide meeting name

  Meeting Purpose: Provide Meeting Purpose

  Meeting Start and End Dates: Provide Meeting Start and End Dates

  Meeting Location: Provide Meeting Location

  Meeting Attendees from this project: Provide Meeting Attendees from this project

  Presentations Made: Provide Yes/No

  Most Relevant Task: Task T2-4

  Other Relevant Tasks: NA or list the tasks that are relevant to this presentation in case it is related to multiple tasks.

- **Meeting Name:** Provide meeting name

  Meeting Purpose: Provide Meeting Purpose

  Meeting Start and End Dates: Provide Meeting Start and End Dates

  Meeting Location: Provide Meeting Location

  Meeting Attendees from this project: Provide Meeting Attendees from this project

  Presentations Made: Provide Yes/No

  Most Relevant Task: Task T2-4

  Other Relevant Tasks: NA or list the tasks that are relevant to this presentation in case it is related to multiple tasks.

- **Meeting Name:** Provide meeting name

  Meeting Purpose: Provide Meeting Purpose

  Meeting Start and End Dates: Provide Meeting Start and End Dates

Meeting Location: Provide Meeting Location

Meeting Attendees from this project: Provide Meeting Attendees from this project

Presentations Made: Provide Yes/No

Most Relevant Task: Task T2-5

Other Relevant Tasks: NA or list the tasks that are relevant to this presentation in case it is related to multiple tasks.

- **Meeting Name:** Provide meeting name

  Meeting Purpose: Provide Meeting Purpose

  Meeting Start and End Dates: Provide Meeting Start and End Dates

  Meeting Location: Provide Meeting Location

  Meeting Attendees from this project: Provide Meeting Attendees from this project

  Presentations Made: Provide Yes/No

  Most Relevant Task: Task T2-5

  Other Relevant Tasks: NA or list the tasks that are relevant to this presentation in case it is related to multiple tasks.

- **Meeting Name:** TECHLAV bi-weekly Seminar Series

  Meeting Purpose To enhance collaboration between project participants and share updates of research results.

  Meeting Start and End Dates April 4, 2016

  Meeting Location North Carolina A&T State University, Greensboro, NC

  Meeting Attendees from this project Students and faculty that are participating in the TECHLAV project.

  Presentations Made Yes

- **Meeting Name:** Master Thesis Defense of Myrielle Allen-Prince

  Meeting Purpose To present the Thesis and receive evaluation from the Thesis Committee.

  Meeting Start and End Dates March 28, 2016

  Meeting Location North Carolina A&T State University, Greensboro, NC

  Meeting Attendees from this project Students and faculty of Mechanical Engineering.

  Presentations Made Yes

- **Meeting Name:** TECHLAV bi-weekly Seminar Series

  Meeting Purpose To enhance collaboration between project participants and share updates of research results.

  Meeting Start and End Dates November 13, 2015

  Meeting Location North Carolina A&T State University, Greensboro, NC

Meeting Attendees from this project Students and faculty that are participating in the TECHLAV project.

Presentations Made Yes

- **Meeting Name:** 5th Annual COE Graduate Poster Presentation Competition

  Meeting Purpose Competition for College of Engineering's graduate students

  Meeting Start and End Dates April 21, 2016

  Meeting Location North Carolina A&T State University

  Meeting Attendees from this project Faculty and students

  Presentations Made Yes

- **Meeting Name:** 2016 International Symposium on Flexible Automation

  Meeting Purpose Presentations of recent research results on dynamical systems and robotics.

  Meeting Start and End Dates August 1-3, 2016

  Meeting Location Case Western Reserve University, OH, USA

  Meeting Attendees from this project Faculty, researchers and students from US and others.

  Presentations Made Yes

- **Meeting Name:** TECHLAV bi-weekly Seminar Series

  Meeting Purpose To enhance collaboration between project participants and share updates of research results.

  Meeting Start and End Dates October 14, 2016

  Meeting Location North Carolina A&T State University, NC, USA

  Meeting Attendees from this project Students and faculty that are participating in the TECHLAV project.

  Presentations Made Yes

- **Meeting Name:** ACIT Seminar

  Meeting Purpose To enhance collaboration between project participants and share updates of research results.

  Meeting Start and End Dates April 28, 2017

  Meeting Location North Carolina A&T State University, Greensboro, NC, USA

  Meeting Attendees from this project Students and faculty that are participating in the TECHLAV project.

  Presentations Made Yes

- **Meeting Name:** 2nd TECHLAV Annual Meeting

  Meeting Purpose To improve and excel to ensure continued success for TECHLAV and its many contributions to Autonomy.

Meeting Start and End Dates May 30-31, 2017

Meeting Location North Carolina A&T State University, Greensboro, NC, USA

Meeting Attendees from this project Researchers and administration members of the TECHLAV project.

Presentations Made Yes

- **Meeting Name:** National Society of Black Engineers 44th Annual Convention

  Meeting Purpose Communities of color and all Black students can envision themselves as engineers

  Meeting Start and End Dates March 21 - 25, 2018

  Meeting Location David L. Lawrence Convention Center, Pittsburgh, PA, USA

  Meeting Attendees from this project Black pre-college, collegiate, or professional engineer.

  Presentations Made Yes

- **Meeting Name:** 2018 WM Symposia

  Meeting Purpose Conference on radioactive waste management & disposal, decommissioning, packaging & transportation, facility siting and site remediation

  Meeting Start and End Dates March 18 - 22, 2018

  Meeting Location Phoenix Convention Center, Phoenix, AZ, USA

  Meeting Attendees from this project Black pre-college, collegiate, or professional engineer.

  Presentations Made Yes (best presentation in session)

- **Meeting Name:** TECHLAV's 3rd Annual Meeting

  Meeting Purpose Reports progress during the last year and make a plan for next year

  Meeting Start and End Dates May 31 - June 1, 2018

  Meeting Location NCAT, Greensboro, NC, USA

  Meeting Attendees from this project All Techlav project participants and evaluators.

  Presentations Made Yes

- **Meeting Name:** 8th Annual COE Graduate Poster Presentation Competition

  Meeting Purpose Competition for College of Engineering's graduate students

  Meeting Start and End Dates April 24, 2018

  Meeting Location NCAT, Greensboro, NC, USA

  Meeting Attendees from this project Faculty and students

  Presentations Made Yes

- **Meeting Name:** 6th IAJC International Conference

Meeting Purpose To disseminate updates of research results.

Meeting Start and End Dates October 11-14, 2018

Meeting Location Orlando, Florida

Meeting Attendees from this project Faculty and students from academia

Presentations Made Yes

- **Meeting Name:** 2019 Waste Management Symposia

  Meeting Purpose To disseminate updates of research results.

  Meeting Start and End Dates March 2-7, 2019

  Meeting Location Phoenix, Arizona

  Meeting Attendees from this project Faculty and students from academia

  Presentations Made Yes

- **Meeting Name:** TECHLAV's 4th Annual Meeting

  Meeting Purpose Reports progress during the last year and make a plan for next year

  Meeting Start and End Dates June 25 - June 26, 2019

  Meeting Location UTSA, San Antonio, TX, USA

  Meeting Attendees from this project All Techlav project participants and evaluators.

  Presentations Made Yes

- **Meeting Name:** IEEE SoutheastCon 2019

  Meeting Purpose To present original technical work on engineering, science, and technology of current interest

  Meeting Start and End Dates April 11-14, 2019

  Meeting Location Huntsville, AL, USA

  Meeting Attendees from this project Faculty and students

  Presentations Made Yes

- **Meeting Name:** TECHLAV Student Seminar Series

  Meeting Purpose To explore the latest transportation innovations and provides attendees with the chance to network with other professionals.

  Meeting Start and End Dates April 24, 2020

  Meeting Location Virtually on Zoom

  Meeting Attendees from this project TECHLAV Faculty and Students

  Presentations Made Yes

- **Meeting Name:** ACEC/NC NCDOT Joint Transportation Conference

  Meeting Purpose To explore the latest transportation innovations and provides attendees with the chance to network with other professionals.

Meeting Start and End Dates October 28 to 30, 2019

Meeting Location Raleigh Convention Center, Raleigh, NC

Meeting Attendees from this project Universities, NC DOT, and Transportation Companies

Presentations Made Yes

- **Meeting Name:** International Conference on Design and Analysis of Protective Structures 2019

  Meeting Purpose Conference to scientists and engineers who are interested in protective structures.

  Meeting Start and End Dates Dec 4 to 6, 2019

  Meeting Location Seoul, South Korea

  Meeting Attendees from this project Faculty from US, EU, Canada and South Korea.

  Presentations Made Yes

- lhjadhjl

  Meeting Name: 12th Annual IEEE SoSE Conference

  Meeting Purpose: An IoT self organizing network for 5G dense network interference alignment

  Meeting Start and End Dates: June 18-21, 2017

  Meeting Location: Waikoloa Hilton Village, Waikoloa, Hawaii, USA

  Meeting Attendees from this project: Brian Kelley

  Presentations Made: Yes

  Most Relevant Task: Task T2-7

  Other Relevant Tasks: NA

- **Meeting Name:** UEMCON New York

  Meeting Purpose: Design of K-user massive MIMO networks

  Meeting Start and End Dates: Oct 19-21, 2017

  Meeting Location: Columbia University, New York City, New York, USA

  Meeting Attendees from this project: Anil Kumar Yerrapragada

  Presentations Made: Yes

  Most Relevant Task: Task T2-7

  Other Relevant Tasks: NA

- **Meeting Name:** 2019 Annual IEEE SoSE Conference

  Meeting Purpose: Error Rate Analysis Of Physical Layer Security For Sub-6 GHz 5G Network Planning

  Meeting Start and End Dates: May 19-22, 2019

Meeting Location: Sheraton Hotel, Anchorage, Alaska, USA
Meeting Attendees from this project: Anil Kumar Yerrapragada
Presentations Made: Yes
Most Relevant Task: Task T2-7
Other Relevant Tasks: NA

- **Meeting Name:** MILCOM Conference Norfolk
  Meeting Purpose: Provide Meeting Purpose
  Meeting Start and End Dates: November 12-14, 2019
  Meeting Location: Hilton Norfolk the Main, Norfolk, Virginia, USA
  Meeting Attendees from this project: Anil Kumar Yerrapragada
  Presentations Made: Yes
  Most Relevant Task: Task T2-7
  Other Relevant Tasks: NA

- **Meeting Name:** TECHLAV 2nd Annual Meeting
  Meeting Purpose: Self Organizing Networks that Learn
  Meeting Start and End Dates: May 31 - July 1, 2017
  Meeting Location: North Carolina A&T University, Greensboro, North Carolina, USA
  Meeting Attendees from this project: Brian Kelley, Taylor Eisman
  Presentations Made: Yes
  Most Relevant Task: Task T2-7
  Other Relevant Tasks: NA

- **Meeting Name:** Provide meeting name
  Meeting Purpose: Provide Meeting Purpose
  Meeting Start and End Dates: Provide Meeting Start and End Dates
  Meeting Location: Provide Meeting Location
  Meeting Attendees from this project: Provide Meeting Attendees from this project
  Presentations Made: Provide Yes/No
  Most Relevant Task: Task T3-1
  Other Relevant Tasks: NA or list the tasks that are relevant to this presentation in case it is related to multiple tasks.

- **Meeting Name:** Provide meeting name
  Meeting Purpose: Provide Meeting Purpose
  Meeting Start and End Dates: Provide Meeting Start and End Dates
  Meeting Location: Provide Meeting Location

Meeting Attendees from this project: Provide Meeting Attendees from this project

Presentations Made: Provide Yes/No

Most Relevant Task: Task T3-1

Other Relevant Tasks: NA or list the tasks that are relevant to this presentation in case it is related to multiple tasks.

- **Meeting Name:** TECHLAV YEAR (YEAR 2019) ANNUAL VIRTUAL MEETING & AGENDA

  Meeting Purpose: Poster Presentation

  Meeting Start and End Dates: 25 – 26 June, 2019

  Meeting Location:

  Meeting Attendees from this project: TECHLAV members

  Presentations Made: Yes (Poster)

  Most Relevant Task: Task T3-3

  Other Relevant Tasks: NA

- **Meeting Name:** IEEE 2020 International Conference on Systems, Man, and Cybernetics (SMC)

  Meeting Purpose: Conference Presentation

  Meeting Start and End Dates: 06 – 09 Oct., 2019

  Meeting Location: Bari, Italy

  Meeting Attendees from this project: Dr. Abdollah Homaifar

  Presentations Made: Yes

  Most Relevant Task: Task T3-3

  Other Relevant Tasks: NA

- **Meeting Name:** TECHLAV (YEAR 2020) ANNUAL VIRTUAL MEETING & AGENDA

  Meeting Purpose: To provide a glimpse of the research outcome

  Meeting Start and End Dates: 25 June, 2020

  Meeting Location: VIRTUAL MEETING, WebEx

  Meeting Attendees from this project: TECHLAV members

  Presentations Made: No

  Most Relevant Task: Task T3-3

  Other Relevant Tasks: NA

- **Meeting Name:** IEEE 2020 International Conference on Systems, Man, and Cybernetics (SMC)

  Meeting Purpose: Conference Presentation (Virtual)

  Meeting Start and End Dates: 11 – 14 Oct., 2020

  Meeting Location: Toronto, ON, Canada

Meeting Attendees from this project: Md Khurram Monir Rabby (self)

Presentations Made: Yes

Most Relevant Task: Task T3-3

Other Relevant Tasks: NA

- **Meeting Name:** Provide meeting name

  Meeting Purpose: Provide Meeting Purpose

  Meeting Start and End Dates: Provide Meeting Start and End Dates

  Meeting Location: Provide Meeting Location

  Meeting Attendees from this project: Provide Meeting Attendees from this project

  Presentations Made: Provide Yes/No

  Most Relevant Task: Task T3-4

  Other Relevant Tasks: NA or list the tasks that are relevant to this presentation in case it is related to multiple tasks.

- **Meeting Name:** Provide meeting name

  Meeting Purpose: Provide Meeting Purpose

  Meeting Start and End Dates: Provide Meeting Start and End Dates

  Meeting Location: Provide Meeting Location

  Meeting Attendees from this project: Provide Meeting Attendees from this project

  Presentations Made: Provide Yes/No

  Most Relevant Task: Task T3-4

  Other Relevant Tasks: NA or list the tasks that are relevant to this presentation in case it is related to multiple tasks.

- **Meeting Name:** Provide meeting name

  Meeting Purpose: Provide Meeting Purpose

  Meeting Start and End Dates: Provide Meeting Start and End Dates

  Meeting Location: Provide Meeting Location

  Meeting Attendees from this project: Provide Meeting Attendees from this project

  Presentations Made: Provide Yes/No

  Most Relevant Task: Task T3-5

  Other Relevant Tasks: NA or list the tasks that are relevant to this presentation in case it is related to multiple tasks.

- **Meeting Name:** Provide meeting name

  Meeting Purpose: Provide Meeting Purpose

  Meeting Start and End Dates: Provide Meeting Start and End Dates

  Meeting Location: Provide Meeting Location

Meeting Attendees from this project: Provide Meeting Attendees from this project

Presentations Made: Provide Yes/No

Most Relevant Task: Task T3-5

Other Relevant Tasks: NA or list the tasks that are relevant to this presentation in case it is related to multiple tasks.

- **Meeting Name:** 2019 Southeast Controls Conference

  Meeting Purpose: To present the research outcome of the project

  Meeting Start and End Dates: November 14-15, 2019

  Meeting Location: Georgia Tech, Schools of Electrical and Computer Engineering, Civil and Environmental Engineering, and Aerospace Engineering.

  Meeting Attendees from this project: Mrinmoy Sarkar

  Presentations Made: Yes

  Most Relevant Task: Task T3-6

  Other Relevant Tasks: NA

- **Meeting Name:** ACIT Seminar Series

  Meeting Purpose: To educate the members of ACIT and TECHLAV in current research

  Meeting Start and End Dates: February 23, 2018

  Meeting Location: North Carolina Agricultural and Technical State University

  Meeting Attendees from this project: Members of TECHLAV

  Presentations Made: Yes

  Most Relevant Task: Task T3-7

  Other Relevant Tasks: NA

- **Meeting Name:** Seventh Annual College of Engineering Research Poster Presentation Competition at NCA&T

  Meeting Purpose Present research work in a forum for the College of Engineering

  Meeting Start and End Dates April 24, 2018

  Meeting Location Alumni Events Center, NCA&T

  Meeting Attendees from this project Ms. Amber Hoenig; TECHLAV students and faculty

  Presentations Made Yes

  Most Relevant Task: Task T3-7

  Other Relevant Tasks: NA

- **Meeting Name:** AFRL Project Supervisor Meeting

  Meeting Purpose Update AFRL Project Supervisor with details of ongoing research

  Meeting Start and End Dates February 22, 2019

Meeting Location Fort IRC, North Carolina Agricultural and Technical State University

Meeting Attendees from this project Provide Meeting Attendees from this project

Presentations Made Yes

Most Relevant Task: Task T3-7

Other Relevant Tasks: NA

- **Meeting Name:** Eighth Annual College of Engineering Research Poster Presentation Competition at NCA&T

  Meeting Purpose Present research work in a forum for the College of Engineering

  Meeting Start and End Dates April 18, 2019

  Meeting Location Alumni Events Center, NCA&T

  Meeting Attendees from this project Ms. Amber Hoenig; TECHLAV students and faculty

  Presentations Made Yes

  Most Relevant Task: Task T3-7

  Other Relevant Tasks: NA

- **Meeting Name:** TECHLAV Fourth Annual Meeting

  Meeting Purpose Present and plan the research of the TECHLAV program; meet with program stakeholders

  Meeting Start and End Dates June 25-26, 2019

  Meeting Location John Peace Library, University of Texas at San Antonio

  Meeting Attendees from this project Dr. Joseph DW Stephens (presenter), Ms. Amber Hoenig, TECHLAV members and stakeholders

  Presentations Made Yes

  Most Relevant Task: Task T3-7

  Other Relevant Tasks: NA

- **Meeting Name:** ACIT Seminar Series

  Meeting Purpose: To educate the members of ACIT and TECHLAV in current research

  Meeting Start and End Dates: October 4, 2019

  Meeting Location: North Carolina Agricultural and Technical State University

  Meeting Attendees from this project: Members of TECHLAV

  Presentations Made: Yes

  Most Relevant Task: Task T3-7

  Other Relevant Tasks: NA

- **Meeting Name:** 9th Annual College of Engineering Graduate Research Poster Presentation Competition

Meeting Purpose To share knowledge from students' research

Meeting Start and End Dates April 8, 2020

Meeting Location North Carolina Agricultural and Technical State University, online

Meeting Attendees from this project Students in the College of Engineering

Presentations Made Yes

Most Relevant Task: Task T3-7

Other Relevant Tasks: NA

- **Meeting Name:** 5th Year Annual TECHLAV Meeting

  Meeting Purpose To share knowledge, research, and progress in TECHLAV with share-holders and members

  Meeting Start and End Dates June 25, 2020

  Meeting Location North Carolina Agricultural and Technical State University, online

  Meeting Attendees from this project Members of TECHLAV

  Presentations Made No

  Most Relevant Task: Task T3-7

  Other Relevant Tasks: NA

- **Meeting Name:** ACIT Seminar Series

  Meeting Purpose: To educate the members of ACIT and TECHLAV in current research

  Meeting Start and End Dates: July 17, 2020

  Meeting Location: North Carolina Agricultural and Technical State University, online

  Meeting Attendees from this project: Members of TECHLAV

  Presentations Made: Yes

  Most Relevant Task: Task T3-7

  Other Relevant Tasks: NA

- **Meeting Name:** Provide meeting name

  Meeting Purpose: Provide Meeting Purpose

  Meeting Start and End Dates: Provide Meeting Start and End Dates

  Meeting Location: Provide Meeting Location

  Meeting Attendees from this project: Provide Meeting Attendees from this project

  Presentations Made: Provide Yes/No

  Most Relevant Task: Task T3-8

  Other Relevant Tasks: NA or list the tasks that are relevant to this presentation in case it is related to multiple tasks.

- **Meeting Name:** Provide meeting name

Meeting Purpose: Provide Meeting Purpose

Meeting Start and End Dates: Provide Meeting Start and End Dates

Meeting Location: Provide Meeting Location

Meeting Attendees from this project: Provide Meeting Attendees from this project

Presentations Made: Provide Yes/No

Most Relevant Task: Task T3-9

Other Relevant Tasks: NA or list the tasks that are relevant to this presentation in case it is related to multiple tasks.

- **Meeting Name:** Provide meeting name

  Meeting Purpose: Provide Meeting Purpose

  Meeting Start and End Dates: Provide Meeting Start and End Dates

  Meeting Location: Provide Meeting Location

  Meeting Attendees from this project: Provide Meeting Attendees from this project

  Presentations Made: Provide Yes/No

  Most Relevant Task: Task T3-9

  Other Relevant Tasks: NA or list the tasks that are relevant to this presentation in case it is related to multiple tasks.

4.3 TECHLAV Seminar Series

TECHLAV Center and its affiliated partners and laboratories are jointly organizing bi-weekly seminar series in different aspects of modeling, control, testing, and evaluation of large-scale systems of autonomous vehicles. The goal is to create this interactive forum as an environment for sharing recent research results and discussing various control, modeling, testing and evaluation problems from different angles by participants with different backgrounds and expertise. During the last reporting period, 4 seminars were delivered which are listed as follows. The presentation content is available to public though the TECHLAV Website at the following URL addresses: "http://techlav.ncat.edu/seminars.html"

4.3.1 Title: Formal Methods in the Development of Highly Assured Software for Unmanned Air- craft Systems

Speaker Dr. Cesar Munoz

Affiliation NASA Langley Research Center

Date 04/09/2021

4.3.2 Title: Copilot

Speaker Dr. Alwyn Goodloe

Affiliation NASA Langley Reseach Center Computer Engineering

Date 04/16/2021

4.3.3 Title: 24.5 GHz airborne sUAS Doppler radar

Speaker Dr. George Szatkowski

Affiliation NASA Langley Safety-Critical Avionics Systems

Date 04/23/2021

4.3.4 Title: Data-Centric Architecture: The foundation to CAV design and performance

Speaker Dr. Bob Leigh

Affiliation Real-Time Innovations (RTI)

Date 04/30/2021

4.3.5 Title: Attention augmented Convolutional Neural Network (CNN) architecture design

Speaker Abenezer Girma

Affiliation North Carolina A&T PhD Candidate

Date 05/07/2021

4.4 Conducted Dissertations and Thesis

16 journal and conference papers (1 accepted journal paper, 3 accepted conference papers, 2 submitted journal papers, and 10 submitted conference papers) were published and submitted as listed below:

- **Thesis Title:** Design, construction and distrubuted control of high degree of freedom modular robot

  Student Name: Ikram Mohammed

  Defense Date: 5/15/2016

  Advisor: Mo Jamshidi

  Link to the Dissertation: https://www-proquest-com.libweb.lib.utsa.edu/dissertations-theses/design-construction-distrubuted-control-high/docview/1834614779/

  Most Relevant Task: Task T1-1

  Other Relevant Tasks: Task T1-3

- **Thesis Title:** Autonomous Intelligent Control for UAV with Self Docking, Charging and Learning Capabilities

  Student Name: Abhijit Majumdar

  Defense Date: 12/11/2018

  Advisor: Mo Jamshidi

  Link to the Thesis: https://www-proquest-com.libweb.lib.utsa.edu/dissertations-theses/autonomous-intelligent-control-uav-with-self/docview/2158349940/

  Most Relevant Task: Task T1-3

  Other Relevant Tasks: Task T1-1, T1-4

- **Thesis Title:** Modeling, Simulation, and Design of a Time-Delayed Multi-Agent System of Autonomous Vehicles

Student Name: Nicolas Gamez

Defense Date: 5/5/2017

Advisor: Mo Jamshidi

Link to the Thesis: https://www-proquest-com.libweb.lib.utsa.edu/dissertations-theses/modeling-simulation-design-time-delayed-multi/docview/1906293552/

Most Relevant Task: Task T1-3, T1-6

Other Relevant Tasks: Task T1-1, T1-2

- **Dissertation Title:** Towards Artificial Emotional Intelligence for Heterogeneous System to Improve Human Robot Interactions

  Student Name: Berat Erol

  Defense Date: 5/15/2018

  Advisor: Mo Jamshidi

  Link to the Thesis: https://www-proquest-com.libweb.lib.utsa.edu/dissertations-theses/towards-artificial-emotional-intelligence/docview/2091369150/

  Most Relevant Task: Task T1-3, T1-4

  Other Relevant Tasks: Task T1-1

- **Dissertation Title:** Environmental Effects of Renewable Energy: A Machine Learning Approach

  Student Name: Yunus Yetis

  Defense Date: 8/12/2020

  Advisor: Mo Jamshidi

  Link to the Dissertation: https://www-proquest-com.libweb.lib.utsa.edu/dissertations-theses/environme effects-renewable-energy-machine/docview/2446028689/

  Most Relevant Task: Task T1-2

  Other Relevant Tasks: NA

- **Thesis Title:** Modeling, Simulation, and Design of a Time-Delayed Multi-Agent System of Autonomous Vehicles

  Student Name: Nicholas Gamez

  Defense Date: 5/5/2017

  Advisor: Mo Jamshidi

  Link to the Thesis: https://www-proquest-com.libweb.lib.utsa.edu/dissertations-theses/modeling-simulation-design-time-delayed-multi/docview/1906293552/

  Most Relevant Task: Task T1-3, T1-6

  Other Relevant Tasks: Task T1-1, T1-2

- **Thesis Title:** Kinematic modeling and control of a human-robot platform for the blind and visually impaired

Student Name: Nicolas Gallardo

Defense Date: 12/10/2016

Advisor: Dr. Patrick Benavidez, Dr. Mo Jamshidi

Link to the Thesis: https://www-proquest-com.libweb.lib.utsa.edu/dissertations-theses/kinematic-modeling-control-human-robot-platform/docview/1873449098/

Most Relevant Task: Task T1-3, T1-6

Other Relevant Tasks: NA

- **Thesis Title:** Autonomous Intelligent Control for UAV with Self Docking, Charging and Learning Capabilities

  Student Name: Abhijit Majumdar

  Defense Date: 12/11/2018

  Advisor: Dr. Patrick Benavidez, Dr. Mo Jamshidi

  Link to the Thesis: https://www-proquest-com.libweb.lib.utsa.edu/dissertations-theses/autonomous-intelligent-control-uav-with-self/docview/2158349940/

  Most Relevant Task: Task T1-3, T1-4

  Other Relevant Tasks: Task T1-1

- **Thesis Title:** Modeling, Simulation, and Design of a Time-Delayed Multi-Agent System of Autonomous Vehicles

  Student Name: Nicholas Gamez

  Defense Date: 5/5/2017

  Advisor: Dr. Patrick Benavidez, Dr. Mo Jamshidi

  Link to the Thesis: https://www-proquest-com.libweb.lib.utsa.edu/dissertations-theses/modeling-simulation-design-time-delayed-multi/docview/1906293552/

  Most Relevant Task: Task T1-3, T1-6

  Other Relevant Tasks: Task T1-1, T1-2

- **Thesis Title:** Reinforcement Learning Control with Genetic Algorithm and Physical UGV Applications

  Student Name: Edgar Perez

  Defense Date: 11/14/2019

  Advisor: Dr. Patrick Benavidez, Dr. Mo Jamshidi

  Link to the Thesis: https://www-proquest-com.libweb.lib.utsa.edu/dissertations-theses/reinforcement-learning-control-with-genetic/docview/2351548613/

  Most Relevant Task: Task T1-3

  Other Relevant Tasks: NA

- **Thesis Title:** 3D-printed Mobile Assistance Platform (MAP) for rehabilitative robotics

  Student Name: Eric Wineman

Defense Date: 12/15/2015

Advisor: Dr. Patrick Benavidez, Dr. Mo Jamshidi

Link to the Thesis: https://www-proquest-com.libweb.lib.utsa.edu/dissertations-theses/3d-printed-mobile-assistance-platform-map/docview/1751135589/

Most Relevant Task: Task T1-3

Other Relevant Tasks: NA

- **Dissertation Title:** Towards Artificial Emotional Intelligence for Heterogeneous System to Improve Human Robot Interactions

  Student Name: Berat Erol

  Defense Date: 5/15/2018

  Advisor: Mo Jamshidi

  Link to the Thesis: https://www-proquest-com.libweb.lib.utsa.edu/dissertations-theses/towards-artificial-emotional-intelligence/docview/2091369150/

  Most Relevant Task: T1-4

  Other Relevant Tasks: T1-1, T1-3

- **Dissertation Title:** Autonomous Intelligent Control for UAV with Self Docking, Charging and Learning Capabilities

  Student Name: Abhijit Majumdar

  Defense Date: 12/11/2018

  Advisor: Mo Jamshidi

  Link to the Thesis: https://www-proquest-com.libweb.lib.utsa.edu/dissertations-theses/autonomous-intelligent-control-uav-with-self/docview/2158349940/

  Most Relevant Task: T1-3, T1-4

  Other Relevant Tasks: T1-1

- **Dissertation Title:** Comparative Performance Analysis of Navigation Algorithm and Deep Learning Application: Different Infrastructure and Cloud Robotics

  Student Name: Divya Bhaskaran

  Defense Date: 12/12/2017

  Advisor: Mo Jamshidi

  Link to the Thesis: https://www-proquest-com.libweb.lib.utsa.edu/dissertations-theses/comparative-performance-analysis-navigation/docview/2001602315

  Most Relevant Task: T1-6, T1-4

  Other Relevant Tasks:

- **Dissertation Title:** Vision based cloud robotics

  Student Name: Mohan Muppidi Kumar

  Defense Date: 12/5/2014

Advisor: Mo Jamshidi

Link to the Thesis: https://www-proquest-com.libweb.lib.utsa.edu/dissertations-theses/vision-based-cloud-robotics/docview/1682256336/

Most Relevant Task: T1-4

Other Relevant Tasks:

- **Dissertation Title:** Optimal navigation of autonomous vehicles

  Student Name: Ibrahim Mohammed

  Defense Date: 10/10/2016

  Advisor: Mo Jamshidi

  Link to the Thesis: https://www-proquest-com.libweb.lib.utsa.edu/dissertations-theses/optimal-navigation-autonomous-vehicles/docview/1835089910/

  Most Relevant Task: T1-4

  Other Relevant Tasks:

- **Dissertation Title:** Cooperative mapping and self-localization for multiple quadrocopters

  Student Name: Satish Vaishnav

  Defense Date: 7/14/2015

  Advisor: Mo Jamshidi

  Link to the Thesis: https://www-proquest-com.libweb.lib.utsa.edu/dissertations-theses/cooperative-mapping-self-localization-multiple/docview/1728065326/

  Most Relevant Task: T1-4

  Other Relevant Tasks:

- **Dissertation Title:** A Complete, Automated and Scalable Framework for Science and Engineering

  Student Name: Mevlut Demir

  Defense Date: 5/16/2020

  Advisor: Mo Jamshidi

  Link to the Thesis: https://www-proquest-com.libweb.lib.utsa.edu/dissertations-theses/complete-automated-scalable-framework-science/docview/2444667555/

  Most Relevant Task: T1-6, T1-4

  Other Relevant Tasks:

- **Dissertation Title:** Low-cost home multi-robot rehabilitation system for the disabled population

  Student Name: Patrick Benavdez

  Defense Date: 7/20/2015

  Advisor: Mo Jamshidi

Link to the Thesis: https://www-proquest-com.libweb.lib.utsa.edu/dissertations-theses/low-cost-home-multi-robot-rehabilitation-system/docview/1719264656/

Most Relevant Task: T1-4

Other Relevant Tasks:

- **Dissertation Title:** Reactive Symbolic Planning and Control of Autonomous Vehicles in Adversarial Environments

  Student Name: Laya Shamgah

  Defense Date: May 24, 2019

  Advisor: Dr. Ali Karimoddini

  Most Relevant Task: Task T1-5

- **Dissertation Title:** End to End Control of a Cloud-Based Heterogeneous Swarm for Cyber-Physical Systems

  Student Name: Jonathan Lwowski

  Defense Date: 5/17/2019

  Advisor: Mo Jamshidi

  Link to the Thesis: https://www-proquest-com.libweb.lib.utsa.edu/dissertations-theses/end-control-cloud-based-heterogeneous-swarm-cyber/docview/2235967055

  Most Relevant Task: T1-6, T2-2

  Other Relevant Tasks:

- **Dissertation Title:** Real-Time Adaptive Data-Driven Perception for Anomaly Priority Scoring at Scale

  Student Name: Ali Miraftabzadeh

  Defense Date: 12/15/2017

  Advisor: Mo Jamshidi

  Link to the Thesis: https://www-proquest-com.libweb.lib.utsa.edu/dissertations-theses/real-time-adaptive-data-driven-perception-anomaly/docview/1981389850

  Most Relevant Task: T1-6

  Other Relevant Tasks:

- **Thesis Title:** Kinematic modeling and control of a human-robot platform for the blind and visually impaired

  Student Name: Nicolas Gallardo

  Defense Date: 12/10/2016

  Advisor: Mo Jamshidi

  Link to the Thesis: https://www-proquest-com.libweb.lib.utsa.edu/dissertations-theses/kinematic-modeling-control-human-robot-platform/docview/1873449098/

  Most Relevant Task: T1-3, T1-6

Other Relevant Tasks:

- **Thesis Title:** Modeling, Simulation, and Design of a Time-Delayed Multi-Agent System of Autonomous Vehicles

  Student Name: Nicholas Gamez

  Defense Date: 5/5/2017

  Advisor: Mo Jamshidi

  Link to the Thesis: https://www-proquest-com.libweb.lib.utsa.edu/dissertations-theses/modeling-simulation-design-time-delayed-multi/docview/1906293552/

  Most Relevant Task: T1-3, T1-6

  Other Relevant Tasks: T1-1, T1-2

- **Thesis Title:** Comparative Performance Analysis of Navigation Algorithm and Deep Learning Application: Different Infrastructure and Cloud Robotics

  Student Name: Divya Bhaskaran

  Defense Date: 12/12/2017

  Advisor: Mo Jamshidi

  Link to the Thesis: https://www-proquest-com.libweb.lib.utsa.edu/dissertations-theses/comparative-performance-analysis-navigation/docview/2001602315

  Most Relevant Task: T1-6, T1-4

  Other Relevant Tasks:

- **Thesis Title:** Implementation and Simulation of a Mobile Sensor Network Using Lambda Architecture

  Student Name: Sean Ackels

  Defense Date: 4/5/2020

  Advisor: Mo Jamshidi

  Link to the Thesis: https://www-proquest-com.libweb.lib.utsa.edu/dissertations-theses/implementation-simulation-mobile-sensor-network/docview/2423624605/

  Most Relevant Task: T1-6

  Other Relevant Tasks:

- **Thesis Title:** Mitigating Time-Delay Problems with Cloud-Based Networked Controlled UGVs

  Student Name: James Nelson

  Defense Date: 3/20/2020

  Advisor: Mo Jamshidi

  Link to the Thesis: https://www-proquest-com.libweb.lib.utsa.edu/dissertations-theses/mitigating-time-delay-problems-with-cloud-based/docview/2415835721

  Most Relevant Task: T1-6

  Other Relevant Tasks: T1-4

- **Dissertation Title:** A Complete, Automated and Scalable Framework for Science and Engineering

  Student Name: Mevlut Demir

  Defense Date: 5/16/2020

  Advisor: Mo Jamshidi

  Link to the Thesis: https://www-proquest-com.libweb.lib.utsa.edu/dissertations-theses/complete-automated-scalable-framework-science/docview/2444667555/

  Most Relevant Task: T1-6, T1-4

  Other Relevant Tasks:

- **Thesis Title:** Trash Collecting Robotic System Using Two Autonomous, Mobile-Manipulator Robots with Convolutional Neural Network Object Detection System

  Student Name: Jacob Hudson

  Defense Date: 12/15/2021

  Advisor: Mo Jamshidi

  Link to the Thesis: https://www-proquest-com.libweb.lib.utsa.edu/pqdtlocal1006280/docview/261513967

  Most Relevant Task: T1-6

  Other Relevant Tasks:

- **Thesis Title:** Terrain Traversability Analysis for Off-Road Robots: Emphasis on Semantic Segmentation And Terrain Classification with Terrain Assessment

  Student Name: Manjari Gummidipundi

  Defense Date: 12/15/2021

  Advisor: Mo Jamshidi

  Link to the Thesis: https://www-proquest-com.libweb.lib.utsa.edu/pqdtlocal1006280/docview/261513968

  Most Relevant Task: T1-6

  Other Relevant Tasks:

- **Project Title:**

  Student Name: Karthik Pai Haradi

  Defense Date: 12/20/2016

  Advisor: Mo Jamshidi

  Link to the Thesis:

  Most Relevant Task: T1-6

  Other Relevant Tasks:

- **Title:** Research based PhD

  Student Name: Benjamin Champion

  Defense Date: 11/17/2018

Advisor: Mo Jamshidi

Link to the Thesis:

Most Relevant Task: T1-6

Other Relevant Tasks:

- **Thesis Title:** Design, construction and distrubuted control of high degree of freedom modular robot

  Student Name: Ikram Mohammed

  Defense Date: 5/15/2016

  Advisor: Mo Jamshidi

  Link to the Dissertation: https://www-proquest-com.libweb.lib.utsa.edu/dissertations-theses/design-construction-distrubuted-control-high/docview/1834614779/

  Most Relevant Task: Task T1-1

  Other Relevant Tasks: Task T1-3

- **Thesis Title:** Autonomous Intelligent Control for UAV with Self Docking, Charging and Learning Capabilities

  Student Name: Abhijit Majumdar

  Defense Date: 12/11/2018

  Advisor: Mo Jamshidi

  Link to the Thesis: https://www-proquest-com.libweb.lib.utsa.edu/dissertations-theses/autonomous-intelligent-control-uav-with-self/docview/2158349940/

  Most Relevant Task: Task T1-3

  Other Relevant Tasks: Task T1-1, T1-4

- **Thesis Title:** Modeling, Simulation, and Design of a Time-Delayed Multi-Agent System of Autonomous Vehicles

  Student Name: Nicolas Gamez

  Defense Date: 5/5/2017

  Advisor: Mo Jamshidi

  Link to the Thesis: https://www-proquest-com.libweb.lib.utsa.edu/dissertations-theses/modeling-simulation-design-time-delayed-multi/docview/1906293552/

  Most Relevant Task: Task T1-3, T1-6

  Other Relevant Tasks: Task T1-1, T1-2

- **Dissertation Title:** Towards Artificial Emotional Intelligence for Heterogeneous System to Improve Human Robot Interactions

  Student Name: Berat Erol

  Defense Date: 5/15/2018

Advisor: Mo Jamshidi

Link to the Thesis: https://www-proquest-com.libweb.lib.utsa.edu/dissertations-theses/towards-artificial-emotional-intelligence/docview/2091369150/

Most Relevant Task: Task T1-3, T1-4

Other Relevant Tasks: Task T1-1

- **Dissertation Title:** Failure Diagnosis of Discrete Event Systems Under Unknown\Uncertain Activation Conditions

  Student Name: Alejandro White

  Defense Date: March 19, 2018

  Advisor: Dr. Ali Karimoddini

  Most Relevant Task: Task T2-1

  Other Relevant Tasks: NA

- **Thesis Title:** Developing a Novel Active-Learning Technique for Fault Diagnosis of Unknown Systems

  Student Name: Ira Wendell Bates, II

  Defense Date: October 21, 2020

  Advisor: Dr. Ali Karimoddini

  Most Relevant Task: Task T2-1

  Other Relevant Tasks: NA

- **Dissertation Title:** Heterogeneous Data Analytics for Fault Diagnosis and Accommodation of a Mobile Robot with Dislocated Actuator Faults

  Student Name: Conor Wallace

  Defense Date: 4/23/2020

  Advisor: Dr. Mo Jamshidi

  Link to the Dissertation: https://www-proquest-com.libweb.lib.utsa.edu/dissertations-theses/heterogeneous-data-analytics-fault-diagnosis/docview/2415841786/

  Most Relevant Task: Task T2-2

  Other Relevant Tasks: NA

- **Dissertation Title:** End to End Control of a Cloud-Based Heterogeneous Swarm for Cyber-Physical Systems

  Student Name: Jonathan Lwowski

  Defense Date: 5/17/2019

  Advisor: Dr. Mo Jamshidi

  Link to the Dissertation: https://www-proquest-com.libweb.lib.utsa.edu/dissertations-theses/end-control-cloud-based-heterogeneous-swarm-cyber/docview/2235967055/

Most Relevant Task: Task T1-6

Other Relevant Tasks: Task T2-2

- **Dissertation Title:** Provide Title

  Student Name: Provide Student Name

  Defense Date: Provide Defense Date

  Advisor: Provide Advisor Name

  Link to the Dissertation: Provide Dissertation Link if available (if not delete this item

  Most Relevant Task: Task T2-4

  Other Relevant Tasks: NA or list the tasks that are relevant to this dissertation in case it is related to multiple tasks.

- **Thesis Title:** Provide Title

  Student Name: Provide Student Name

  Defense Date: Provide Defense Date

  Advisor: Provide Advisor Name

  Link to the Thesis: Provide Thesis Link if available (if not delete this item

  Most Relevant Task: Task T2-4

  Other Relevant Tasks: NA or list the tasks that are relevant to this thesis in case it is related to multiple tasks.

- **Dissertation Title:** Device-to-Device Communication in 5G Wireless Networks

  Student Name: Niloofar Bahadori

  Defense Date: February, 9, 2021

  Advisor: Dr. John Kelley, Dr. Abdollah Homaifar

  Most Relevant Task: Task T2-5

  Other Relevant Tasks: NA

- **Thesis Title:** Modeling and Analyzing the Effects of Delays Consensus of Networks of Multi Agent Systems

  Student Name: Myrielle Allen Prince

  Defense Date: March 28, 2016

  Advisor: Dr. Sun Yi

  Most Relevant Task: Task T2-6

  Other Relevant Tasks: NA

- **Dissertation Title:** Internet Teleoperation and Time Delay in Multi-Agent Systems

  Student Name: Christopher Thomas

  Defense Date: March 18, 2020

Advisor: Dr. Sun Yi
Most Relevant Task: Task T2-6
Other Relevant Tasks: NA

- **Thesis Title:** Adaptive and Neural Network Based Control of Unmanned Aerial Vehicles
  Student Name: Mackenzie Truman Matthews
  Defense Date: March 5, 2021
  Advisor: Dr. Sun Yi
  Most Relevant Task: Task T2-6
  Other Relevant Tasks: NA

- **Dissertation Title:** Advanced Algorithms for secure and spectrally efficient 5G-IoT Networks
  Student Name: Anil Kumar Yerrapragada
  Defense Date: 10/19/2020
  Advisor: Dr. Brian Kelley
  Most Relevant Task: Task T2-7
  Other Relevant Tasks: NA

- **Dissertation Title:** Motion planning and control framework for robotic swarms
  Student Name: Solomon Genene Gudeta
  Defense Date: August 05, 2021
  Advisor: Dr. Ali Karimoddini
  Most Relevant Task: Task T3-1
  Other Relevant Tasks: NA

- **Dissertation Title:** Automatic Tasking of Multi-Agent Systems Using Behavior Tree
  Student Name: Tadewos G. Tadewos
  Defense Date: June 2021
  Advisor: Dr. Ali Karimoddini
  Most Relevant Task: Task T3-2
  Other Relevant Tasks: NA

- **Dissertation Title:** PhD is now under progress
  Student Name: Md Khurram Monir Rabby
  Defense Date: Expected *Dec,* 2021
  Advisor: Dr. Ali Karimoddini
  Link to the Dissertation: NA
  Most Relevant Task: Task T3-3

Other Relevant Tasks: NA

- **Dissertation Title:** A Novel Clustering-based Online Learning Framework For Data Stream (COLF)

  Student Name: Xuyang Yan

  Defense Date: 09/09/2022

  Advisor: Dr.Abdollah Homaifar

  Most Relevant Task: Task T3-4

  Other Relevant Tasks: NA

- **Dissertation Title:** Attention Mechanism Augmented Deep Neural Network Design for Time-series and Computer Vision Applications

  Student Name: Abenezer Girma

  Defense Date: 06/28/2022

  Advisor: Dr.Abdollah Homaifar

  Most Relevant Task: Task T3-4

  Other Relevant Tasks: NA

- **Thesis Title:** Knowledge Acquisition from Streaming Data through a Novel Dynamic Clustering Algorithm

  Student Name: Xuyang Yan

  Defense Date: 07/03/2018

  Advisor: Dr. Abdollah Homaifar

  Link to the Thesis: NA

  Most Relevant Task: Task T3-4

  Other Relevant Tasks: NA

- **Dissertation Title:** Managing Uncertainty In Sensor Data: An Evidence Theory Based Multisensor Data Fusion Approach.

  Student Name: Gabriel Awogbami

  Defense Date: 10/2019

  Advisor: Dr. Abdollah Homaifar

  Most Relevant Task: Task T3-5

  Other Relevant Tasks: NA or list the tasks that are relevant to this dissertation in case it is related to multiple tasks.

- **Dissertation Title:** Evaluation of trust in autonomous systems: human trust sensing and trustworthy autonomous driving

  Student Name: Biniam Gebru

  Defense Date: 02/2023

Advisor: Dr. Abdollah Homaifar

Most Relevant Task: Task T3-5

Other Relevant Tasks: NA

- **Dissertation Title:** Performance Evaluation of Autonomous UAV Systems Using Data-Driven Techniques

  Student Name: Mrinmoy Sarkar

  Defense Date: 06/14/2022

  Advisor: Dr. Abdollah Homaifar

  Most Relevant Task: Task T3-6

  Other Relevant Tasks: NA

- **Dissertation Title:** Provide Title

  Student Name: Provide Student Name

  Defense Date: Provide Defense Date

  Advisor: Provide Advisor Name

  Link to the Dissertation: Provide Dissertation Link if available (if not delete this item

  Most Relevant Task: Task T3-7

  Other Relevant Tasks: NA or list the tasks that are relevant to this dissertation in case it is related to multiple tasks.

- **Thesis Title:** Provide Title

  Student Name: Provide Student Name

  Defense Date: Provide Defense Date

  Advisor: Provide Advisor Name

  Link to the Thesis: Provide Thesis Link if available (if not delete this item

  Most Relevant Task: Task T3-7

  Other Relevant Tasks: NA or list the tasks that are relevant to this thesis in case it is related to multiple tasks.

- **Dissertation Title:** An Investigation of Neural Correspondence of Human Trust in Automation

  Student Name: Seeung Oh

  Defense Date: 2019

  Advisor: Younho Seong

  Link to the Dissertation: Provide Dissertation Link if available (if not delete this item

  Most Relevant Task: Task T3-8

  Other Relevant Tasks: Task 3-9

- **Thesis Title:** Provide Title

  Student Name: Provide Student Name

  Defense Date: Provide Defense Date

  Advisor: Provide Advisor Name

  Link to the Thesis: Provide Thesis Link if available (if not delete this item

  Most Relevant Task: Task T3-8

  Other Relevant Tasks: NA or list the tasks that are relevant to this thesis in case it is related to multiple tasks.

- **Dissertation Title:** Provide Title

  Student Name: Provide Student Name

  Defense Date: Provide Defense Date

  Advisor: Provide Advisor Name

  Link to the Dissertation: Provide Dissertation Link if available (if not delete this item

  Most Relevant Task: Task T3-9

  Other Relevant Tasks: NA or list the tasks that are relevant to this dissertation in case it is related to multiple tasks.

- **Thesis Title:** Provide Title

  Student Name: Provide Student Name

  Defense Date: Provide Defense Date

  Advisor: Provide Advisor Name

  Link to the Thesis: Provide Thesis Link if available (if not delete this item

  Most Relevant Task: Task T3-9

  Other Relevant Tasks: NA or list the tasks that are relevant to this thesis in case it is related to multiple tasks.

4.5 Claimed Intellectual Properties

16 journal and conference papers (1 accepted journal paper, 3 accepted conference papers, 2 submitted journal papers, and 10 submitted conference papers) were published and submitted as listed below:

- **Invention Title:** Surveillance systems and methods for automatic real-time monitoring

  Inventors' name: Berat Alper Erol, Abhijit Majumdar, Patrick Benavidez, Divya Bhaskaran, Mohammad Jamshidi, Benjamin Factor, Arman Rezakhani

  Disclosure Date: 9/20/2020

  Publication Date: 4/13/2021

  Status of the Innovation: Active Patent

  Most Relevant Task: Task T1-4

Other Relevant Tasks: NA

- **Invention Title:** Provide Title

  Inventors' name: Provide name of inventors

  Disclosure Date: Provide Defense Date

  Date of Disclosure: Provide Date of Disclosure

  Status of the Innovation: Provide Status of the innovation

  Most Relevant Task: Task T1-5

  Other Relevant Tasks: NA or list the tasks that are relevant to this invention in case it is related to multiple tasks.

- **Invention Title:** Provide Title

  Inventors' name: Provide name of inventors

  Disclosure Date: Provide Defense Date

  Date of Disclosure: Provide Date of Disclosure

  Status of the Innovation: Provide Status of the innovation

  Most Relevant Task: Task T1-5

  Other Relevant Tasks: NA or list the tasks that are relevant to this invention in case it is related to multiple tasks.

- **Invention Title:** Provide Title

  Inventors' name: Provide name of inventors

  Disclosure Date: Provide Defense Date

  Date of Disclosure: Provide Date of Disclosure

  Status of the Innovation: Provide Status of the innovation

  Most Relevant Task: Task T2-1

  Other Relevant Tasks: NA or list the tasks that are relevant to this invention in case it is related to multiple tasks.

- **Invention Title:** Provide Title

  Inventors' name: Provide name of inventors

  Disclosure Date: Provide Defense Date

  Date of Disclosure: Provide Date of Disclosure

  Status of the Innovation: Provide Status of the innovation

  Most Relevant Task: Task T2-1

  Other Relevant Tasks: NA or list the tasks that are relevant to this invention in case it is related to multiple tasks.

- **Invention Title:** Provide Title

Inventors' name: Provide name of inventors

Disclosure Date: Provide Defense Date

Date of Disclosure: Provide Date of Disclosure

Status of the Innovation: Provide Status of the innovation

Most Relevant Task: Task T2-4

Other Relevant Tasks: NA or list the tasks that are relevant to this invention in case it is related to multiple tasks.

- **Invention Title:** Provide Title

  Inventors' name: Provide name of inventors

  Disclosure Date: Provide Defense Date

  Date of Disclosure: Provide Date of Disclosure

  Status of the Innovation: Provide Status of the innovation

  Most Relevant Task: Task T2-4

  Other Relevant Tasks: NA or list the tasks that are relevant to this invention in case it is related to multiple tasks.

- **Invention Title:** Provide Title

  Inventors' name: Provide name of inventors

  Disclosure Date: Provide Defense Date

  Date of Disclosure: Provide Date of Disclosure

  Status of the Innovation: Provide Status of the innovation

  Most Relevant Task: Task T2-5

  Other Relevant Tasks: NA or list the tasks that are relevant to this invention in case it is related to multiple tasks.

- **Invention Title:** Provide Title

  Inventors' name: Provide name of inventors

  Disclosure Date: Provide Defense Date

  Date of Disclosure: Provide Date of Disclosure

  Status of the Innovation: Provide Status of the innovation

  Most Relevant Task: Task T2-5

  Other Relevant Tasks: NA or list the tasks that are relevant to this invention in case it is related to multiple tasks.

- **Invention Title:** Provide Title

  Inventors' name: Provide name of inventors

  Disclosure Date: Provide Defense Date

  Date of Disclosure: Provide Date of Disclosure

Status of the Innovation: Provide Status of the innovation

Most Relevant Task: Task T2-6

Other Relevant Tasks: NA or list the tasks that are relevant to this invention in case it is related to multiple tasks.

- **Invention Title:** Provide Title

  Inventors' name: Provide name of inventors

  Disclosure Date: Provide Defense Date

  Date of Disclosure: Provide Date of Disclosure

  Status of the Innovation: Provide Status of the innovation

  Most Relevant Task: Task T2-6

  Other Relevant Tasks: NA or list the tasks that are relevant to this invention in case it is related to multiple tasks.

- **Invention Title:** Provide Title

  Inventors' name: Provide name of inventors

  Disclosure Date: Provide Defense Date

  Date of Disclosure: Provide Date of Disclosure

  Status of the Innovation: Provide Status of the innovation

  Most Relevant Task: Task T3-1

  Other Relevant Tasks: NA or list the tasks that are relevant to this invention in case it is related to multiple tasks.

- **Invention Title:** Provide Title

  Inventors' name: Provide name of inventors

  Disclosure Date: Provide Defense Date

  Date of Disclosure: Provide Date of Disclosure

  Status of the Innovation: Provide Status of the innovation

  Most Relevant Task: Task T3-1

  Other Relevant Tasks: NA or list the tasks that are relevant to this invention in case it is related to multiple tasks.

- **Invention Title:** Provide Title

  Inventors' name: Provide name of inventors

  Disclosure Date: Provide Defense Date

  Date of Disclosure: Provide Date of Disclosure

  Status of the Innovation: Provide Status of the innovation

  Most Relevant Task: Task T3-4

Other Relevant Tasks: NA or list the tasks that are relevant to this invention in case it is related to multiple tasks.

- **Invention Title:** Provide Title

  Inventors' name: Provide name of inventors

  Disclosure Date: Provide Defense Date

  Date of Disclosure: Provide Date of Disclosure

  Status of the Innovation: Provide Status of the innovation

  Most Relevant Task: Task T3-4

  Other Relevant Tasks: NA or list the tasks that are relevant to this invention in case it is related to multiple tasks.

- **Invention Title:** Provide Title

  Inventors' name: Provide name of inventors

  Disclosure Date: Provide Defense Date

  Date of Disclosure: Provide Date of Disclosure

  Status of the Innovation: Provide Status of the innovation

  Most Relevant Task: Task T3-5

  Other Relevant Tasks: NA or list the tasks that are relevant to this invention in case it is related to multiple tasks.

- **Invention Title:** Provide Title

  Inventors' name: Provide name of inventors

  Disclosure Date: Provide Defense Date

  Date of Disclosure: Provide Date of Disclosure

  Status of the Innovation: Provide Status of the innovation

  Most Relevant Task: Task T3-5

  Other Relevant Tasks: NA or list the tasks that are relevant to this invention in case it is related to multiple tasks.

- **Invention Title:** Provide Title

  Inventors' name: Provide name of inventors

  Disclosure Date: Provide Defense Date

  Date of Disclosure: Provide Date of Disclosure

  Status of the Innovation: Provide Status of the innovation

  Most Relevant Task: Task T3-6

  Other Relevant Tasks: NA or list the tasks that are relevant to this invention in case it is related to multiple tasks.

- **Invention Title:** Provide Title

Inventors' name: Provide name of inventors

Disclosure Date: Provide Defense Date

Date of Disclosure: Provide Date of Disclosure

Status of the Innovation: Provide Status of the innovation

Most Relevant Task: Task T3-6

Other Relevant Tasks: NA or list the tasks that are relevant to this invention in case it is related to multiple tasks.

- **Invention Title:** Provide Title

  Inventors' name: Provide name of inventors

  Disclosure Date: Provide Defense Date

  Date of Disclosure: Provide Date of Disclosure

  Status of the Innovation: Provide Status of the innovation

  Most Relevant Task: Task T3-7

  Other Relevant Tasks: NA or list the tasks that are relevant to this invention in case it is related to multiple tasks.

- **Invention Title:** Provide Title

  Inventors' name: Provide name of inventors

  Disclosure Date: Provide Defense Date

  Date of Disclosure: Provide Date of Disclosure

  Status of the Innovation: Provide Status of the innovation

  Most Relevant Task: Task T3-7

  Other Relevant Tasks: NA or list the tasks that are relevant to this invention in case it is related to multiple tasks.

- None

- **Invention Title:** Provide Title

  Inventors' name: Provide name of inventors

  Disclosure Date: Provide Defense Date

  Date of Disclosure: Provide Date of Disclosure

  Status of the Innovation: Provide Status of the innovation

  Most Relevant Task: Task T3-9

  Other Relevant Tasks: NA or list the tasks that are relevant to this invention in case it is related to multiple tasks.

- **Invention Title:** Provide Title

  Inventors' name: Provide name of inventors

Disclosure Date: Provide Defense Date

Date of Disclosure: Provide Date of Disclosure

Status of the Innovation: Provide Status of the innovation

Most Relevant Task: Task T3-9

Other Relevant Tasks: NA or list the tasks that are relevant to this invention in case it is related to multiple tasks.

4.6 Financially Supported and Involved Students

16 journal and conference papers (1 accepted journal paper, 3 accepted conference papers, 2 submitted journal papers, and 10 submitted conference papers) were published and submitted as listed below:

- **Student Name:** Nicolas Gamez

  Level of Student: Masters

  Period of Support: From 1/16/2015 to 5/10/2017

  Type of Support: Stipend

  Dissertation/Thesis Title: Modeling, Simulation, and Design of a Time-Delayed Multi-Agent System of Autonomous Vehicles

  Most Relevant Involved Task: Task T1-3

  Other Relevant Involved Tasks: T1-1

  Defense Date: 5/5/2017

  Student's Achievements: Two conference papers, book chapter

  Current Employer of the Student: SAIC

- **Student Name:** Ikram Mohammed

  Level of Student: Masters

  Period of Support: From Fall 2015 to Spring 2016

  Type of Support: Stipend

  Dissertation/Thesis Title: Design, construction and distrubuted control of high degree of freedom modular robot

  Most Relevant Involved Task: Task T1-1

  Other Relevant Involved Tasks: T1-3

  Defense Date: 5/15/2016

  Student's Achievements: Two conference papers

  Current Employer of the Student: Neosem Technology, Inc

- **Student Name:** Prasanna Kolar

  Level of Student: PhD

  Period of Support: From Fall 2018 to (current)

Type of Support: Tuition and stipend

Dissertation/Thesis Title: Smart Mobility in Known Constricted Environment using Brain Computer Interfaces Implementing Riemannian Geometry

Most Relevant Involved Task: Task T1-1

Other Relevant Involved Tasks: Task T1-3

Defense Date: TBD

Student's Achievements: Book chapter

Current Employer of the Student: Southwest Research Institute

- **Student Name:** Abhijit Majumdar

  Level of Student: Masters

  Period of Support: From Fall 2016 to Fall 2018

  Type of Support: Stipend

  Dissertation/Thesis Title: Autonomous Intelligent Control for UAV with Self Docking, Charging and Learning Capabilities

  Most Relevant Involved Task: Task T1-3, T1-4

  Other Relevant Involved Tasks: T1-1

  Defense Date: 12/11/2018

  Student's Achievements: Conference papers, journal paper, book chapter, patent

  Current Employer of the Student: PlusOne Robotics

- **Student Name:** Berat Erol

  Level of Student: PhD

  Period of Support: From Fall 2015 to Summer 2018

  Type of Support: Stipend and Tuition

  Dissertation/Thesis Title: Towards Artificial Emotional Intelligence for Heterogeneous System to Improve Human Robot Interactions

  Most Relevant Involved Task: Task T1-3, T1-4

  Other Relevant Involved Tasks: T1-1

  Defense Date: 5/15/2018

  Student's Achievements: Conference papers, journal paper, book chapter, patent

  Current Employer of the Student: Technical Univ of Izmir

- **Student Name:** Nima Ebadi

  Level of Student: PhD

  Period of Support: From Spring 2017 to Summer 2018

  Type of Support: Stipend

  Dissertation/Thesis Title: TBD

Most Relevant Involved Task: Task T1-1

Other Relevant Involved Tasks: T1-1

Defense Date: 12/20/2021

Student's Achievements: Conference paper

Current Employer of the Student: UTSA

- **Student Name:** Amin Sahba

  Level of Student: PhD

  Period of Support: From Spring 2019 to Spring 2021

  Type of Support: Tuition and stipend

  Dissertation/Thesis Title: Intelligent Optimized Flow Control of Connected Driverless Vehicles in Smart Cities

  Most Relevant Involved Task: Task T1-2

  Other Relevant Involved Tasks: NA

  Defense Date: Summer 2021

  Student's Achievements: 3 conference papers

  Current Employer of the Student: UTSA

- **Student Name:** Ramin Sahba

  Level of Student: PhD

  Period of Support: From Spring 2019 to Spring 2021

  Type of Support: Tuition and stipend

  Dissertation/Thesis Title: 3D Object Detection for Autonomous Vehicles Perception Based on Combination of LiDAR, Radar, and Image Data

  Most Relevant Involved Task: Task T1-2

  Other Relevant Involved Tasks: NA

  Defense Date: Summer 2021

  Student's Achievements: 3 conference papers

  Current Employer of the Student: UTSA

- **Student Name:** Halid Kaplan

  Level of Student: PhD

  Period of Support: From Spring 2019 to Spring 2021

  Type of Support: Tuition and stipend

  Dissertation/Thesis Title: A health monitoring design based on deep learning approach for smart grid applications.

  Most Relevant Involved Task: Task T1-2

  Other Relevant Involved Tasks: Task T1-3

Defense Date: Summer 2021
Student's Achievements: 2 book chapter, 2 Journal papers, 1 conference paper
Current Employer of the Student: Hallmark University

- **Student Name:** Yunus Yetis
  Level of Student: PhD
  Period of Support: From Spring 2018 to Spring 2020
  Type of Support: Tuition and stipend
  Dissertation/Thesis Title:
  Most Relevant Involved Task: Task T1-2
  Other Relevant Involved Tasks: NA
  Defense Date: Summer 2020
  Student's Achievements: Conference papers, journals
  Current Employer of the Student: USAA
- **Student Name:** Parsa Yousefi
  Level of Student: PhD
  Period of Support: From Spring 2017 to Summer 2018
  Type of Support: Stipend
  Dissertation/Thesis Title: N/A
  Most Relevant Involved Task: Task T1-2
  Other Relevant Involved Tasks: NA
  Defense Date: N/A
  Student's Achievements: 1 Conference Paper
  Current Employer of the Student: Unknown
- **Student Name:** Hamid Fekriazgomi
  Level of Student: PhD
  Period of Support: From Spring 2017 to Summer 2018
  Type of Support: Stipend
  Dissertation/Thesis Title: N/A
  Most Relevant Involved Task: Task T1-2
  Other Relevant Involved Tasks: NA
  Defense Date: N/A
  Student's Achievements: 1 Conference Paper
  Current Employer of the Student: Unknown
- **Student Name:** Nicolas Gamez

Level of Student: MS

Period of Support: From Fall 2015 to Spring 2017

Type of Support: Stipend

Dissertation/Thesis Title: Modeling, Simulation, and Design of a Time-Delayed Multi-Agent System of Autonomous Vehicles

Most Relevant Involved Task: Task T1-3, T1-6

Other Relevant Involved Tasks: Task T1-1, T1-2

Defense Date: 5/5/2017

Student's Achievements: 1 Conference Paper

Current Employer of the Student: SAIC

- **Student Name:** Berat Erol

  Level of Student: PhD

  Period of Support: From Fall 2015 to Summer 2018

  Type of Support: Stipend and Tuition

  Dissertation/Thesis Title: Towards Artificial Emotional Intelligence for Heterogeneous System to Improve Human Robot Interactions

  Most Relevant Involved Task: Task T1-3, T1-4

  Other Relevant Involved Tasks: T1-1

  Defense Date: 5/15/2018

  Student's Achievements: Conference papers, journal paper, book chapter, patent

  Current Employer of the Student: Technical Univ of Izmir

- **Student Name:** Ikram Mohammed

  Level of Student: Masters

  Period of Support: From Fall 2015 to Spring 2016

  Type of Support: Stipend

  Dissertation/Thesis Title: Design, construction and distrubuted control of high degree of freedom modular robot

  Most Relevant Involved Task: Task T1-1

  Other Relevant Involved Tasks: T1-3

  Defense Date: 5/15/2016

  Student's Achievements: Two conference papers

  Current Employer of the Student: Neosem Technology, Inc

- **Student Name:** Prasanna Kolar

  Level of Student: PhD

  Period of Support: From Fall 2018 to Spring 2022

Type of Support: Tuition and stipend

Dissertation/Thesis Title: Smart Mobility in Known Constricted Environment using Brain Computer Interfaces Implementing Riemannian Geometry

Most Relevant Involved Task: Task T1-1

Other Relevant Involved Tasks: Task T1-3

Defense Date: Spring 2022

Student's Achievements: Book chapter

Current Employer of the Student: Southwest Research Institute

- **Student Name:** Abhijit Majumdar

  Level of Student: Masters

  Period of Support: From Fall 2016 to Fall 2018

  Type of Support: Stipend

  Dissertation/Thesis Title: Autonomous Intelligent Control for UAV with Self Docking, Charging and Learning Capabilities

  Most Relevant Involved Task: Task T1-3, T1-4

  Other Relevant Involved Tasks: T1-1

  Defense Date: 12/11/2018

  Student's Achievements: Conference papers, journal paper, book chapter, patent

  Current Employer of the Student: PlusOne Robotics

- **Student Name:** Nicolas Gamez

  Level of Student: Masters

  Period of Support: From 1/16/2015 to 5/10/2017

  Type of Support: Stipend

  Dissertation/Thesis Title: Modeling, Simulation, and Design of a Time-Delayed Multi-Agent System of Autonomous Vehicles

  Most Relevant Involved Task: Task T1-3

  Other Relevant Involved Tasks: T1-1

  Defense Date: 5/5/2017

  Student's Achievements: Two conference papers, book chapter

  Current Employer of the Student: SAIC

- **Student Name:** Tyrell Lewis

  Level of Student: PhD

  Period of Support: From 01/2021 to 05/2022

  Type of Support: Tuition and stipend

  Dissertation/Thesis Title: Reinforcement Learning for Autonomous Robotic Systems

Most Relevant Involved Task: Task T1-3

Other Relevant Involved Tasks: NA

Defense Date: TBD

Student's Achievements: Provide number of publications, presentations in national or international conferences, awards, etc

Current Employer of the Student: University of Texas at San Antonio

- **Student Name:** Berat Erol

  Level of Student: PhD

  Period of Support: From 9/1/2015 to 7/14/2018

  Type of Support: Stipend and Tuition

  Dissertation/Thesis Title: Towards Artificial Emotional Intelligence for Heterogeneous System to Improve Human Robot Interactions

  Most Relevant Involved Task: T1-4

  Other Relevant Involved Tasks:

  Defense Date: 5/15/2018

  Student's Achievements: Conference papers, journal paper, book chapter, patent

  Current Employer of the Student: Technical Univ of Izmir

- **Student Name:** Abhijit Majumdar

  Level of Student: MS

  Period of Support: From 9/1/2016 to 12/15/2018

  Type of Support: Stipend

  Dissertation/Thesis Title: Autonomous Intelligent Control for UAV with Self Docking, Charging and Learning Capabilities

  Most Relevant Involved Task: T1-3, T1-4

  Other Relevant Involved Tasks: T1-1

  Defense Date: 12/11/2018

  Student's Achievements: Conference papers, journal paper, book chapter, patent

  Current Employer of the Student: PlusOne Robotics

- **Student Name:** Divya Bhaskaran

  Level of Student: MS

  Period of Support: From 1/1/2015 to 12/20/2017

  Type of Support: Stipend

  Dissertation/Thesis Title: Comparative Performance Analysis of Navigation Algorithm and Deep Learning Application: Different Infrastructure and Cloud Robotics

  Most Relevant Involved Task: T1-6, T1-4

Other Relevant Involved Tasks:
Defense Date: 12/12/2017
Student's Achievements: Conference papers, journal paper, book chapter, patent
Current Employer of the Student: American NITTO SEIKO CORPORATION

- **Student Name:** Mohan Muppidi Kumar
  Level of Student: MS
  Period of Support: From 4/14/2013 to 12/20/2014
  Type of Support: Stipend
  Dissertation/Thesis Title: Vision based cloud robotics
  Most Relevant Involved Task: T1-4
  Other Relevant Involved Tasks:
  Defense Date: 12/5/2014
  Student's Achievements: Conference papers, journal paper, book chapter
  Current Employer of the Student: iROBOT

- **Student Name:** Ibrahim Mohammed
  Level of Student: MS
  Period of Support: From 6/15/2016 to 12/14/2016
  Type of Support: Stipend
  Dissertation/Thesis Title: Optimal navigation of autonomous vehicles
  Most Relevant Involved Task: T1-4
  Other Relevant Involved Tasks:
  Defense Date: 10/10/2016
  Student's Achievements: Conference papers
  Current Employer of the Student: Works in Saudi Arabia

- **Student Name:** Satish Vaishnav
  Level of Student: MS
  Period of Support: From 1/23/2015 to 8/14/2015
  Type of Support: Stipend
  Dissertation/Thesis Title: Cooperative mapping and self-localization for multiple quadro-copters
  Most Relevant Involved Task: T1-4
  Other Relevant Involved Tasks:
  Defense Date: 7/14/2015
  Student's Achievements: Conference papers, journal paper
  Current Employer of the Student: Aptiv

- **Student Name:** Mevlut Demir

  Level of Student: PhD

  Period of Support: From 9/15/2016 to 12/12/2016

  Type of Support: Stipend

  Dissertation/Thesis Title: A Complete, Automated and Scalable Framework for Science and Engineering

  Most Relevant Involved Task: T1-6, T1-4

  Other Relevant Involved Tasks:

  Defense Date: 5/16/2020

  Student's Achievements: Conference papers, journal paper, book chapter

  Current Employer of the Student: UTSA

- **Student Name:** Patrick Benavdez

  Level of Student: PhD

  Period of Support: From 5/5/2015 to 8/15/2015

  Type of Support: Stipend and Tuition

  Dissertation/Thesis Title: Low-cost home multi-robot rehabilitation system for the disabled population

  Most Relevant Involved Task: T1-4

  Other Relevant Involved Tasks:

  Defense Date: 7/20/2015

  Student's Achievements: Conference papers, journal paper, book chapter, patent

  Current Employer of the Student: UTSA

- **Student Name:** Laya Shamgah

  Level of Student: PhD

  Period of Support: From August 2014 to June 2019

  Type of Support: Tuition and stipend

  Dissertation/Thesis Title: Reactive Symbolic Planning and Control of Autonomous Vehicles in Adversarial Environments

  Most Relevant Involved Task: Task T1-5

  Other Relevant Involved Tasks: NA

  Defense Date: May 24, 2019

  Student's Achievements: 3 conference papers and 2 journal papers

  Current Employer of the Student: Lowe's Corp.

- **Student Name:** Jonathan Lwowski

  Level of Student: PhD

Period of Support: From 1/1/2016 to 5/20/2019

Type of Support: Tuition and Stipend

Dissertation/Thesis Title: End to End Control of a Cloud-Based Heterogeneous Swarm for Cyber-Physical Systems

Most Relevant Involved Task: T1-6, T2-2

Other Relevant Involved Tasks:

Defense Date: 5/17/2019

Student's Achievements: Journals and Conference Papers

Current Employer of the Student: Drone Base

- **Student Name:** Ali Miraftabzadeh

  Level of Student: PhD

  Period of Support: From 1/10/2015 to 12/15/2017

  Type of Support: Tuition

  Dissertation/Thesis Title: Real-Time Adaptive Data-Driven Perception for Anomaly Priority Scoring at Scale

  Most Relevant Involved Task: T1-6

  Other Relevant Involved Tasks:

  Defense Date: 12/15/2017

  Student's Achievements: Journals and Conference Papers

  Current Employer of the Student: VRBO

- **Student Name:** Nicolas Gallardo

  Level of Student: MS

  Period of Support: From 9/1/2014 to 12/15/2016

  Type of Support: Stipend

  Dissertation/Thesis Title: Kinematic modeling and control of a human-robot platform for the blind and visually impaired

  Most Relevant Involved Task: T1-3, T1-6

  Other Relevant Involved Tasks:

  Defense Date: 12/10/2016

  Student's Achievements: Journals and Conference Papers

  Current Employer of the Student: Unknown in San Antonio

- **Student Name:** Nicholas Gamez

  Level of Student: MS

  Period of Support: From 1/16/2015 to 5/10/2017

  Type of Support: Stipend

Dissertation/Thesis Title: Modeling, Simulation, and Design of a Time-Delayed Multi-Agent System of Autonomous Vehicles

Most Relevant Involved Task: T1-3, T1-6

Other Relevant Involved Tasks:

Defense Date: 5/5/2017

Student's Achievements: Journals and Conference Papers

Current Employer of the Student: SAIC

- **Student Name:** Divya Bhaskaran

  Level of Student: MS

  Period of Support: From 1/1/2015 to 12/20/2017

  Type of Support: Stipend

  Dissertation/Thesis Title: Comparative Performance Analysis of Navigation Algorithm and Deep Learning Application: Different Infrastructure and Cloud Robotics

  Most Relevant Involved Task: T1-6, T1-4

  Other Relevant Involved Tasks:

  Defense Date: 12/12/2017

  Student's Achievements: Conference Papers

  Current Employer of the Student: American NITTO SEIKO CORPORATION

- **Student Name:** Sean Ackels

  Level of Student: PhD

  Period of Support: From 1/18/2018 to 4/30/2020

  Type of Support: Tuition and Stipend

  Dissertation/Thesis Title: Implementation and Simulation of a Mobile Sensor Network Using Lambda Architecture

  Most Relevant Involved Task: T1-6

  Other Relevant Involved Tasks:

  Defense Date: 4/5/2020

  Student's Achievements: Journals and Conference Papers

  Current Employer of the Student: UTSA

- **Student Name:** James Nelson

  Level of Student: MS

  Period of Support: From 1/18/2018 to 4/20/2020

  Type of Support: Tuition and Stipend

  Dissertation/Thesis Title: Mitigating Time-Delay Problems with Cloud-Based Networked Controlled UGVs

Most Relevant Involved Task: T1-6

Other Relevant Involved Tasks:

Defense Date: 3/20/2020

Student's Achievements: Conference Papers

Current Employer of the Student: PlusOne Robotics

- **Student Name:** Mevlut Demir

  Level of Student: PhD

  Period of Support: From 9/15/2016 to 12/12/2016

  Type of Support: Stipend

  Dissertation/Thesis Title: A Complete, Automated and Scalable Framework for Science and Engineering

  Most Relevant Involved Task: T1-6, T1-4

  Other Relevant Involved Tasks:

  Defense Date: 5/16/2020

  Student's Achievements: Journals and Conference Papers

  Current Employer of the Student: UTSA

- **Student Name:** Jacob Hudson

  Level of Student: MS

  Period of Support: From 1/17/2020 to 10/15/2021

  Type of Support: Stipend

  Dissertation/Thesis Title: Trash Collecting Robotic System Using Two Autonomous, Mobile-Manipulator Robots with Convolutional Neural Network Object Detection System

  Most Relevant Involved Task: T1-6

  Other Relevant Involved Tasks:

  Defense Date: 12/15/2021

  Student's Achievements: Conference Papers

  Current Employer of the Student: USAF

- **Student Name:** Manjari Gummidipundi

  Level of Student: MS

  Period of Support: From 5/25/2021 to 8/15/2021

  Type of Support: Stipend

  Dissertation/Thesis Title: Terrain Traversability Analysis for Off-Road Robots: Emphasis on Semantic Segmentation And Terrain Classification with Terrain Assessment

  Most Relevant Involved Task: T1-6

  Other Relevant Involved Tasks:

Defense Date: 12/15/2021

Student's Achievements: Conference Papers

Current Employer of the Student: Schneider Electric

- **Student Name:** Karthik Pai Haradi

  Level of Student: MS

  Period of Support: From 1/15/2016 to 12/16/2016

  Type of Support: Stipend

  Dissertation/Thesis Title:

  Most Relevant Involved Task: T1-6

  Other Relevant Involved Tasks:

  Defense Date: 12/20/2016

  Student's Achievements: Conference Papers

  Current Employer of the Student: Texas Oil Industry

- **Student Name:** Benjamin Champion

  Level of Student: PhD

  Period of Support: From 6/10/2016 to 6/30/2016

  Type of Support: Stipend

  Dissertation/Thesis Title:

  Most Relevant Involved Task: T1-6

  Other Relevant Involved Tasks:

  Defense Date: 11/17/2018

  Student's Achievements: Conference Papers

  Current Employer of the Student: Deakin University

- **Student Name:** Ira Wendell Bates, II

  Level of Student: Master or PhD

  Period of Support: From Fall 2018 to Fall 2020

  Type of Support: Tuition and stipend

  Dissertation/Thesis Title: Developing a Novel Active-Learning Technique for Fault Diagnosis of Unknown Systems

  Most Relevant Involved Task: Task T2-1

  Other Relevant Involved Tasks: NA

  Defense Date: October 21, 2020

  Student's Achievements: 2 publications, 1 publication under review, 2 conference presentations

  Current Employer of the Student: Raytheon Technologies

- **Student Name:** Jonathan Lwowski

  Level of Student: PhD

  Period of Support: From 1/15/16 to 5/15/2019

  Type of Support: Tuition and Stipend

  Dissertation/Thesis Title: End to End Control of a Cloud-Based Heterogeneous Swarm for Cyber-Physical Systems

  Most Relevant Involved Task: Task T1-6

  Other Relevant Involved Tasks: Task T2-2

  Defense Date: 5/17/2019

  Student's Achievements: 2 book chapters, 4 conference papers, 1 journal, 1 magazine chapter

  Current Employer of the Student: Kespry

- **Student Name:** Conor Wallace

  Level of Student: Master

  Period of Support: From 9/12/17 to 4/23/20

  Type of Support: Tuition and Stipend

  Dissertation/Thesis Title: Heterogeneous Data Analytics for Fault Diagnosis and Accommodation of a Mobile Robot with Dislocated Actuator Faults

  Most Relevant Involved Task: Task T2-2

  Other Relevant Involved Tasks: NA

  Defense Date: 4/23/2020

  Student's Achievements: 3 publications

  Current Employer of the Student: Sierra Nevada

- **Student Name:** Mauricio Figueroa

  Level of Student: Master

  Period of Support: From 8/22/20 to 8/15/21

  Type of Support: Stipend

  Dissertation/Thesis Title: NA

  Most Relevant Involved Task: Task T2-2

  Other Relevant Involved Tasks: NA

  Defense Date: Fall 2021

  Student's Achievements: NA

  Current Employer of the Student: UTSA

- **Student Name:** Zack Hudson

  Level of Student: Master

Period of Support: From 1/22/20 to 8/10/20

Type of Support: Stipend

Dissertation/Thesis Title: NA

Most Relevant Involved Task: Task T2-2

Other Relevant Involved Tasks: NA

Defense Date: NA

Student's Achievements: NA

Current Employer of the Student: USAF

- **Student Name:** Provide student name

  Level of Student: Master or PhD

  Period of Support: From ... to ....

  Type of Support: Tuition and stipend

  Dissertation/Thesis Title: Provide the research topic

  Most Relevant Involved Task: Task T2-4

  Other Relevant Involved Tasks: NA or list the tasks the student has been involved in case the student has been involved in multiple tasks.

  Defense Date: provide the defense data

  Student's Achievements: Provide number of publications, presentations in national or international conferences, awards, etc

  Current Employer of the Student: Provide the name of the Current Employer of the Student

- **Student Name:** Provide student name

  Level of Student: Master or PhD

  Period of Support: From ... to ....

  Type of Support: Tuition and stipend

  Dissertation/Thesis Title: Provide the research topic

  Most Relevant Involved Task: Task T2-4

  Other Relevant Involved Tasks: NA or list the tasks the student has been involved in case the student has been involved in multiple tasks.

  Defense Date: provide the defense data

  Student's Achievements: Provide number of publications, presentations in national or international conferences, awards, etc

  Current Employer of the Student: Provide the name of the Current Employer of the Student

- **Student Name:** Niloofar Bahadori

  Level of Student: PhD

Period of Support: From 2016 to 2021

Type of Support: Tuition and stipend

Dissertation/Thesis Title: Device-to-Device Communication in 5G Wireless Networks

Most Relevant Involved Task: Task T2-5

Other Relevant Involved Tasks: NA

Defense Date: Feb. , 9, 2021

Student's Achievements: 6 publications, 3 presentations in national or international conferences, 1 best paper awards in IEEE WTS conference.

Current Employer of the Student: Northeastern University

- **Student Name:** Myrielle Allen Prince

  Level of Student: Master

  Period of Support: Spring 2016

  Type of Support: Tuition and stipend

  Dissertation/Thesis Title: Modeling and Analyzing the Effects of Delays Consensus of Networks of Multi Agent Systems

  Most Relevant Involved Task: Task T2-6

  Other Relevant Involved Tasks: NA

  Defense Date: March 28, 2016

  Student's Achievements: 1 publication

  Current Employer of the Student: Boeing

- **Student Name:** Christopher Thomas

  Level of Student: PhD

  Period of Support: From Fall 2016 to Spring 2020

  Type of Support: Stipend

  Dissertation/Thesis Title: Internet Teleoperation and Time Delay in Multi-Agent Systems

  Most Relevant Involved Task: Task T2-6

  Other Relevant Involved Tasks: NA

  Defense Date: March 18, 2020

  Student's Achievements: 3 publications, 1 conference presentation

  Current Employer of the Student: PolarOnyx

- **Student Name:** Aaron Amusan

  Level of Student: MS

  Period of Support: Spring 2021

  Type of Support: Stipend and Tuition

  Dissertation/Thesis Title: Teleoperation of Robot Arms Using Gesture Recognition

Most Relevant Involved Task: Task T2-6
Other Relevant Involved Tasks: NA
Defense Date: May, 2022
Student's Achievements: 0 publication
Current Employer of the Student:

- **Student Name:** Taylor Eisman
  Level of Student: Master
  Period of Support: 2018
  Type of Support: Stipend
  Dissertation/Thesis Title: Physical Layer Security
  Most Relevant Involved Task: Task T2-7
  Other Relevant Involved Tasks: NA
  Defense Date: Summer 2021
  Student's Achievements: NA
  Current Employer of the Student: University of Texas, San Antonio
- **Student Name:** Dr. Anil Kumar Yerrapragada
  Level of Student: PhD
  Period of Support: NA
  Type of Support: NA
  Dissertation/Thesis Title: Physical Layer Security
  Most Relevant Involved Task: Task T2-7
  Other Relevant Involved Tasks: NA
  Defense Date: provide the defense data
  Student's Achievements: 7
  Current Employer of the Student: Indian Institute of Technology, Madras
- **Student Name:** Solomon Genene Gudeta
  Level of Student: PhD
  Period of Support: From January 2018 to July 2021
  Type of Support: Tuition and stipend
  Dissertation/Thesis Title: Motion planning and control framework for robotic swarms
  Most Relevant Involved Task: Task T3-1
  Other Relevant Involved Tasks: NA
  Defense Date: August 05, 2021
  Student's Achievements: 4 conferences and 3 journal papers

Current Employer of the Student: Embark Trucks Inc.

- **Student Name:** Tadewos G. Tadewos
  Level of Student: PhD
  Period of Support: From August 2016 to July 2021
  Type of Support: Tuition and stipend
  Dissertation/Thesis Title: Automatic Tasking of Multi-Agent Systems Using Behavior Tree
  Most Relevant Involved Task: Task T3-2
  Other Relevant Involved Tasks: NA
  Defense Date: June 2021
  Student's Achievements: 2 conferences and 3 journal papers
  Current Employer of the Student: SYNOPSYS

- **Student Name:** Md Khurram Monir Rabby
  Level of Student: PhD
  Period of Support: From August 2017 to *Dec.,* 2021 (expected)
  Type of Support: Tuition and stipend
  Dissertation/Thesis Title: PhD research on Human-Robot Collaboration is now under progress
  Most Relevant Involved Task: Task T3-3
  Other Relevant Involved Tasks: NA
  Student's Achievements: 2 conference papers and 2 journal papers (under review)
  Current Employer of the Student: NA as the student has not completed dissertation

- **Student Name:** Xuyang Yan
  Level of Student: Master
  Period of Support: From 2016 to 2018
  Type of Support: Tuition and stipend
  Dissertation/Thesis Title: Knowledge Acquisition from Streaming Data through a Novel Dynamic Clustering Algorithm
  Most Relevant Involved Task: Task T3-4
  Other Relevant Involved Tasks: NA
  Defense Date: 07/03/2018
  Student's Achievements: Three conference paper
  Current Employer of the Student: JP Morgan Chase Co.

- **Student Name:** Xuyang Yan
  Level of Student: PhD

Period of Support: From 2018 to 2022

Type of Support: Tuition and stipend

Dissertation/Thesis Title: A Novel Clustering-based Online Learning Framework For Data Stream (COLF)

Most Relevant Involved Task: Task T3-4

Other Relevant Involved Tasks: NA

Defense Date: 09/09/2022

Student's Achievements: Ten conference paper and seven journal paper

Current Employer of the Student: JP Morgan Chase Co.

- **Student Name:** Abenezer Girma

  Level of Student: PhD

  Period of Support: From 2017 to 2022

  Type of Support: Tuition and stipend

  Dissertation/Thesis Title: Attention Mechanism Augmented Deep Neural Network Design for Time-series and Computer Vision Applications

  Most Relevant Involved Task: Task T3-4

  Other Relevant Involved Tasks: NA.

  Defense Date: 06/28/2022

  Student's Achievements: ten conference paper and three journal paper.

  Current Employer of the Student: JP Morgan Chase Co.

- **Student Name:** Gabriel Awogbami

  Level of Student: PhD

  Period of Support: From 2014 to 2019

  Type of Support: Tuition and stipend

  Dissertation/Thesis Title: Managing Uncertainty In Sensor Data: An Evidence Theory Based Multisensor Data Fusion Approach.

  Most Relevant Involved Task: Task T3-5

  Other Relevant Involved Tasks: NA

  Defense Date: provide the defense data

  Student's Achievements: 5 publications

  Current Employer of the Student: NA

- **Student Name:** Biniam Gebru

  Level of Student: PhD

  Period of Support: From 2017 to 2023

  Type of Support: Tuition and stipend

Dissertation/Thesis Title: Evaluation of trust in autonomous systems: human trust sensing and trustworthy autonomous driving.

Most Relevant Involved Task: Task T3-5

Other Relevant Involved Tasks: NA

Defense Date: 02/2023

Student's Achievements: 6 publications

Current Employer of the Student: NA

- **Student Name:** Mrinmoy Sarkar

  Level of Student: PhD

  Period of Support: From aug, 2017 to dec, 2020

  Type of Support: Tuition and stipend

  Dissertation/Thesis Title: NA

  Most Relevant Involved Task: Task T3-6

  Other Relevant Involved Tasks: NA

  Defense Date: NA

  Student's Achievements: Two journal papers, Two conference papers and one conference presentation.

  Current Employer of the Student: ACIT Institute

- **Student Name:** Amber Hoenig

  Level of Student: PhD

  Period of Support: January 2017 to August 2020

  Type of Support: Tuition and stipend

  Dissertation/Thesis Title: Decision Making Using Automated Estimates in the Classification of Novel Stimuli

  Most Relevant Involved Task: Task T3-7

  Other Relevant Involved Tasks: NA

  Defense Date: N/A

  Student's Achievements: Paper presentation at the 10th International Conference on Applied Human Factors and Ergonomics held in Washington, D.C.; one paper publication.

  Current Employer of the Student: Student is continuing in the doctoral program at NCA&T.

-

- **Student Name:** Seeung Oh

  Level of Student: PhD

  Period of Support: From Jan 2018 to December 2018

  Type of Support: stipend

Dissertation/Thesis Title: An Investigation of Neural Correspondence of Human Trust in Automation

Most Relevant Involved Task: Task T3-8

Other Relevant Involved Tasks: task 3-9

Defense Date: 2018 May

Student's Achievements: journal papers, 5 conference proceedings, a few presentations, selected for NSF PhD

Current Employer of the Student: North Carolina A & T State Universituy

- **Student Name:** Marcia Nealy

  Level of Student: Master

  Period of Support: From Jan 2018 to December 2018

  Type of Support: Tuition and stipend

  Dissertation/Thesis Title: Development of Simulated Environment for Decision Making Performance with an Autonomous Systems under Uncertainty

  Most Relevant Involved Task: Task T3-8

  Other Relevant Involved Tasks: task 3-9

  Defense Date: 2018 May

  Student's Achievements: 2 Conference Proceedings, and 3 professional presentations provided.

  Current Employer of the Student:

- **Student Name:** Lavoris Langley

  Level of Student: MS

  Period of Support: From 2017 Summer to 2017 December

  Type of Support: Tuition and stipend

  Dissertation/Thesis Title: Human Trust In Automation

  Most Relevant Involved Task: Task T3-8

  Other Relevant Involved Tasks: NNone

  Defense Date: December 2017

  Student's Achievements:

  Current Employer of the Student:

- **Student Name:** Provide student name

  Level of Student: Master or PhD

  Period of Support: From ... to ....

  Type of Support: Tuition and stipend

  Dissertation/Thesis Title: Provide the research topic

Most Relevant Involved Task: Task T3-9

Other Relevant Involved Tasks: NA or list the tasks the student has been involved in case the student has been involved in multiple tasks.

Defense Date: provide the defense data

Student's Achievements: Provide number of publications, presentations in national or international conferences, awards, etc

Current Employer of the Student: Provide the name of the Current Employer of the Student

- **Student Name:** Provide student name

  Level of Student: Master or PhD

  Period of Support: From ... to ....

  Type of Support: Tuition and stipend

  Dissertation/Thesis Title: Provide the research topic

  Most Relevant Involved Task: Task T3-9

  Other Relevant Involved Tasks: NA or list the tasks the student has been involved in case the student has been involved in multiple tasks.

  Defense Date: provide the defense data

  Student's Achievements: Provide number of publications, presentations in national or international conferences, awards, etc

  Current Employer of the Student: Provide the name of the Current Employer of the Student

5 Education and Outreach Plans

5.1 Student Support-Training Pathway to Next Generation of STEM Leaders

During this period, we have provided research assistantship to 15 PhD and 4 Masters Students (the latter of which includes two USAF Smart Program civilian employees) as well as 16 undergraduate students at N.C. A&T, and SIPI. These students, in collaboration with their respective faculty advisors are involved in various research tasks and test bed preparations.

5.2 Collaborative Curriculum Development

The delivered autonomy-related courses included:

5.2.1 Optimal Linear Control Systems (ECE, N.C. A&T)

5.2.2 Machine Learning and Evolutionary Algorithms (ECE, N.C. A&T)

5.2.3 Sensation & Perception (ISE, N.C. A&T)

5.2.4 System Dynamics (ME, N.C. A&T)

5.2.5 Introduction to Modern Telecommunications (ECE, N.C. A&T)

5.2.6 Big data analytic and cloud infrastructure at UTSA

5.2.7 Robot Operating System (ROS) at SIPI

5.2.8 Engineering Internship at SIPI

The developed and offered collaborative courses that are being offered at both N.C. A&T and UTSA include:

5.2.9 Advanced Robotic Systems

5.2.10 4G LTE Wireless Communications

This assisted in sharing the resources and facilitate the exchange of knowledge between the students and instructors at N.C. A&T and UTSA. Furthermore, it did further foster the collaboration among students from both campuses to have common advisers, sharing and assisting each other in the experimental setup and writing papers.

5.3 Organizing a workshop

To aid our students in effective communication, a workshop was held at N.C. A&T on March 4, 2016. This workshop covered the fundamentals of effective communication, writing and public speaking, and over 40 students and faculty members were in attendance. The speakers were Dr. Yahya R. Kamalipour, Chair, Department of Journalism and Mass Communication, as well as Dr. Faye Spencer Maor, Associate Professor, Department of English (see attached appendix for details of this workshop).

5.4 Other educational activities

Within the framework proposed in this proposal, N.C. A&T, UTSA, and SIPI focus on the development of the following activities:

5.4.1 Summer Faculty Program for SIPI faculty in order to promote research collaboration among institutions.

5.4.2 Joint educational design projects involving undergraduate student teams from both institutions.

5.4.3 SIPI provided assistance in identifying and encouraging potential Native American students to apply and enroll in graduate programs at School of Engineering, The University of New Mexico.

5.5 Outreach Activities

In January 2016, two of the TECHLAV outreach undergraduate students, Na'Kayla and Dunamis, sent communication to the coaches of the 5 FTC teams, requesting information on any pressing issues they might have been experiencing and to see if the teams would be participating in any of the qualifying tournaments. Only two schools responded. Ben L. Smith High School attended a tournament at TSDCH on January 30, 2016 in Durham. Their team seemed to be fairly comfortable with the new technology. Southern Guilford High School attended a tournament on January 23, 2016 at Northern Guilford H.S. and on February 6, 2016 at Southern Guilford H.S. The outreach students also spoke with Brian Flack and Warren Berger, two experienced FTC coaches with experienced FTC teams. The purpose was to gain additional insight into how to be a successful mentor and how to develop a successful team. Southern Guilford is struggling with the basics and needed help with storm bots, robust robot build, short critical hardware, and Android Studios. After the intervention, Southern Guilford was left with a basic drive program. Penn Griffin School for the Arts was assisted with basic questions surrounding App Inventor.

In February, Na'Kayla and Dunamis distributed another round of communication to the teams. Penn Griffin responded and indicated that they would be attending the qualifier at Northern Guilford on 2/13/16 and that they needed help with autonomous programming. Dunamis attended the FTC State Championship as a volunteer. They began working on a feedback survey for the coaches. The first revision has been completed and is currently being modified. They are also having discussions on how to assist these schools in the summer.

SIPI has expanded its outreach to the Bernalillo high school, Bernalillo middle school and Pueblo of Jemez Education region. Program Outline for these Partner High Schools (and Lower Grades) are:

5.5.1 To develop creative and demonstrate effective ways to engage students in STEM subjects.

5.5.2 To develop and deliver curricula to meet this objective.

5.5.3 To engage student interest in STEM subjects by having them build Mars Rovers and learn to program and operate Mars Rovers in Mars-yards that have been constructed on the SIPI campus.

5.5.4 To teach students the math and physics required to build and operate the rovers.

5.5.5 To enable students to program and operate the Mars Rovers from their current classrooms.

References

[1] P. Corke, *Robotics, vision and control: fundamental algorithms in MATLAB*. Springer Science & Business Media, 2011, vol. 73.

[2] G. Cook, *Mobile robots: navigation, control and remote sensing*. John Wiley & Sons, 2011.

[3] S. Bouabdallah and R. Siegwart, "Backstepping and sliding-mode techniques applied to an indoor micro quadrotor," in *Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference on*. IEEE, 2005, pp. 2247–2252.

[4] J. Nilsson, "Real-time control systems with delays," Ph.D. dissertation, Lund Institute of Technology, 1998.

[5] A. H. Lang, S. Vora, H. Caesar, L. Zhou, J. Yang, and O. Beijbom, "Pointpillars: Fast encoders for object detection from point clouds," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 12 697–12 705.

[6] H. Caesar, V. Bankiti, A. H. Lang, S. Vora, V. E. Liong, Q. Xu, A. Krishnan, Y. Pan, G. Baldan, and O. Beijbom, "nuscenes: A multimodal dataset for autonomous driving," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 11 621–11 631.

[7] A. Simonelli, S. R. Bulo, L. Porzi, M. López-Antequera, and P. Kontschieder, "Disentan- gling monocular 3d object detection," in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 1991–1999.

[8] P. Sun, H. Kretzschmar, X. Dotiwalla, A. Chouard, V. Patnaik, P. Tsui, J. Guo, Y. Zhou, Y. Chai, B. Caine *et al.*, "Scalability in perception for autonomous driving: Waymo open dataset," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 2446–2454.

[9] D. H. Zhizhong Li, "Learning without forgetting," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2016, pp. 614–629.

[10] C. Belta and L. C. Habets, "Controlling a class of nonlinear systems on rectangles," *IEEE Transactions on Automatic Control*, vol. 51, no. 11, pp. 1749–1759, 2006.

[11] A. Ranjan and P. Jetley, "Evaluation of signal smoothing algorithms for stability of a quadrotor mav," in *IEEE International Conference on Autonomous Robot Systems and Competitions (ICARSC)*, May 2014, pp. 309–314.

[12] J. J. Park and B. Kuipers, "A smooth control law for graceful motion of differential wheeled mobile robots in 2d environment," in *IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2011, pp. 4896–4902.

[13] Y. B. Shtessel, I. A. Shkolnikov, and A. Levant, "Smooth second-order sliding modes: Missile guidance application," *Automatica*, vol. 43, no. 8, pp. 1470–1476, 2007.

[14] M. Zamani, G. Pola, M. Mazo, and P. Tabuada, "Symbolic models for nonlinear control systems without stability assumptions," *IEEE Transactions on Automatic Control*, vol. 57, no. 7, pp. 1804–1809, 2012.

[15] E. A. Emerson, "Temporal and modal logic, handbook of theoretical computer science (jan van leeuwen, ed.), vol. b," 1990.

[16] N. Piterman, A. Pnueli, and Y. Sa'ar, "Synthesis of reactive (1) designs," in *International Workshop on Verification, Model Checking, and Abstract Interpretation*. Springer, 2006, pp. 364–380.

[17] R. Bloem, B. Jobstmann, N. Piterman, A. Pnueli, and Y. Saar, "Synthesis of reactive (1) designs," *Journal of Computer and System Sciences*, vol. 78, no. 3, pp. 911–938, 2012.

[18] A. Karimoddini and H. Lin, "Hierarchical hybrid symbolic robot motion planning and control," *Asian Journal of Control*, vol. 17, no. 1, pp. 23–33, 2015.

[19] M. Muratori, J. Holden, M. Lammert, A. Duran, S. Young, and J. Gonder, "Potentials for platooning in u.s. highway freight transport," *SAE International Journal of Commercial Vehicles*, vol. 10, no. 1, p. 45–49, 2017.

[20] G. D. Cameron and G. I. Duncan, "Paramics—parallel microscopic simulation of road traffic," *The Journal of Supercomputing*, vol. 10, no. 1, pp. 25–53, 1996.

[21] G. Lammel, "Simulation of urban mobility—sumo: Key features and uses," *Jaxenter Newslet- ter*, 2017.

[22] Z. Y. Rawshdeh and S. M. Mahmud, "Toward strongley connected clustering structure in vehicular ad hoc networks," in *2009 IEEE 70th Vehicular Technology Conference Fall*, 2009, pp. 1–5.

[23] C. Thomas, S. Yi, S. Meadows, and R. Sherrill, "Adaptive smith predictor for teleoperation of uavs with time-varying internet delay," *International Journal of Control, Automation and Systems*, vol. 18, no. 6, pp. 1465–1473, 2020.

[24] S. K. Armah and S. Yi, "Adaptive control for quadrotor uavs considering time delay: Study with flight payload," *Robotics and Automation Engineering Journal*, vol. 2, no. 5, 2018.

[25] H. P. Breivold and K. Sandström, "Internet of Things for Industrial Automation–Challenges and Technical Solutions," in *Data Science and Data Intensive Systems (DSDIS), 2015 IEEE International Conference on*. IEEE, 2015, pp. 532–539.

[26] I. Yaqoob, E. Ahmed, I. A. T. Hashem, A. I. A. Ahmed, A. Gani, M. Imran, and M. Guizani, "Internet of things architecture: Recent advances, taxonomy, requirements, and open challenges," *IEEE wireless communications*, vol. 24, no. 3, pp. 10–16, 2017.

[27] M. Jaber, M. A. Imran, R. Tafazolli, and A. Tukmanov, "5G Backhaul Challenges and Emerging Research Directions: A Survey," *IEEE Access*, vol. 4, pp. 1743–1766, 2016.

[28] H. Ishii, Y. Kishiyama, and H. Takahashi, "A Novel Architecture for LTE-B: C-plane/U-plane Split and Phantom Cell Concept," in *Globecom Workshops (GC Wkshps), 2012 IEEE*. IEEE, 2012, pp. 624–630.

[29] K. Zheng, L. Zhao, J. Mei, M. Dohler, W. Xiang, and Y. Peng, "10 Gb/s Hetsnets with Millimeter-Wave Communications: Access and Networking-Challenges and Protocols," *IEEE Communications Magazine*, vol. 53, no. 1, pp. 222–231, 2015.

[30] Z. Pi, J. Choi, and R. Heath, "Millimeter-Wave Gigabit Broadband Evolution Toward 5G: Fixed Access and Backhaul," *IEEE Communications Magazine*, vol. 54, no. 4, pp. 138–144, 2016.

[31] X. Hou, X. Wang, H. Jiang, and H. Kayama, "Investigation of Massive MIMO in Dense Small Cell Deployment for 5G," in *Vehicular Technology Conference (VTC-Fall), 2016 IEEE 84th*. IEEE, 2016, pp. 1–6.

[32] M. Agiwal, A. Roy, and N. Saxena, "Next Generation 5G Wireless Networks: A Comprehensive Survey," *IEEE Communications Surveys & Tutorials*, vol. 18, no. 3, pp. 1617–1655, 2016.

[33] S.-R. Lee, S.-H. Moon, H.-B. Kong, and I. Lee, "Optimal Beamforming Schemes and its Capacity Behavior for Downlink Distributed Antenna Systems," *IEEE Transactions on Wireless Communications*, vol. 12, no. 6, pp. 2578–2587, 2013.

[34] R. Heath, T. Wu, Y. Kwon, and A. Soong, "Multiuser MIMO in Distributed Antenna Systems," *2010 Conference Record of the Forty Fourth Asilomar Conference on Signals, Systems and Computers*, pp. 1202–1206, 2010.

[35] F. Talebi, "A Tutorial on Interference Alignment."

[36] C. M. Yetis, T. Gou, S. A. Jafar, and A. H. Kayran, "On Feasibility of Interference Alignment in MIMO Interference Networks," *IEEE Transactions on Signal Processing*, vol. 58, no. 9, pp. 4771–4782, 2010.

[37] V. R. Cadambe and S. A. Jafar, "Interference Alignment and Degrees of Freedom of the *K*-user Interference Channel," *IEEE Transactions on Information Theory*, vol. 54, no. 8, pp. 3425–3441, 2008.

[38] K. Gomadam, V. R. Cadambe, and S. A. Jafar, "Approaching the Capacity of Wireless Networks Through Distributed Interference Alignment," in *Global Telecommunications Conference, 2008. IEEE GLOBECOM 2008. IEEE*. IEEE, 2008, pp. 1–6.

[39] P. Sudheesh, M. Magarini, and P. Muthuchidambaranathan, "Interference Alignment for the K-user MIMO X Network Using Time Division Multiple Access," in *2016 IEEE Distributed Computing, VLSI, Electrical Circuits and Robotics (DISCOVER)*. IEEE, 2016, pp. 123–127.

[40] W. Liu, J.-X. Sun, J. Li, and Y. Ma, "Interference Alignment for MIMO Downlink Multicell Networks," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 8, pp. 6159–6167, 2016.

[41] S.-H. Park and Y.-C. Ko, "K-user MIMO X Network System With Perfect Interference Alignment," in *2011 IEEE International Conference on Communications (ICC)*. IEEE, 2011, pp. 1–5.

[42] A. K. Yerrapragada and B. Kelley, "Design of K-user massive MIMO networks," in *Ubiquitous Computing, Electronics and Mobile Communication Conference (UEMCON), 2017 IEEE 8th Annual*. IEEE, 2017, pp. 377–383.

[43] O. El Ayach, A. Lozano, and R. W. Heath, "On the overhead of interference alignment: Training, feedback, and cooperation," *IEEE Transactions on Wireless Communications*, vol. 11, no. 11, pp. 4192–4203, 2012.

[44] R. K. Mungara, G. George, and A. Lozano, "Overhead and spectral efficiency of pilot-assisted interference alignment in time-selective fading channels," *IEEE Transactions on Wireless Communications*, vol. 13, no. 9, pp. 4884–4895, 2014.

[45] A. Yerrapragada and B. Kelley, "An IoT Self Organizing Network for 5G Dense Network Interference Alignment," in *2017 12th System of Systems Engineering Conference (SoSE)*. IEEE, 2017, pp. 1–6.

[46] A. K. Yerrapragada and B. Kelley, "Very High Throughput Internet of Things Networks with K access points and K devices," in *MILCOM 2019-2019 IEEE Military Communications Conference (MILCOM)*. IEEE, 2019, pp. 640–645.

[47] T. L. Marzetta, "Massive MIMO: an introduction," *Bell Labs Technical Journal*, vol. 20, pp. 11–22, 2015.

[48] M. Juntti, "Capacity Limits of MIMO Channels."

[49] J. Medbo, "Channel models for HIPERLAN/2 in different indoor scenarios," *ETSI/BRAN*, 1998.

[50] S. Dang, O. Amin, B. Shihada, and M.-S. Alouini, "What should 6G be?" *Nature Electronics*, vol. 3, no. 1, pp. 20–29, 2020.

[51] P. Yang, Y. Xiao, M. Xiao, and S. Li, "6G wireless communications: Vision and potential techniques," *IEEE Network*, vol. 33, no. 4, pp. 70–75, 2019.

[52] K. B. Letaief, W. Chen, Y. Shi, J. Zhang, and Y.-J. A. Zhang, "The roadmap to 6G: AI empowered wireless networks," *IEEE Communications Magazine*, vol. 57, no. 8, pp. 84–90, 2019.

[53] G. H. Golub and C. F. Van Loan, *Matrix computations*. JHU press, 2012, vol. 3.

[54] M. K. Ozdemir and H. Arslan, "Channel estimation for wireless OFDM systems," *IEEE Communications Surveys & Tutorials*, vol. 9, no. 2, pp. 18–48, 2007.

[55] M. Morelli and U. Mengali, "A Comparison of Pilot-Aided Channel Estimation Methods for OFDM Systems," *IEEE Transactions on signal processing*, vol. 49, no. 12, pp. 3065–3073, 2001.

[56] I. Parvez, A. Rahmati, I. Guvenc, A. I. Sarwat, and H. Dai, "A survey on low latency towards 5g: Ran, core network and caching solutions," *IEEE Communications Surveys & Tutorials*, vol. 20, no. 4, pp. 3098–3130, 2018.

[57] M. Cai, K. Hwang, Y.-K. Kwok, S. Song, and Y. Chen, "Collaborative internet worm containment," *IEEE Security & Privacy*, vol. 3, no. 3, pp. 25–33, 2005.

[58] A. V. Uskov, "Information security of mobile vpn: Conceptual models and design methodology," in *Electro/Information Technology (EIT), 2012 IEEE International Conference on*. IEEE, 2012, pp. 1–6.

[59] C. Yi, J. Abraham, A. Afanasyev, L. Wang, B. Zhang, and L. Zhang, "On the role of routing in named data networking," in *Proceedings of the 1st ACM Conference on Information-Centric Networking*. ACM, 2014, pp. 27–36.

[60] A. Mukherjee, S. A. A. Fakoorian, J. Huang, and A. L. Swindlehurst, "Principles of physical layer security in multiuser wireless networks: A survey," *IEEE Communications Surveys & Tutorials*, vol. 16, no. 3, pp. 1550–1573, 2014.

[61] X. Chen, D. W. K. Ng, W. H. Gerstacker, and H.-H. Chen, "A survey on multiple-antenna techniques for physical layer security," *IEEE Communications Surveys & Tutorials*, vol. 19, no. 2, pp. 1027–1053, 2017.

[62] A. Khisti and G. W. Wornell, "Secure transmission with multiple antennas i: The misome wiretap channel," *IEEE Transactions on Information Theory*, vol. 56, no. 7, pp. 3088–3104, 2010.

[63] A. Khisti and G. Wornell, "Secure transmission with multiple antennas ii: The mimome wiretap channel," *arXiv preprint arXiv:1006.5879*, 2010.

[64] H. M. Furqan, J. M. Hamamreh, and H. Arslan, "Secret key generation using channel quantization with SVD for reciprocal MIMO channels," in *Wireless Communication Systems (ISWCS), 2016 International Symposium on*. IEEE, 2016, pp. 597–602.

[65] H. Taha and E. Alsusa, "Secret Key Exchange Using Private Random Precoding in MIMO FDD and TDD Systems," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 6, pp. 4823–4833, 2017.

[66] C.-Y. Wu, P.-C. Lan, P.-C. Yeh, C.-H. Lee, and C.-M. Cheng, "Practical physical layer security schemes for MIMO-OFDM systems using precoding matrix indices," *IEEE journal on selected areas in communications*, vol. 31, no. 9, pp. 1687–1700, 2013.

[67] J.-P. Cheng, Y.-H. Li, P.-C. Yeh, and C.-M. Cheng, "MIMO-OFDM PHY integrated (MOPI) scheme for confidential wireless transmission," in *Wireless Communications and Networking Conference (WCNC), 2010 IEEE*. IEEE, 2010, pp. 1–6.

[68] R. Ferrús, O. Sallent, J. Pérez-Romero, and R. Agustí, "Management of network slicing in 5g radio access networks: Functional framework and information models," *arXiv preprint arXiv:1803.01142*, 2018.

[69] G. Geraci, H. S. Dhillon, J. G. Andrews, J. Yuan, and I. B. Collings, "Physical Layer Security in Downlink Multi-Antenna Cellular Networks," *IEEE Transactions on Communications*, vol. 62, no. 6, pp. 2006–2021, 2014.

[70] C. Wang and H.-M. Wang, "Physical layer security in millimeter wave cellular networks," *IEEE Transactions on Wireless Communications*, vol. 15, no. 8, pp. 5569–5585, 2016.

[71] H. Wang, X. Zhou, and M. C. Reed, "Physical layer security in cellular networks: A stochastic geometry approach," *IEEE Transactions on Wireless Communications*, vol. 12, no. 6, pp. 2776–2787, 2013.

[72] D. S. Karas, A.-A. A. Boulogeorgos, G. K. Karagiannidis, and A. Nallanathan, "Physical Layer Security in the Presence of Interference," *IEEE Wireless Communications Letters*, vol. 6, no. 6, pp. 802–805, 2017.

[73] P. Ormond, A. Yerrapragada, and B. Kelley, "Error rate analysis of physical layer security for sub-6 ghz 5g network planning," in *14th Annual System of Systems Engineering Conference*. IEEE, 2019.

[74] ——, "On the application of mimo space-time coding to physical layer security in sub-6 ghz 5g," in *14th Annual System of Systems Engineering Conference*. IEEE, 2019.

[75] I. Šeremet and S. Čaušević, "Benefits of using 5G Network Slicing to implement Vehicle-to-Everything (V2X) technology," in *2019 18th International Symposium INFOTEH-JAHORINA (INFOTEH)*. IEEE, 2019, pp. 1–6.

[76] *TS 123 501 - V15.2.0 - 5G; System Architecture for the 5G System (3GPP TS 23.501 version 15.2.0 Release 15)*, 3GPP, 2018.

[77] F. Baccelli, B. Błaszczyszyn *et al.*, "Stochastic geometry and wireless networks: Volume II Applications," *Foundations and Trends® in Networking*, vol. 4, no. 1–2, pp. 1–312, 2010.

[78] Samsung, "MIMO for Long Term Evolution," *R1-050889*, vol. GPPTSG RANWG1 42, 2005.

[79] E. Dahlman, S. Parkvall, and J. Skold, *5G NR: The next generation wireless access technology*. Academic Press, 2018.

[80] A. K. Yerrapragada and B. Kelley, "On the application of k-user mimo for 6g enhanced mobile broadband," *Sensors*, vol. 20, no. 21, p. 6252, 2020.

[81] ——, "An iot self organizing network for 5g dense network interference alignment," in *2017 12th System of Systems Engineering Conference (SoSE)*. IEEE, 2017, pp. 1–6.

[82] P. Ormond, A. K. Yerrapragada, and B. Kelley, "On the application of mimo space-time coding to physical layer security in sub-6 ghz 5g," in *2019 14th Annual Conference System of Systems Engineering (SoSE)*. IEEE, 2019, pp. 284–289.

[83] A. K. Yerrapragada, P. Ormond, and B. Kelley, "On the application of key-based physical layer security in 5g heterogeneous networks," in *MILCOM 2019-2019 IEEE Military Communications Conference (MILCOM)*. IEEE, 2019, pp. 1–6.

[84] M. Rubenstein, A. Cornejo, and R. Nagpal, "Programmable self-assembly in a thousand-robot swarm," *Science*, vol. 345, no. 6198, pp. 795–799, 2014.

[85] S. Bandyopadhyay, S.-J. Chung, and F. Y. Hadaegh, "Probabilistic and distributed control of a large-scale swarm of autonomous agents," *IEEE Transactions on Robotics*, vol. 33, no. 5, pp. 1103–1123, 2017.

[86] Z. Miao, J. Yu, J. Ji, and J. Zhou, "Multi-objective region reaching control for a swarm of robots," *Automatica*, vol. 103, pp. 81–87, 2019.

[87] E. Teruel, R. Aragues, and G. López-Nicolás, "A distributed robot swarm control for dynamic region coverage," *Robotics and Autonomous Systems*, vol. 119, pp. 51–63, 2019.

[88] D. Morgan, G. P. Subramanian, S.-J. Chung, and F. Y. Hadaegh, "Swarm assignment and trajectory optimization using variable-swarm, distributed auction assignment and sequential convex programming," *The International Journal of Robotics Research*, vol. 35, no. 10, pp. 1261–1285, 2016.

[89] R. Olfati-Saber, "Flocking for multi-agent dynamic systems: Algorithms and theory," *IEEE Transactions on automatic control*, vol. 51, no. 3, pp. 401–420, 2006.

[90] C. Belta and V. Kumar, "Abstraction and control for groups of robots," *IEEE Transactions on robotics*, vol. 20, no. 5, pp. 865–875, 2004.

[91] S. Zhao, "Affine formation maneuver control of multiagent systems," *IEEE Transactions on Automatic Control*, vol. 63, no. 12, pp. 4140–4155, 2018.

[92] A. Soni and H. Hu, "Formation control for a fleet of autonomous ground vehicles: A survey," *Robotics*, vol. 7, no. 4, p. 67, 2018.

[93] M. Brambilla, E. Ferrante, M. Birattari, and M. Dorigo, "Swarm robotics: a review from the swarm engineering perspective," *Swarm Intelligence*, vol. 7, no. 1, pp. 1–41, 2013.

[94] J. Vanualailai, "Stable emergent formations for a swarm of autonomous car-like vehicles," *International Journal of Advanced Robotic Systems*, vol. 16, no. 5, p. 1729881419849780, 2019.

[95] C. R. Weisbin and G. Rodriguez, "NASA robotics research for planetary surface exploration," *IEEE Robotics & Automation Magazine*, vol. 7, no. 4, pp. 25–34, 2000.

[96] J. R. Lawton, R. W. Beard, and B. J. Young, "A decentralized approach to formation maneuvers," *IEEE transactions on robotics and automation*, vol. 19, no. 6, pp. 933–941, 2003.

[97] M. A. Joordens and M. Jamshidi, "Consensus control for a system of underwater swarm robots," *IEEE Systems Journal*, vol. 4, no. 1, pp. 65–73, 2010.

[98] Y. Liu, J. Gao, X. Shi, and C. Jiang, "Decentralization of virtual linkage in formation control of multi-agents via consensus strategies," *Applied Sciences*, vol. 8, no. 11, p. 2020, 2018.

[99] Y. Hong, J. Hu, and L. Gao, "Tracking control for multi-agent consensus with an active leader and variable topology," *Automatica*, vol. 42, no. 7, pp. 1177–1182, 2006.

[100] A. Karimoddini, H. Lin, B. M. Chen, and T. H. Lee, "Hybrid three-dimensional formation control for unmanned helicopters," *Automatica*, vol. 49, no. 2, pp. 424–433, 2013.

[101] A. Karimoddini, M. Karimadini, and H. Lin, "Decentralized hybrid formation control of unmanned aerial vehicles," in *2014 American Control Conference*, 2014, pp. 3887–3892.

[102] A. Loria, J. D, and N. A. J, "Leader–follower formation and tracking control of mobile robots along straight paths," *IEEE transactions on control systems technology*, vol. 24, no. 2, pp. 727–732, 2015.

[103] A. Karimoddini, H. Lin, B. M. Chen, and T. Heng Lee, "Hybrid formation control of the unmanned aerial vehicles," *Mechatronics*, vol. 21, no. 5, pp. 886–898, 2011.

[104] A. V. Savkin, C. Wang, A. Baranzadeh, Z. Xi, and H. T. Nguyen, "Distributed formation building algorithms for groups of wheeled mobile robots," *Robotics and Autonomous Systems*, vol. 75, pp. 463–474, 2016.

[105] R. A. Freeman, P. Yang, and K. M. Lynch, "Distributed estimation and control of swarm formation statistics," in *2006 American Control Conference*. IEEE, 2006, pp. 7–pp.

[106] P. Yang, R. A. Freeman, and K. M. Lynch, "Multi-agent coordination by decentralized estimation and control," *IEEE Transactions on Automatic Control*, vol. 53, no. 11, pp. 2480–2496, 2008.

[107] D. Wang and G. Xu, "Full-state tracking and internal dynamics of nonholonomic wheeled mobile robots," *IEEE/ASME Transactions on mechatronics*, vol. 8, no. 2, pp. 203–214, 2003.

[108] S. Gudeta, A. Karimoddini, and M. Davoodi, "Robust dynamic average consensus for a net- work of agents with time-varying reference signals," in *2020 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*. IEEE, 2020, pp. 1368–1373.

[109] J. Turner, Q. Meng, G. Schaefer, A. Whitbrook, and A. Soltoggio, "Distributed task rescheduling with time constraints for the optimization of total task allocations in a multirobot system," *IEEE transactions on cybernetics*, vol. 48, no. 9, pp. 2583–2597, 2017.

[110] A. Ulusoy, S. L. Smith, X. C. Ding, C. Belta, and D. Rus, "Optimality and robustness in multi-robot path planning with temporal logic constraints," *The International Journal of Robotics Research*, vol. 32, no. 8, pp. 889–911, 2013.

[111] I. Filippidis, D. V. Dimarogonas, and K. J. Kyriakopoulos, "Decentralized multi-agent control from local ltl specifications," in *2012 IEEE 51st IEEE Conference on Decision and Control (CDC)*, Dec 2012, pp. 6235–6240.

[112] Y. Liu, M. Ficocelli, and G. Nejat, "A supervisory control method for multi-robot task allocation in urban search and rescue," in *2015 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR)*, Oct 2015, pp. 1–6.

[113] M. Darrah, W. Niland, and B. Stolarik, "Multiple uav dynamic task allocation using mixed integer linear programming in a sead mission," in *Infotech@ Aerospace*, 2005, p. 7164.

[114] A. Falsone, K. Margellos, and M. Prandini, "A distributed iterative algorithm for multi-agent milps: finite-time feasibility and performance characterization," *IEEE control systems letters*, vol. 2, no. 4, pp. 563–568, 2018.

[115] M. Colledanchise, R. M. Murray, and P. Ögren, "Synthesis of correct-by-construction behavior trees," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Sept 2017, pp. 6039–6046.

[116] M. Colledanchise, A. Marzinotto, D. V. Dimarogonas, and P. Oegren, "The advantages of using behavior trees in mult-robot systems," in *Proceedings of ISR 2016: 47st International Symposium on Robotics*. VDE, 2016, pp. 1–8.

[117] M. Colledanchise, D. Almeida, and P. Ogren, "Towards blended reactive planning and acting using behavior trees," *arXiv preprint arXiv:1611.00230*, 11 2016.

[118] G. A. Korsah, A. Stentz, and M. B. Dias, "A comprehensive taxonomy for multi-robot task allocation," *The International Journal of Robotics Research*, vol. 32, no. 12, pp. 1495–1512, 2013. [Online]. Available: https://doi.org/10.1177/0278364913496484

[119] B. M. Muir, "Trust in Automation: Part I. Theoretical Issues in the Study of Trust and Human Intervention in Automated Systems," *Ergonomics*, vol. 37, no. 11, pp. 1905–1922, 1994.

[120] B. M. Muir and N. Moray, "Trust in Automation. Part II. Experimental Studies of Trust and Human Intervention in a Process Control Simulation," *Ergonomics*, vol. 39, no. 3, pp. 429–460, 1996.

[121] M. Itoh and K. Tanaka, "Mathematical Modeling of Trust in Automation: Trust, Distrust, and Mistrust," in *Proceedings of the human factors and ergonomics society annual meeting*, vol. 44, no. 1. SAGE Publications Sage CA: Los Angeles, CA, 2000, pp. 9–12.

[122] B. Sadrfaridpour, "Trust-Based Control of Robotic Manipulators in Collaborative Assembly in Manufacturing," 2018.

[123] G. Roth, A. Schulte, F. Schmitt, and Y. Brand, "Transparency for a Workload-Adaptive Cognitive Agent in a Manned–Unmanned Teaming Application," *IEEE Transactions on Human-Machine Systems*, 2019.

[124] B. Sadrfaridpour and Y. Wang, "Human and Robot Collaborative Assembly Manufacturing : Trust Dynamics and Control," 2010.

[125] S. A. Fayazi, N. Wan, S. Lucich, A. Vahidi, and G. Mocko, "Optimal Pacing in a Cycling Time-Trial Considering Cyclist's Fatigue Dynamics," in *2013 American Control Conference*, June 2013, pp. 6442–6447.

[126] L. Ma, D. Chablat, F. Bennis, and W. Zhang, "A new simple dynamic muscle fatigue model and its validation," *International Journal of Industrial Ergonomics*, vol. 39, no. 1, pp. 211 – 220, 2009. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0169814108000863

[127] B. Luber and S. H. Lisanby, "Enhancement of Human Cognitive Performance Using Transcranial Magnetic Stimulation (TMS)," *Neuroimage*, vol. 85, pp. 961–970, 2014.

[128] J. Schmitt, M. Wingen, J. Ramaekers, E. Evers, and W. Riedel, "Serotonin and Human Cognitive Performance," *Current pharmaceutical design*, vol. 12, no. 20, pp. 2473–2486, 2006.

[129] S. U. Din and J. Shao, "Exploiting evolving micro-clusters for data stream classification with emerging class detection," *Information Sciences*, vol. 507, pp. 404–420, 2020.

[130] M. M. Masud, Q. Chen, L. Khan, C. Aggarwal, J. Gao, J. Han, and B. Thuraisingham, "Addressing concept-evolution in concept-drifting data streams," in *2010 IEEE International Conference on Data Mining*. IEEE, 2010, pp. 929–934.

[131] D. Brzezinski and J. Stefanowski, "Combining block-based and online methods in learning ensembles from concept drifting data streams," *Information Sciences*, vol. 265, pp. 50–67, 2014.

[132] S. U. Din, J. Shao, J. Kumar, W. Ali, J. Liu, and Y. Ye, "Online reliable semi-supervised learning on evolving data streams," *Information Sciences*, 2020.

[133] M. J. Hosseini, A. Gholipour, and H. Beigy, "An ensemble of cluster-based classifiers for semi-supervised classification of non-stationary data streams," *Knowledge and information systems*, vol. 46, no. 3, pp. 567–597, 2016.

[134] I. Khamassi, M. Sayed-Mouchaweh, M. Hammami, and K. Ghédira, "Discussion and review on evolving data streams and concept drift adapting," *Evolving systems*, vol. 9, no. 1, pp. 1–23, 2018.

[135] V. Losing, B. Hammer, and H. Wersing, "Tackling heterogeneous concept drift with the self-adjusting memory (sam)," *Knowledge and Information Systems*, vol. 54, no. 1, pp. 171–201, 2018.

[136] J. Shao, F. Huang, Q. Yang, and G. Luo, "Robust prototype-based learning on data streams," *IEEE Transactions on Knowledge and Data Engineering*, vol. 30, no. 5, pp. 978–991, 2017.

[137] P. ZareMoodi, H. Beigy, and S. K. Siahroudi, "Novel class detection in data streams using local patterns and neighborhood graph," *Neurocomputing*, vol. 158, pp. 234–245, 2015.

[138] E. R. de Faria, A. C. P. de Leon Ferreira, J. Gama *et al.*, "Minas: multiclass learning algorithm for novelty detection in data streams," *Data mining and knowledge discovery*, vol. 30, no. 3, pp. 640–680, 2016.

[139] M. Masud, J. Gao, L. Khan, J. Han, and B. M. Thuraisingham, "Classification and novel class detection in concept-drifting data streams under time constraints," *IEEE Transactions on Knowledge and Data Engineering*, vol. 23, no. 6, pp. 859–874, 2010.

[140] X. Mu, K. M. Ting, and Z.-H. Zhou, "Classification under streaming emerging new classes: A solution using completely-random trees," *IEEE Transactions on Knowledge and Data Engineering*, vol. 29, no. 8, pp. 1605–1618, 2017.

[141] X. Mu, F. Zhu, J. Du, E.-P. Lim, and Z.-H. Zhou, "Streaming classification with emerging new class by class matrix sketching." AAAI, 2017.

[142] P. ZareMoodi, S. K. Siahroudi, and H. Beigy, "Concept-evolution detection in non-stationary data streams: a fuzzy clustering approach," *Knowledge and Information Systems*, vol. 60, no. 3, pp. 1329–1352, 2019.

[143] X. Yan, M. Razeghi-Jahromi, A. Homaifar, B. A. Erol, A. Girma, and E. Tunstel, "A novel streaming data clustering algorithm based on fitness proportionate sharing," *IEEE Access*, vol. 7, pp. 184 985–185 000, 2019.

[144] N. S. Altman, "An introduction to kernel and nearest-neighbor nonparametric regression," *The American Statistician*, vol. 46, no. 3, pp. 175–185, 1992.

[145] S. Gu, Y. Cai, J. Shan, and C. Hou, "Active learning with error-correcting output codes," *Neurocomputing*, vol. 364, pp. 182–191, 2019.

[146] E. Lughofer, E. Weigl, W. Heidl, C. Eitzinger, and T. Radauer, "Recognizing input space and target concept drifts in data streams with scarcely labeled and unlabelled instances," *Information Sciences*, vol. 355, pp. 127–151, 2016.

[147] D. Dheeru and E. Karra Taniskidou, "UCI machine learning repository," 2017. [Online]. Available: http://archive.ics.uci.edu/ml

[148] A. Bifet, G. Holmes, and B. Pfahringer, "Leveraging bagging for evolving data streams," in *Joint European conference on machine learning and knowledge discovery in databases*. Springer, 2010, pp. 135–150.

[149] A. Bifet, G. Holmes, B. Pfahringer, R. Kirkby, and R. Gavaldà, "New ensemble methods for evolving data streams," in *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2009, pp. 139–148.

[150] A. Bifet and R. Gavaldà, "Adaptive learning from evolving data streams," in *International Symposium on Intelligent Data Analysis*. Springer, 2009, pp. 249–260.

[151] K. H. Brodersen, C. S. Ong, K. E. Stephan, and J. M. Buhmann, "The balanced accuracy and its posterior distribution," in *2010 20th International Conference on Pattern Recognition*. IEEE, 2010, pp. 3121–3124.

[152] J. D. Kelleher, B. Mac Namee, and A. D'arcy, *Fundamentals of machine learning for predictive data analytics: algorithms, worked examples, and case studies*. MIT press, 2020.

[153] H. I. Turhan, M. Demirekler, and M. Gunay, "A novel methodology for target classification based on dempster-shafer theory," in *Belief Functions: Theory and Applications: Third International Conference, BELIEF 2014, Oxford, UK, September 26-28, 2014. Proceedings 3*. Springer, 2014, pp. 393–402.

[154] G. Shafer, *A mathematical theory of evidence*. Princeton university press, 1976, vol. 42.

[155] M. C. Florea, A.-L. Jousselme, D. Grenier, and E. Bosse, "Combining belief functions and fuzzy membership functions," in *Multisensor, Multisource Information Fusion: Architectures, Algorithms, and Applications 2003*, vol. 5099. SPIE, 2003, pp. 113–122.

[156] C. K. Murphy, "Combining belief functions when evidence conflicts," *Decision support systems*, vol. 29, no. 1, pp. 1–9, 2000.

[157] L. Jiao, T. Denoeux, and Q. Pan, "A hybrid belief rule-based classification system based on uncertain training data and expert knowledge," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 46, no. 12, pp. 1711–1723, 2015.

[158] W. Kim, N. Kim, J. B. Lyons, and C. S. Nam, "Factors affecting trust in high-vulnerability human-robot interaction contexts: A structural equation modelling approach," *Applied ergonomics*, vol. 85, p. 103056, 2020.

[159] K. Akash, G. McMahon, T. Reid, and N. Jain, "Human trust-based feedback control: Dynamically varying automation transparency to optimize human-machine interactions," *IEEE Control Systems Magazine*, vol. 40, no. 6, pp. 98–116, 2020.

[160] H. M. Khalid, L. W. Shiung, P. Nooralishahi, Z. Rasool, M. G. Helander, L. C. Kiong, and C. Ai-vyrn, "Exploring psycho-physiological correlates to trust: Implications for human-robot-human interaction," in *Proceedings of the human factors and ergonomics society annual meeting*, vol. 60, no. 1. SAGE Publications Sage CA: Los Angeles, CA, 2016, pp. 697–701.

[161] P. Linardatos, V. Papastefanopoulos, and S. Kotsiantis, "Explainable ai: A review of machine learning interpretability methods," *Entropy*, vol. 23, no. 1, p. 18, 2020.

[162] F. Myers, "The future of dod test and evaluation resources," OFFICE OF THE UNDER SECRETARY OF DEFENSE WASHINGTON DC TEST RESOURCE ..., Tech. Rep., 2009.

[163] F. J. Damerau, D. E. Johnson, and M. C. Buskirk Jr, "Automatic labeling of unlabeled text data," Feb. 24 2004, uS Patent 6,697,998.

[164] J. Yang *et al.*, "Automatically labeling video data using multi-class active learning," in *Proceedings Ninth IEEE International Conference on Computer Vision*. IEEE, 2003, pp. 516–523.

[165] S. B. Kotsiantis, I. Zaharakis, and P. Pintelas, "Supervised machine learning: A review of classification techniques," *Emerging artificial intelligence applications in computer engineering*, vol. 160, pp. 3–24, 2007.

[166] D. Tomandl and A. Schober, "A modified general regression neural network (mgrnn) with new, efficient training algorithms as a robust 'black box'-tool for data analysis," *Neural networks*, vol. 14, no. 8, pp. 1023–1034, 2001.

[167] N. Friedman, D. Geiger, and M. Goldszmidt, "Bayesian network classifiers," *Machine learning*, vol. 29, no. 2-3, pp. 131–163, 1997.

[168] L. Breiman, *Classification and regression trees*. Routledge, 2017.

[169] M. Quigley, K. Conley, B. P. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, and A. Y. Ng, "Ros: an open-source robot operating system," in *ICRA Workshop on Open Source Software*, 2009.

[170] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel *et al.*, "Scikit-learn: Machine learning in python," *Journal of machine learning research*, vol. 12, no. Oct, pp. 2825–2830, 2011.

[171] The U.S. Air Force Research Laboratory, "Amase tutorial session," 2010. [Online]. Available: https://github.com/afrl-rq/OpenAMASE

[172] G. De Luca, J. M. N Silva, S. Cerasoli, J. Araújo, J. Campos, S. Di Fazio, and G. Modica, "Object-based land cover classification of cork oak woodlands using uav imagery and orfeo toolbox," *Remote Sensing*, vol. 11, no. 10, p. 1238, 2019.

[173] M. L. Cummings, S. Bruni, S. Mercier, and P. Mitchell, "Automation architecture for single operator, multiple uav command and control," Massachusetts Inst Of Tech Cambridge, Tech. Rep., 2007.

[174] J. S. McCarley and C. D. Wickens, "Human factors concerns in uav flight," 2004.

[175] N. Nigam, "The multiple unmanned air vehicle persistent surveillance problem: A review," *Machines*, vol. 2, no. 1, pp. 13–72, 2014.

[176] H. A. Ruff, S. Narayanan, and M. H. Draper, "Human interaction with levels of automation and decision-aid fidelity in the supervisory control of multiple simulated unmanned air vehicles," *Presence: Teleoperators & Virtual Environments*, vol. 11, no. 4, pp. 335–351, 2002.

[177] A. Appriou, "Multisensor data fusion in situation assessment processes," in *Qualitative and quantitative practical reasoning*. Springer, 1997, pp. 1–15.

[178] R. M. Taylor, "Human automation integration for supervisory control of uavs," Defence Science and Technology Lab Farnborough, United Kingdom, Tech. Rep., 2006.

[179] J. S. Warm, R. Parasuraman, and G. Matthews, "Vigilance requires hard mental work and is stressful," *Human factors*, vol. 50, no. 3, pp. 433–441, 2008.

[180] A. Hoenig and J. D. Stephens, "Decision making using automated estimates in the classification of novel stimuli," in *International Conference on Applied Human Factors and Ergonomics*. Springer, 2019, pp. 25–35.

[181] T. Fong and C. Thorpe, "Vehicle teleoperation interfaces," *Autonomous robots*, vol. 11, no. 1, pp. 9–18, 2001.

[182] A. Hocraffer and C. S. Nam, "A meta-analysis of human-system interfaces in unmanned aerial vehicle (uav) swarm management," *Applied ergonomics*, vol. 58, pp. 66–80, 2017.

[183] B. L. Hooey, D. B. Kaber, J. A. Adams, T. W. Fong, and B. F. Gore, "The underpinnings of workload in unmanned vehicle systems," *IEEE Transactions on Human-Machine Systems*, vol. 48, no. 5, pp. 452–467, 2017.

[184] L. Chérif, V. Wood, A. Marois, K. Labonté, and F. Vachon, "Multitasking in the military: Cognitive consequences and potential solutions," *Applied cognitive psychology*, vol. 32, no. 4, pp. 429–439, 2018.

[185] L. Wang, G. A. Jamieson, and J. G. Hollands, "Trust and reliance on an automated combat identification system," *Human factors*, vol. 51, no. 3, pp. 281–291, 2009.

[186] X. Xu, C. D. Wickens, and E. M. Rantanen, "Effects of conflict alerting system reliability and task difficulty on pilots' conflict detection with cockpit display of traffic information," *Ergonomics*, vol. 50, no. 1, pp. 112–130, 2007, pMID: 17178655. [Online]. Available: https://doi.org/10.1080/00140130601002658

[187] R. Parasuraman and V. Riley, "Humans and automation: Use, misuse, disuse, abuse," *Human factors*, vol. 39, no. 2, pp. 230–253, 1997.

[188] R. Hall, "Trusting your assistant," *Proceedings of the 11th Knowledge-Based Software Engineering Conference, KBSE'96*, vol. 31, no. 1, pp. 42–51, 1996.

[189] . M. N. Lee, J.D., "Trust, control strategies and allocation of function in human-machine systems," *Ergonomics*, vol. 35, no. 10, pp. 1243–1270, 1992.

[190] B. M. Muir, "Trust in automation. part i theoretical issues in the study of trust and human intervention in automated systems," *Ergonomics*, vol. 37, no. 11, pp. 1905–1922, 1994.

[191] T. B. Sheridan, "Trustworthiness of command and control systems," *Paper presented at the IFAC Man-Machine Systems*, vol. 1, no. 1, p. 1, 1988.

[192] P. D. F. . S. B. C. Glover, S. M., "The influence of decision aids on user behavior: Implications for knowledge acquisition and inappropriate reliance," *Organizational Behavior and Human Decision Processes*, vol. 72, no. 2, pp. 232–255, 1997.

[193] . R. J. Vicente, K. J., "Ecological interface design: Theoretical foundations," *IEEE Transactions on Systems, Man, and Cybernetics, SMC-22*, vol. 1, no. 1, pp. 589–606, 1992.

[194] C. B. Grounds and D. Wiley, "Alternative methods of mitigating automation distrust impacts," *Proceedings of the Human Factors & Ergonomics Society 45th Annual Meeting*, pp. 1728–1732, 2001.

[195] K. R. T. M. R. A. . H. S. G. Morrison, J. G., "Implications for decision-making research for decision support and displays." *In J. A. Cannon-Bowers & E. Salas (Eds.), Making decisions under stress: Implications for individual and team training*, pp. 375–406, 1998.

[196] H. J. G. . Z. M. P. Rempel, J. K., "Trust in close relationship," *Journal of Personality and Social Psychology,*, vol. 49, no. 1, pp. 95–112, 1985.

[197] . M. N. Muir, B. M., "Trust in automation. part ii experimental studies of trust and human intervention in a process control simulation," *Ergonomics*, vol. 39, no. 3, pp. 429–460, 1996.

[198] V. Riley, "Operator reliance on automation: Theory and data," *In R. Parasuraman & M. Mouloua (Eds.), Automation and Human Performance: Theory and Application*, pp. 22–27, 1996.

[199] . M. S. Jones, S., "Human-computer-human interaction: Trust in cscw," *Ergonomics International*, pp. 172–174, 1997.

[200] . P. M. J. Lerch, F. J., "How do we trust machine advise?" *In G. S. a. M. J. Smith (Ed.), Designing and using human-computer interface and knowledge based systems*, pp. 410–419, 1989.

[201] . S. A. F. Moffa, A. J., "Trust in a medical system: can we generalize between domains?" *In M. Mouloua & J. M. Koonce (Eds.), Human-Automation Interaction: Research and Practice*, pp. 218–224, 1996.

[202] V. M. . K. B. Beth, T., "Valuation of trust in open network," *In D. Gollman (Ed.), Computer Security - ESORICS 94, Third European Symposium on research in Computer Security*, pp. 3–18, 1994.

[203] D. Gefen, "E-commerce: the role of familiarity and trust," *Omega*, vol. 28, pp. 725–737, 2000.

[204] . S. K. A. Lee, J.D., "Trust in automation: Designing for appropriate reliance," *Human Factors*, vol. 46, no. 1, pp. 50–80, 2004.

[205] M. Deutsch, "New haven, cn: Yale university press." *STATE UNIV OF NEW YORK AT BUFFALO CENTER OF MULTISOURCE INFORMATION FUSION*, 1973.

[206] B. Barber, "The logic and limits of trust," *New Brunswick. Rutgers University Press*, 1983.

[207] D. J. H. . S. F. D. Mayer, R. C., "Trust between humans and machines, and design of decision aids," *International Journal of Man-Machine Studies*, vol. 27, pp. 527–539, 1995.

[208] A. M. B. J. . T. J. Uggiralaa, Ananth. Gramopadhyea, "Measurement of trust in complex and dynamic systems using a quantitative approach," *International Journal of Industrial Ergonomics*, vol. 34, pp. 175–186, 2004.

[209] N. Lee, J. D. & Moray, "Trust, self-confidence, and operators' adaptation to automation," *International Journal of Human-Computer Studies*, vol. 40, pp. 153–184, 1994.

[210] D. T. M. . P. M. M. Bhattacharya, R., "A formal model of trust based on outcomes," *Academy of management review*, vol. 23, no. 3, pp. 459–472, 1998.

[211] D. J. H. . S. F. D. Mayer, R. C., "An integrative model of organizational trust," *Academy of Management Review*, vol. 20, pp. 709–734, 1995.

[212] T. L. Larzelere, R. E. & Huston, "The dyadic trust scale: Toward understanding interpersonal trust in close relationships," *Journal of Marriage and the Family*, pp. 595–604, 1980.

[213] B. A. M. . D. C. G. Jian, Jiun-Yin, "Foundations for an empirically determined scale of trust in automated systems," *International journal of cognitive ergonomics*, vol. 4, no. 1, pp. 53–71, 2000.

[214] L. J. S. Y. F. R. . J. J. Y. Bisantz, A., "Empirical investigations of trust-related systems vulnerabilities in aided, adversarial decision making," *STATE UNIV OF NEW YORK AT BUFFALO CENTER OF MULTISOURCE INFORMATION FUSION*, 2000.

[215] S. B. O. J. Winston, J.S. and R. Dolan, "Automatic and intentional brain responses during evaluation of trustworthiness of faces," *Nature Neuroscience*, vol. 5, no. 3, pp. 277–283, 2002.

[216] M. K. M. J. K. N. Z. R. S. M. H. A. Krueger, Frank. and J. Grafman, "Neural correlates of trust," *PNAS*, vol. 104, no. 50, pp. 20 084–20 089, 2007.

[217] A. Dimoka, "Brain mapping of psychological processes with psychometric scales: An fmri method for social neuroscience." *NeuroImage*, vol. 54, no. 1, pp. S263–S271, 1973.

[218] S. A. . M. G. Huettel, S. A., *Functional magnetic resonance imaging*, 2004.

[219] G. C. B. C. K.-R. A. O. S. H. T. S. E. S. M. Frokjer, J.B. and A. Drewes, "Integrity of central nervous function in diabetes mellitus assessed by resting state eeg frequency analysis and source localization." *Journal of diabetes and its complications*, vol. 31, no. 2, pp. 400–406, 2017.

[220] S. A. Bunge and I. Kahn, "Cognition: An overview of neuroimaging techniques," *Encyclopedia of neuroscience*, vol. 2, pp. 1063–1067, 2009.

[221] B. Metcalf, "Frontal lobe function," *Encyclopedia of Theory in Psychology Thousand Oaks*, pp. 340–344, 2016.

[222] J. M. Fuster, "Frontal lobe and cognitive development," *Journal of Neuropsychology*, vol. 31, pp. 373–385, 2002.

[223] . R. A. A. Yang, Yaling., "Prefrontal structural and functional brain imaging findings in antisocial, violent, and psychopathic individuals: a meta-analysis," *Psychiatry Research*, vol. 174, no. 2, pp. 91–98, 2009.

[224] . A. R. A. Buneo, Christopher A., "The posterior parietal cortex: Sensorimotor interface for the planning and online control of visually guided movements," *Neuropsychologia*, pp. 2594–2606, 2006.

[225] . E. A. T. Semrud-Clikeman, M., "Child neuropsychology assessment and interventions for neurodevelopmental disorders," *New York*, 2009.

[226] E. A. Phelps, "Human emotion and memory: interactions of the amygdala and hippocampal complex," *Current opinion in Neurobiology*, vol. 14, no. 2, pp. 198–202, 2004.

LIST OF SYMBOLS, ABBREVIATIONS, AND ACRONYMS

Clinical decision support systems	CDSSs
Computational intelligence	CI
Demonstration, Implementation, and Integration	DII
Device-to-device	D2D
DOF	Degree of Freedom
Electroencephalography	EEG
Extended Kalman Filter	EKF
Functional magnetic resonance imaging	fMRI
Indian Polytechnic Institute	SIPI
Inertial Measurement Unit	IMU
K-nearest neighbor	KNN
Large- scale Autonomous Systems of Vehicles	LSASVs
Large-Scale Autonomous Systems of Vehicles	LSASVs
Learning without Forgetting	LwF
Light Detection and Ranging	LIDAR
Linear quadratic regulator	LQR
Linear Temporal Logic	LTL
Long Short-Term Memory	LSTM
MAPE	Mean Absolute Percentage Error
Maximum contaminant level	MCL
Modeling, Analysis and Control of Large-scale Autonomous systems of Vehicles	MACLAV
Multi-target dynamic reach-avoid	MTDRA
Neural network	ANN
North Carolina Agricultural and Technical State University	N.C. A&T
ORB	Oriented FAST and Rotated BRIEF
Perception inference engine	PIE
Principal component analysis	PCA
Quality of service	QoS
Recurrent Neural Network	RNN
Resilient Control and Communication of Large-scale Autonomous systems of Vehicles	RC2LAV
Robot Description File	URDF