



AFRL-RY-WP-TR-2020-0365

**ANALYTICAL APPROACH TO MULTI-OBJECTIVE
JOINT INFERENCE CONTROL FOR FIXED WING
UNMANNED AERIAL VEHICLES**

**Julian L. Casey
Wright State University**

**DECEMBER 2020
Final Report**

Approved for public release; distribution is unlimited.

See additional restrictions described on inside pages

© 2020 Julian L. Casey

STINFO COPY

**AIR FORCE RESEARCH LABORATORY
SENSORS DIRECTORATE
WRIGHT-PATTERSON AIR FORCE BASE, OH 45433-7320
AIR FORCE MATERIEL COMMAND
UNITED STATES AIR FORCE**

REPORT DOCUMENTATION PAGE				<i>Form Approved</i> OMB No. 0704-0188	
The public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Department of Defense, Washington Headquarters Services, Directorate for Information Operations and Reports (0704-0188), 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to any penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number. PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.					
1. REPORT DATE (DD-MM-YY) December 2020		2. REPORT TYPE Thesis		3. DATES COVERED (From - To) 30 October 2020 –30 October 2020	
4. TITLE AND SUBTITLE ANALYTICAL APPROACH TO MULTI-OBJECTIVE JOINT INFERENCE CONTROL FOR FIXED WING UNMANNED AERIAL VEHICLES				5a. CONTRACT NUMBER FA8650-15-D-1845-0001	
				5b. GRANT NUMBER	
				5c. PROGRAM ELEMENT NUMBER 61102F/62204F	
6. AUTHOR(S) Julian L. Casey				5d. PROJECT NUMBER 3003/6095	
				5e. TASK NUMBER N/A	
				5f. WORK UNIT NUMBER Y191	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Wright State University 3640 Colonel Glenn Hwy. Dayton, OH 45435				8. PERFORMING ORGANIZATION REPORT NUMBER	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) Air Force Research Laboratory Sensors Directorate Wright-Patterson Air Force Base, OH 45433-7320 Air Force Materiel Command United States Air Force				10. SPONSORING/MONITORING AGENCY ACRONYM(S) AFRL/RYPAR	
				11. SPONSORING/MONITORING AGENCY REPORT NUMBER(S) AFRL-RY-WP-TR-2020-0365	
12. DISTRIBUTION/AVAILABILITY STATEMENT Approved for public release; distribution is unlimited.					
13. SUPPLEMENTARY NOTES PAO case number 88ABW-2020-3356, Clearance Date 30 October 2020. © 2020 Julian L. Casey. Presented in partial fulfillment of the requirements for the degree of Master of Science in Engineering to Wright State University. This work was funded in whole or in part by Department of the Air Force. The U.S. Government has for itself and others acting on its behalf an unlimited, paid-up, nonexclusive, irrevocable worldwide license to use, modify, reproduce, release, perform, display, or disclose the work by or on behalf of the U.S. Government. Report contains color.					
14. ABSTRACT Fixed-wing Unmanned Aerial Vehicles (UAVs) have been found highly useful in various environments, including military and law enforcement. With the increased use of fixed-wing UAVs, there becomes an increased need to optimize the resources available. One approach to resource management is to create multi-objective flights. This thesis presents the design, analysis, and experimental implementation of multi-objective resource management for the resource of Range, distance available to the UAV, from the viewpoint of Intelligence Surveillance and Reconnaissance (ISR). First, a Simulation Environment is created capable of tracking multiple fixed-wing UAVs and to allow for the UAVs' being controlled by an externally driven algorithm. Second, an Inference algorithm is developed with the objective of information seeking. Several algorithms are developed and used in conjunction with a Sequential Analysis test to allow for calculating Target Value, calculating Target Confidence, and validating the calculated Target Value. Third, a Control algorithm is developed with the objective of Target seeking. The Control algorithm uses several approaches to path generation, including Dubins path, Optimized Order path, and Closest Target path. Finally, a supervisor algorithm termed Joint Inference and Control (JIC) joins Inference and Control together. Monte Carlo simulated test flight results are shown to illustrate the effectiveness of the developed algorithms.					
15. SUBJECT TERMS control, information theory, ISR, joint inference and control (JIC)					
16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT: SAR	18. NUMBER OF PAGES 82	19a. NAME OF RESPONSIBLE PERSON (Monitor) Trevor J. Bihl 19b. TELEPHONE NUMBER (Include Area Code) N/A
a. REPORT Unclassified	b. ABSTRACT Unclassified	c. THIS PAGE Unclassified			

ANALYTICAL APPROACH TO MULTI-OBJECTIVE JOINT INFERENCE CONTROL FOR FIXED WING UNMANNED AERIAL VEHICLES

A Thesis submitted in partial fulfillment
of the requirements for the degree of
Master of Science in Engineering

by

Julian L. Casey
B.S.E.E., Wright State University, 2019

2020
Wright State University

Wright State University
COLLEGE OF GRADUATE STUDIES

August 25, 2020

I HEREBY RECOMMEND THAT THE THESIS PREPARED UNDER MY SUPERVISION BY Julian L. Casey ENTITLED ANALYTICAL APPROACH TO MULTI-OBJECTIVE JOINT INFERENCE CONTROL FOR FIXED WING UNMANNED AERIAL VEHICLES BE ACCEPTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF Master of Science in Engineering.

Luther Palmer III, Ph.D.
Thesis Director

Fred Garber, Ph.D, Ph.D.
Interim Chair of Electrical Engineering

Committee on
Final Examination

Luther Palmer III, Ph.D

Xiaodong Zhang, Ph.D

Pradeep Misra, Ph.D

Trevor J. Bihl, Ph.D.

ABSTRACT

Casey, Julian. M.S.E.E., Department of Electrical Engineering, Wright State University, 2020. *ANALYTICAL APPROACH TO MULTI-OBJECTIVE JOINT INFERENCE CONTROL FOR FIXED WING UNMANNED AERIAL VEHICLES*.

Fixed-wing Unmanned Aerial Vehicles (UAVs) have been found highly useful in various environments, including military and law enforcement. With the increased use of fixed-wing UAVs, there becomes an increased need to optimize the resources available. One approach to resource management is to create multi-objective flights. This thesis presents the design, analysis, and experimental implementation of multi-objective resource management for the resource of Range, distance available to the UAV, from the viewpoint of Intelligence Surveillance and Reconnaissance (ISR). First, a Simulation Environment is created capable of tracking multiple fixed-wing UAVs and to allow for the UAVs' being controlled by an externally driven algorithm. Second, an Inference algorithm is developed with the objective of *information seeking*. Several algorithms are developed and used in conjunction with a Sequential Analysis test to allow for calculating Target Value, calculating Target Confidence, and validating the calculated Target Value. Third, a Control algorithm is developed with the objective of *Target seeking*. The Control algorithm uses several approaches to path generation, including Dubins path, Optimized Order path, and Closest Target path. Finally, a supervisor algorithm termed Joint Inference and Control (JIC) joins Inference and Control together. Monte Carlo simulated test flight results are shown to illustrate the effectiveness of the developed algorithms.

Contents

1	INTRODUCTION	1
2	SIMULATION ENVIRONMENT	5
2.1	Environment management	5
2.2	Requirements	6
2.3	Trajectory calculation	6
2.4	Restrictions	8
3	INFERENCE	10
3.0.1	Variance	11
3.0.2	Target Value	12
3.0.3	Target Confidence	13
3.1	Wald's Sequential Analysis test	14
3.1.1	Wald's Sequential Analysis test example	16
3.2	Expanding Wald's Sequential Analysis test	18
3.2.1	Sequential Analysis test example	18
3.2.2	Calculating Target Confidence	21
4	CONTROL	24
4.1	Dubins Path	25
4.1.1	3D translation of 2D Dubins path	27
4.1.2	Control with Dubins Path	27
4.2	Optimized Order	29
4.2.1	Control with Optimized Order Path	29
4.3	Closest Target Path	30
4.3.1	Control with Closest Target Path	30
5	JOINT INFERENCE and CONTROL (JIC)	31
5.1	Results	31
5.1.1	γ equals 0	33
5.1.2	γ equals 1	38
5.1.3	Target Value calculation	39

5.1.4	JIC Two Targets	40
5.1.5	JIC Three Targets	49
5.1.6	JIC Four Targets	58
6	CONCLUSION AND FUTURE WORK	67
	Bibliography	69

List of Figures

3.1	Noisy test signal	17
3.2	Sequential Analysis test	17
3.3	Noisy test signal with variance as function of distance	19
3.4	Calculated Target Value (weighted mean)	20
3.5	Sequential Analysis Test with variance as a function of distance	21
4.1	Dubins path	26
5.1	Simulated flight trajectory with two Targets where $\gamma = 0$ (Inference only)	34
5.2	Influence of individual Targets on Inference Algorithm when $\gamma = 0$	35
5.3	Simulated flight trajectory with two Targets where $\gamma = 0$ (Inference only) where $target'_1$ s Value=8 and $target'_2$ s Value=1	36
5.4	Influence of individual Targets on Inference Algorithm when $\gamma = 0$ and $target_1$ has known Value of 8	37
5.5	Simulated flight trajectory with two Targets where $\gamma = 1$ (Control only)	38
5.6	Value calculation during simulation	39
5.7	Simulated flight trajectory with two Targets when all axis are bound from 0 km to 5 km	41
5.8	Simulated flight trajectory with two Targets when all axis are bound from 0 km to 10 km	42
5.9	Simulated flight trajectory with two Targets when all axis are bound from 0 km to 20 km	43
5.10	Simulated flight trajectory with two Targets when all axis are bound from 0 km to 50 km	44
5.11	Simulated flight trajectory with two Targets when all axis are bound from 0 km to 100 km	45
5.12	Simulated flight trajectory with two Targets when all axis are bound from 0 km to 200 km	46
5.13	Mean comparison of two Target simulations	47
5.14	Peak savings comparison of two Target simulations	48
5.15	Simulated flight trajectory with three Targets when all axis are bound from 0 km to 5 km	50

5.16	Simulated flight trajectory with three Targets when all axis are bound from 0 km to 10 km	51
5.17	Simulated flight trajectory with three Targets when all axis are bound from 0 km to 20 km	52
5.18	Simulated flight trajectory with three Targets when all axis are bound from 0 km to 50 km	53
5.19	Simulated flight trajectory with three Targets when all axis are bound from 0 km to 100 km	54
5.20	Simulated flight trajectory with three Targets when all axis are bound from 0 km to 200 km	55
5.21	Mean comparison of three Target simulations	56
5.22	Peak savings comparison of three Target simulations	57
5.23	Simulated flight trajectory with four Targets when all axis are bound from 0 km to 5 km	59
5.24	Simulated flight trajectory with four Targets when all axis are bound from 0 km to 10 km	60
5.25	Simulated flight trajectory with four Targets when all axis are bound from 0 km to 20 km	61
5.26	Simulated flight trajectory with four Targets when all axis are bound from 0 km to 50 km	62
5.27	Simulated flight trajectory with four Targets when all axis are bound from 0 km to 100 km	63
5.28	Simulated flight trajectory with four Targets when all axis are bound from 0 km to 200 km	64
5.29	Mean comparison of four Target simulations	65
5.30	Peak savings comparison of four Target simulations	66

List of Tables

5.1	Statistical analysis of two Targets where all axis are bound from 0 km to 5 km	41
5.2	Statistical analysis of two Targets where all axis are bound from 0 km to 10 km	42
5.3	Statistical analysis of two Targets where all axis are bound from 0 km to 20 km	43
5.4	Statistical analysis of two Targets where all axis are bound from 0 km to 50 km	44
5.5	Statistical analysis of two Targets where all axis are bound from 0 km to 100 km	45
5.6	Statistical analysis of two Targets where all axis are bound from 0 km to 200 km	46
5.7	Statistical analysis of two Targets, mean savings	47
5.8	Statistical analysis of two Targets, peak savings	48
5.9	Statistical analysis of three Targets where all axis are bound from 0 km to 5 km	50
5.10	Statistical analysis of three Targets where all axis are bound from 0 km to 10 km	51
5.11	Statistical analysis of three Targets where all axis are bound from 0 km to 20 km	52
5.12	Statistical analysis of three Targets where all axis are bound from 0 km to 50 km	53
5.13	Statistical analysis of three Targets where all axis are bound from 0 km to 100 km	54
5.14	Statistical analysis of three Targets where all axis are bound from 0 km to 200 km	55
5.15	Statistical analysis of three Targets, mean savings	56
5.16	Statistical analysis of three Targets, peak savings	57
5.17	Statistical analysis of four Targets where all axis are bound from 0 km to 5 km	59
5.18	Statistical analysis of four Targets where all axis are bound from 0 km to 10 km	60

5.19	Statistical analysis of four Targets where all axis are bound from 0 km to 20 km	61
5.20	Statistical analysis of four Targets where all axis are bound from 0 km to 50 km	62
5.21	Statistical analysis of four Targets where all axis are bound from 0 km to 100 km	63
5.22	Statistical analysis of four Targets where all axis are bound from 0 km to 200 km	64
5.23	Statistical analysis of four Targets, Optimal mean savings	65
5.24	Statistical analysis of four Targets, peak savings	66

Acknowledgments

First and foremost, I would like to express my gratitude to Dr. Luther Palmer III for his guidance and insight into the development of this research.

I wish to thank my thesis committee members Dr. Xiaodong Zhang, Dr. Pradeep Misra, and Dr. Trevor J. Bihl for their time and feedback, it is greatly appreciated.

I wish to also express my gratitude towards Dr. Luther Palmer III and Dr. Xiaodong Zhang, for all of their challenging and enjoyable lectures I was able to attend. They have inspired and fueled a passion within me towards the area of controls.

INTRODUCTION

Unmanned Aerial Vehicles (UAVs) have received a significant amount of attention over the past decade. To this day, UAV applications continue to increase. These applications cover a wide range from agriculture to military and law enforcement. In the area of Intelligence Surveillance and Reconnaissance (ISR), the use of UAVs has become the preferred choice for many reasons. With the increased use of UAVs, there is an increased need to optimize the resources available to the UAV.

Inspired by the challenge of resource management, this research aims towards the design, analysis, and experimental implementation of multi-objective Joint Inference and Control (JIC) while managing the resource of Range available to the UAV by reducing the required amount of distance traveled. While the resource management focus of this research is the resource of Range available to a UAV, using less of the available Range provides the additional benefits of using less time in-flight as well as less Range-driven maintenance and fuel usage. This research focuses on resource management for the area of ISR.

First, a Simulation Environment is created that will allow for tracking of multiple fixed-wing UAVs as well as being capable of controlling the fixed-wing UAVs by externally run algorithms. In the simulation environment there exists one fixed-wing UAV used for the simulated ISR tasks, subsequently referred to as the *ISR vehicle* along with n number of fixed-wing UAV points of interest, subsequently referred to as *Targets*. The ISR vehicle trajectory is calculated through the use of tunable kinematic guidance equations allowing

for the ability to model any fixed-wing UAV. As well, the ISR vehicle uses onboard sensors to collect data about Targets. Targets are assigned a Value ranging from [1 10] representing their priority level where 1 is the lowest and default priority level and 10 is the highest level. Additionally, Targets are assigned a Confidence level ranging from [0 1], denoting the lowest and highest achievable Target Confidence in the validated Target Value.

Second, two objective independent algorithms are developed: (1) An Inference algorithm is designed with the objective of *Information seeking*: (2) A Control algorithm is designed with the objective of *Target seeking*.

Within the Inference algorithm, the calculation of a Target's Value, validation of the calculated Target's Value, and calculation of the Target Confidence take place. To allow for additional tune-ability, a **varying-variance** is used when the ISR vehicle collects data about Targets from its onboard sensors. The varying-variance is defined as a function of distance to resemble the Signal to Noise Ratio (SNR) seen when collecting data from any sensor. Where the general characteristics of any SNR are such that the further the sensor is from the source, the lower the SNR is, meaning the less accurate the reading is due to the more noise and variance that is introduced into the data.

Several difficulties that arise and are addressed in this thesis due to using a varying-variance include:

- Developing an algorithm to calculate Target Value's which takes into account the effect of variance as a function of distance [1].
- Development of a method of validating the calculated Target Value.
- Development of a method of calculating Target Confidence.

Within the Control algorithm, three different approaches to path generation are used showing the efficiency of resource management when Control is joined with Inference. The first approach is a widely used and adapted method of creating an ideal shortest path, known as Dubins path [2]. The second approach is termed Optimized Order, where the order of Targets that produces the shortest distance required, is used in conjugation with the kinematic guidance equations in the simulation environment. The final approach is termed Closest Target, where the Control algorithm is used in conjugation with the kinematic guidance equations built into the simulation environment to drive the ISR vehicle to seek the closest target.

Finally, a supervisor algorithm is developed to join Inference and Control together. Along with a weighting parameter designed to adjust the amount of influence that Inference and Control will have on the supervisor algorithm, Monte Carlo experimental flight simulation results are shown to illustrate the effectiveness of the proposed methods.

My research presented in this thesis comes from a larger project that is in its early development stage. The purpose of this thesis is to show the basic operation and foundation of JIC through the development of basic algorithms, for Inference, Control, and JIC. Developing JIC in a basic algorithmic way allows for analysis of the effectiveness of JIC at this stage of project development. Future work, from this thesis, involves increasing the complexity of the Inference, Control, and JIC algorithms along with increasing the complexity of the project. The layout of this thesis is as follows:

- **Chapter 2** includes the design and development of the Simulation Environment along with the requirements and restrictions.
- **Chapter 3** includes the development of the Inference algorithm, Target Value calculation, Target Confidence calculation, and calculated Target Value validation.
- **Chapter 4** includes the development of the Control algorithm as well as exploring three different approaches to path generation allowing for a more robust analysis of

resource management when Control is joined with Inference.

- **Chapter 5** includes the supervisor algorithm which joins Inference and Control together, termed Joint Inference Control (JIC). This chapter also includes the results from simulated flights, illustrating the effectiveness of the developed algorithms with the presence of two, three, and four Targets.

SIMULATION ENVIRONMENT

A simulation environment is created to test the efficiency of the algorithms presented in this thesis through the use of MATLAB®. The simulation environment resembles an unbounded three-dimensional space representing the normal operating environment of a fixed-wing UAV. Within the environment there exists one fixed-wing UAV used for the simulated ISR tasks, subsequently referred to as the *ISR vehicle* along with n number of fixed-wing UAV points of interest, subsequently referred to as *Targets*. Targets are assigned a Value from [1 10] which represents their priority level. The simulated tasks involve placing Targets at random locations while the JIC algorithm drives the trajectory of the ISR vehicle. Simulations are conducted using two, three, and four Targets while increasing the size of the area that Targets can occupy from 5 km to 200 km per axis. Within the simulation environment, the location of the ISR vehicle and the Targets is continuously tracked.

2.1 Environment management

Before a simulation is run the following parameters must be set:

- The number of fixed-wing UAVs in the environment.
- The initial location of the ISR vehicle and Targets.
- The ISR vehicle's velocity.

- The initial ISR vehicle's course.
- Simulation run time.

When a simulation is running, the ISR vehicle's and Targets' north and east coordinates are returned to allow for the resource management algorithms to calculate the next desired heading. Consequently, the next desired heading is then fed back into the simulation. This process continues until the remainder of the simulation run time.

2.2 Requirements

The Simulation Environment replicates realistic flight conditions by adhering to the following requirements:

- The environment shall be three-dimensional restricted only by the positive z-axis, representing the height above ground level.
- The number of fixed-wing UAVs in the environment capable of being set by the user, or at random within a predetermined range.
- The location of fixed-wing UAVs in the environment capable of being set by the user, or at random within a predetermined range.
- The environment shall be able to support multiple types of UAVs.
- The environment shall be capable of tracking the location of all UAVs in real-time.

2.3 Trajectory calculation

There are two options for calculating the trajectory of vehicles in the simulation environment – the use of kinematic or dynamic equations [3]. Kinematic equations focus only on

the motion of an object, where dynamic equations focus on the motion of an object as well as the forces and torques applied to the object.

Kinematic equations are chosen for this work as this allows for a more universal approach to the project by not requiring the use of specific information modeled after one particular UAV. The vehicle's trajectory is calculated and created primarily through the use of following kinematic equations [3], where it is assumed that the autopilot controls airspeed, altitude, and course angle.

$$\dot{P}_n = V_a \cos \psi + w_n \quad (2.1)$$

$$\dot{P}_e = V_a \sin \psi + w_e \quad (2.2)$$

$$\ddot{\chi} = b_{\dot{\chi}}(\dot{\chi}^c - \dot{\chi}) + b_{\chi}(\chi^c - \chi) \quad (2.3)$$

$$\ddot{h} = b_{\dot{h}}(\dot{h}^c - \dot{h}) + b_h(h^c - h) \quad (2.4)$$

Where $\dot{P}_n$ is the vehicle velocity in the north direction and w_n is the north directional wind velocity. $\dot{P}_e$ is the vehicle velocity in the east direction and w_e is the east directional wind velocity. V_a is the vehicle velocity, $\ddot{\chi}$ is the course acceleration, and $\ddot{h}$ is the height acceleration. The parameters $b_{\dot{\chi}}$, b_{χ} , $b_{\dot{h}}$, and b_h are tune-able gains chosen to represent the in flight characteristics specific to the modeled vehicle. The inputs are the commanded altitude h^c and the commanded course χ^c .

Further construction of trajectory calculations are completed by taking the above kinematic equations and using Euler's method [4] to result in the following equations:

$$P_n(k+1) = P_n(k) + \dot{P}_n(k)dt \quad (2.5)$$

$$P_e(k+1) = P_e(k) + \dot{P}_e(k)dt \quad (2.6)$$

$$\dot{h}(k+1) = \dot{h}(k) + \ddot{h}(k)dt \quad (2.7)$$

$$h(k+1) = h(k) + \dot{h}(k)dt \quad (2.8)$$

$$\dot{\chi}(k+1) = \dot{\chi}(k) + \ddot{\chi}(k)dt \quad (2.9)$$

$$\chi(k+1) = \chi(k) + \dot{\chi}(k)dt \quad (2.10)$$

Where k is the computation number in the simulation and dt is the time between computations. Euler's method [4] provides an approximation to solving the differentiation equations (2.1-2.4) and can be seen to provide a near accurate approximation when dt is a very small number. Due to the absence of wind speed, the heading becomes the course

$$\psi(k+1) = \chi(k+1) \quad (2.11)$$

the commanded course is defined as

$$\chi^c(k) = \text{atan2}(a, b) \quad (2.12)$$

where a represents the difference from the Joint Inference and Control (JIC) algorithm calculated desired east coordinate and the vehicle's east coordinate and b represents the difference from the JIC algorithm calculated desired north coordinate and the vehicle's north coordinate. This is due to kinematic 0 degrees pointing North.

2.4 Restrictions

- There is an absence of wind.
- The ISR UAV velocity is set at constant cruising speed.
- All non-ISR UAVs are in-flight in a fixed position.

These restrictions can easily be modified as the complexity of the problem increases. They are set in place at this time to get the basic operation of the stimulation environment working, to allow for developing and testing the resource management algorithms.

INFERENCE

Inference is first defined to be an *information seeking* algorithm. To develop the Inference algorithm Targets are assigned both a value for Target Value and a value for Target Confidence. Target Value takes on a range from [1 10] denoting the priority level and Target Confidence takes on a range of [0 1] denoting the Confidence in the validated Target Value. The Inference algorithm will take on different definitions depending on the current resource to be managed. For this research, the focus of resource management is solely on managing the resource of Range available to the ISR vehicle, by reducing the amount of distance required to travel. At any point in time, the only information available to the Inference algorithm is the location of the Target provided from the simulation environment, the calculated distance from the ISR vehicle to the Target, the calculated Target Value, and the calculated Target Confidence. Using the information available the Inference algorithm is defined as

$$I_{t_n} = \hat{t}_n \left(\frac{V_{t_n} d_{t_n}}{1 + c_{t_n}} \right) \quad (3.1)$$

where, $\hat{t}_n$ is the unit vector in the direction from the ISR vehicle to the n^{th} Target, V_{t_n} is the Target Value, c_{t_n} is the Target Confidence, and d_{t_n} is the distance from the vehicle to the n^{th} Target.

With the presence of multiple Targets in the environment, the Inference algorithm then becomes the sum of individual Inference values (3.2).

$$I = \sum_{n=1}^m I_{t_n} \quad (3.2)$$

For m number of Targets

When the ISR vehicle enters sensor range with a Target, it needs to begin calculating the Target's Value, while simultaneously validating the calculated Target Value. Validating the calculated Target Value is completed through the use of a sequential analysis test developed by Wald [5] and expanded upon by Siegmund [6, 7], Lai [8], and Kim [9]. Target Confidence is calculated from the sequential analysis test, as the test validates the calculated Target Value.

3.0.1 Variance

In statistics, variance [10] is defined as the measurement of the spread between numbers in a data set and can be seen in equation (3.3).

$$S^2 = \frac{\sum (X - \bar{x})^2}{n - 1} \quad (3.3)$$

X takes on the individual data points and $\bar{x}$ is the mean of the data set and n is the sample size.

Varying variance

This research explores the use of a varying-variance. Varying-variance is a function of distance to resemble the signal-to-noise ratio (SNR) seen in any sensor. SNR represents the ratio of the strength of a signal being measured to that of the interference. As the

distance from the source increases, the variance, noise, in the data increase. The varying-variance is applied to both the data received during simulations and the variance used for the sequential analysis test. Varying-variance becomes a tune-able parameter allowing for future work to design a specific sequential analysis test for a specific ISR vehicle sensor. For this thesis, varying-variance is defined as equation (3.4),

$$\text{varying-variance} = t_d \quad (3.4)$$

where t_d is the distance to the target.

3.0.2 Target Value

In general, Target Value is a numerical representation of the Targets' priority levels determined by an external algorithm after analyzing the data collected from onboard sensors of the ISR vehicle. One example can be an onboard camera [11] where the image is analyzed by an external algorithm that determines its priority level and is translated into a Target Value. The external prioritizing algorithm is not a focus of this work; it is mentioned as a way of understanding where a Target's Value originates. The Value of a Target takes on a range of [1 10] , where 10 denotes the highest priority and 1 is both the default and lowest priority. A Target's Value is unknown until the ISR vehicle is within sensor range to the Target, where the ISR vehicle begins to calculate the Target's Value.

Calculating Target Values

Target Values calculated through the use of a *weighted mean* [12, 13]. This allows for a tune-able parameter that lessens the effect of low signal-to-noise Ratio (SNR) signals on the Target Value calculation process, which could otherwise cause an incorrect Target Value to be calculated. Given any sensor's SNR, the weighted mean can be designed to give priority

to sensor readings that contain higher SNR over readings containing a lower SNR. This is done by modeling the weighting function w to match the ISR vehicles onboard sensor's SNR. The weighted mean is defined as:

$$\text{weightedMean}(i) = \frac{w(i)V_r(i)}{\sum w} \quad (3.5)$$

where V_r is the Target Value received from the external analyzing algorithm and w is the weighting function modeled to resemble the SNR for an ISR vehicle's onboard sensor. This thesis uses equation (3.6) to determine the SNR [14] resulting in the definition of w as equation (3.7).

$$SNR = \frac{\mu}{\sigma} \quad (3.6)$$

$$w = SNR = \frac{\mu}{\sigma} \quad (3.7)$$

Where μ is the mean and σ is the standard deviation of the signal received from the ISR vehicle onboard sensor.

3.0.3 Target Confidence

Target Confidence takes on a range of [0 1]. Where 1 denotes the highest level of confidence achievable and 0 is the lowest level of confidence achievable. The value of Target Confidence is calculated through the result of the sequential analysis test, while the test is validating the calculated Target Value. Calculation of Target Confidence is seen in greater detail in section 3.2.2, after the sequential analysis test is defined. With the Inference algorithms' objective set as information-seeking, Confidence needs be a significant influencing factor. Confidence is placed in the denominator of the individual Target's Inference value such that the lower the value of Confidence, the more influence the individual Target has on the Inference algorithm. The case of Confidence = 0, represents the most unknown information and needs to have the most influence on Inference. Since the lowest value of

Confidence is 0, the addition of 1 is added to the denominator of the individual Inference calculation (3.1) to prevent division by 0 scenarios.

Target Confidence decay

The moment the ISR vehicle exits sensor range with a Target, the value for Target Confidence begins to decay. The decay of Target Confidence is in place to represent the real-life situation where unless a Target is being observed its actions are not known. At any point, the Target's priority level can change, changing the Target's Value. The more time that passes when the ISR vehicle is outside of sensor range with a Target, the more the Target Confidence decays. Decay is a tune-able time-based function that is defined specific to the environment the ISR vehicle is in. Decay is applied as equation (3.8)

$$\text{Target Confidence}(i + 1) = D_c * \text{Target Confidence}(i) \quad (3.8)$$

where D_c is a function of time weighting term which takes on different definitions, depending on the environment the ISR vehicle is in. Future work for this project will involve the design of environment specific decay functions. This thesis uses the definition of $D_c = 0.999$ to provide decay to the Target Confidence during simulations.

3.1 Wald's Sequential Analysis test

Wald [5] gives us a way to test the mean of a normal distribution to a **known** Target Value, with a known **fixed** variance (3.3). The equation for testing an assumed Target Value is given as

$$\frac{\sigma}{\delta} \phi(\log B + m \frac{\delta^2}{2}) < |\sum (\chi_\epsilon - \theta_0)| < \frac{\sigma}{\delta} \phi(\log A + m \frac{\delta^2}{2}) \quad (3.9)$$

where σ is the standard deviation of the signal, δ is the allowance of error in the test, m is the quantity of signals received, χ_ϵ is the current signal being tested, θ_0 is the Target Value being tested against, A represents the relationship $\frac{1-\zeta}{\eta}$, and B represents the relationship $\frac{\zeta}{1-\eta}$.

Wald also allows for the addition of weighting terms that bias the test to either be more allowing of a false acceptance ζ or false rejection η . False acceptance is the case that the Target Value accepted is the incorrect Target Value and false rejection is the case that the correct Target Value is rejected.

In equation (3.9), ϕ is the function defined as (3.10).

$$\phi(\nu) = \log(e^\nu + \sqrt{e^{2\nu} - 1}) \quad (3.10)$$

From equation (3.9) an assumed Target Value is accepted (not rejected) when the following relationship is a true statement,

$$|\sum(\chi_\epsilon - \theta_0)| < \frac{\sigma}{\delta} \phi(\log A + m \frac{\delta^2}{2}). \quad (3.11)$$

defining the acceptance (non-rejection) threshold as equation (3.12).

$$\frac{\sigma}{\delta} \phi(\log A + m \frac{\delta^2}{2}) \quad (3.12)$$

Equation (3.9) also shows that an assumed Target Value is rejected when the following relationship is a true statement,

$$\frac{\sigma}{\delta} \phi(\log B + m \frac{\delta^2}{2}) < |\sum(\chi_\epsilon - \theta_0)|. \quad (3.13)$$

Defining the rejection threshold as equation (3.14).

$$\frac{\sigma}{\delta} \phi(\log B + m \frac{\delta^2}{2}) \quad (3.14)$$

Along with the testing summation defined as equation (3.15).

$$|\sum (\chi_{\epsilon} - \theta_0)|. \quad (3.15)$$

The testing summation (3.15), is the absolute value of the sum of difference of every data point, χ_{ϵ} , and the assumed Target Value, θ_0 . When the testing summation is greater than the rejection threshold (3.14), the test ends concluding that the assumed Target Value, θ_0 , is not the actual Target Value within the δ range of error. When testing summation is less than the acceptance (non-rejection) threshold (3.12), the test ends concluding that the assumed Target Value, θ_0 , is the actual Target Value within the δ range of error.

3.1.1 Wald's Sequential Analysis test example

This section is to demonstrate how Wald's [5] sequential analysis test works. Figure 3.1 shows 100 samples of noisy data collected, where the actual Target Value of the signal is 10, however, when the sample is collected the noise produces an effect seen as a variance of 2.

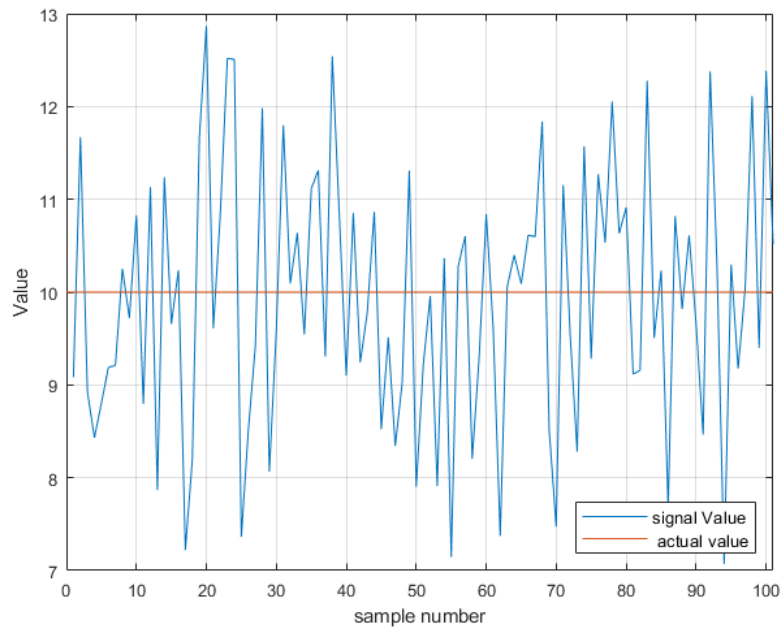


Figure 3.1: Noisy test signal

When the sample data from figure 3.1 is used with the Sequential Analysis test, the noisy signal is used for the value of χ_ϵ and the actual Target Value of 10 is used for θ_0 . The results of this test can be seen in figure 3.2.

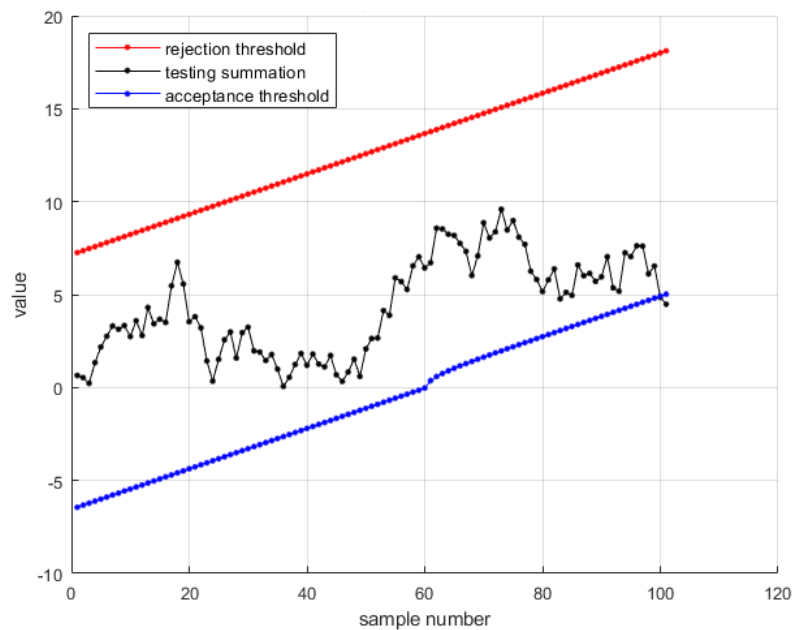


Figure 3.2: Sequential Analysis test

Figure 3.2 shows the resulting Sequential Analysis test from the noisy data collected from figure 3.1. For this test δ , the amount of error allowance in the test, is set at 0.5. The bias factor for false rejection, η , and the bias factor for false acceptance, ζ , are set at near-zero values to make the test unbiased. However, increasing the η and ζ values would effectively lower the rejection threshold and raise the acceptance (non-rejection) threshold respectively. At every sample, the testing summation value is the absolute value of the sum of error between the Target Value, θ_0 and the sampled signal, χ_ϵ . The thresholds calculations are designed having weighting terms including the amount of allowance for error, δ , and the standard deviation, σ , of the sampled signal, χ_ϵ , which create boundary lines around the testing summation. When the testing summation, amount of error between the sampled signal and actual Target Value, goes outside of the boundary lines then the test ends. It can be seen in figure 3.2 at sample number 100 the testing summation crosses the acceptance (non-rejection) threshold. At this point, equation (3.12) holds concluding that the assumed Target Value, θ_0 , is the actual Target Value within the δ range of error.

3.2 Expanding Wald's Sequential Analysis test

My research proposes using a variance that is a function distance to resemble the inherent SNR ratio characteristics found in any sensor. Variance as a function of distance will replace the *fixed variance* used in the sequential analysis test [6, 7, 8, 9]. Additionally in place of the **known** Target Value, θ_0 , is the calculated Target Value (weighted mean) [12].

3.2.1 Sequential Analysis test example

This section is to demonstrate how the sequential analysis test works with variance as a function of distance and Target Value being a weighted mean. Figure 3.3 shows 100 samples of noisy data collected. Where the actual Target Value is 5, however, when the

sample is collected, the noise produces an effect seen as a variance which is a function of distance.

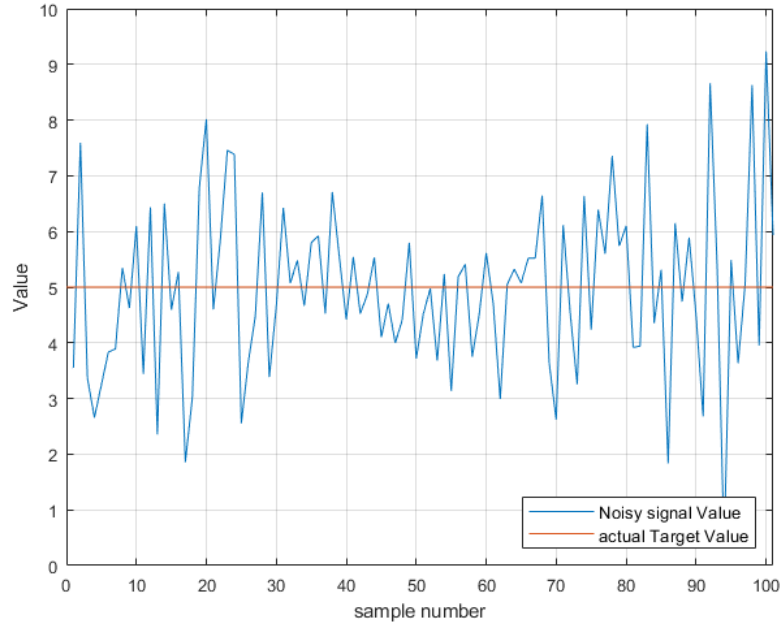


Figure 3.3: Noisy test signal with variance as function of distance

When the sample data from figure 3.3 is used with the sequential analysis test, the noisy signal is used for the value of χ_ϵ . The value used for θ_0 is the calculated Target Value resulting from applying the noisy signal from figure 3.3 to equation (3.5). The results of calculating the Target Value can be seen in figure 3.4.

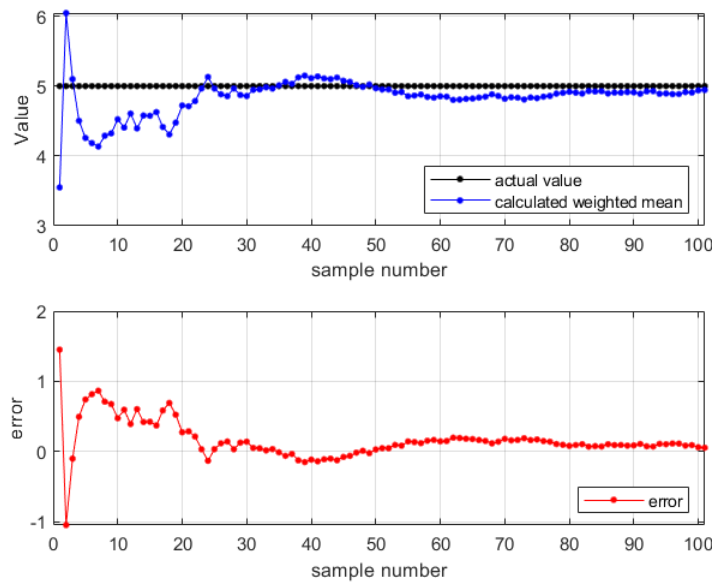


Figure 3.4: Calculated Target Value (weighted mean)

Figure 3.4 shows the results of the calculated Target Value and the error between calculated and actual Target Value with respect to the sample number. While there is a steady-state error of 0.125 in the calculated Target Value, remember that Target Values take on integer values. At every sample of data, the Target's Value is calculated and the calculated Target Value is fed into the sequential analysis test producing figure 3.5. When the testing summation crosses the acceptance (non-rejection) threshold of the sequential analysis test, the calculated Target Value becomes *validated*. The validated Target Value is the calculated Target Value corresponding to the same sample number after it is rounded to the nearest integer value.

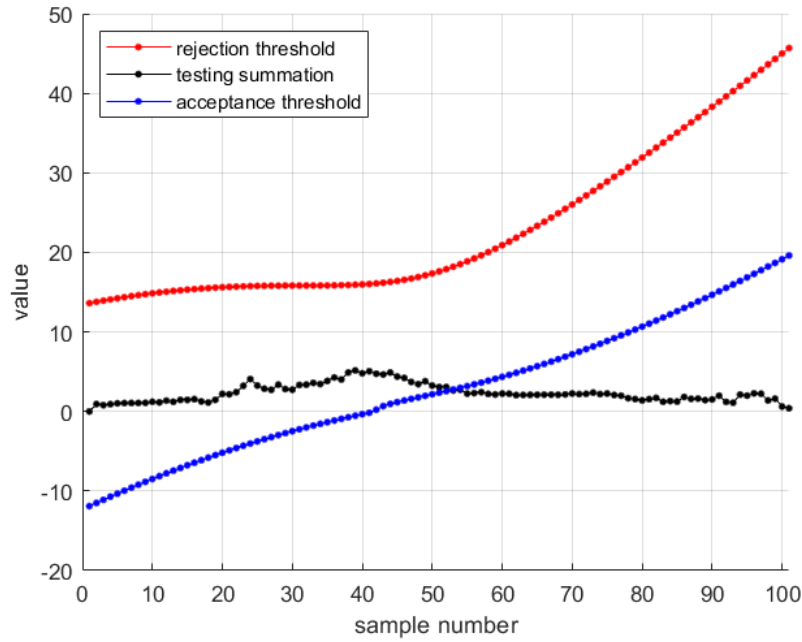


Figure 3.5: Sequential Analysis Test with variance as a function of distance

Figure 3.5 shows the result of the sequential analysis test with a variance as a function of distance. For this test δ , the amount of error allowance in the test, is set at 0.5. The bias factor for false rejection, η , and the bias factor for false acceptance, ζ , are set at near-zero values to make the test unbiased. It can be seen that at approximately sample number 55, the testing summation crosses the acceptance (non-rejection) threshold. At this point, the test fails to reject the calculated Target Value, validating the calculated Target Value. Looking at sample number 55 in figure 3.4, the calculated Target Value is 4.85, which is rounded to the nearest integer value of 5. At this time the sequential analysis test validates the calculated Target Value to be a Value of 5, which is the same as the actual Target Value of 5.

3.2.2 Calculating Target Confidence

The method of calculating Target Confidence for the validated Target Value used in this thesis is derived directly from the resulting modified Sequential Analysis Test 3.2. When the

testing summation crosses the acceptance (non-rejection) threshold, the calculated Target Value is validated. When the testing summation crosses the rejection threshold, the calculated Target Value is rejected. Using the acceptance (non-rejection) and rejection thresholds as boundary lines [6, 7], Target Confidence is derived to be the percentage of the distance of the test summation to the acceptance (non-rejection) threshold, with respect to the overall distance from the rejection threshold to the acceptance (non-rejection) threshold. The percentage is then converted into a range of [0 1].

Confidence derivation

The distance between the acceptance (non-rejection) threshold (3.12) and the rejection threshold (3.14) can be found by taking the absolute value of their difference creating

$$|\frac{\sigma}{\delta}\phi(\log A + m\frac{\delta^2}{2}) - \frac{\sigma}{\delta}\phi(\log B + m\frac{\delta^2}{2})|. \quad (3.16)$$

The distance between the acceptance (non-rejection) threshold (3.12) and the testing summation (3.15) can be found by taking the absolute value of their difference creating

$$|\frac{\sigma}{\delta}\phi(\log A + m\frac{\delta^2}{2}) - |\sum(\chi_\epsilon - \theta_0)||. \quad (3.17)$$

Dividing (3.17) by (3.16) creates

$$\frac{|\frac{\sigma}{\delta}\phi(\log A + m\frac{\delta^2}{2}) - |\sum(\chi_\epsilon - \theta_0)||}{|\frac{\sigma}{\delta}\phi(\log A + m\frac{\delta^2}{2}) - \frac{\sigma}{\delta}\phi(\log B + m\frac{\delta^2}{2})|} \quad (3.18)$$

the ratio of percentage of distance of the test summation to the rejection threshold, with respect to the overall distance from the rejection threshold to the acceptance (non-rejection) threshold as a range of [0 1]. Taking the value of one and subtracting (3.18) from it, creates

$$Target\ Confidence = 1 - \frac{|\frac{\sigma}{\delta}\phi(\log A + m\frac{\delta^2}{2}) - |\sum(\chi_\epsilon - \theta_0)||}{|\frac{\sigma}{\delta}\phi(\log A + m\frac{\delta^2}{2}) - \frac{\sigma}{\delta}\phi(\log B + m\frac{\delta^2}{2})|} \quad (3.19)$$

the value of Target Confidence ranging [0 1]. 1 is the highest level of Target Confidence, seen when the testing summation crosses the acceptance (non-rejection) threshold during the modified sequential analysis test, validating the calculated Target Value.

CONTROL

The Control algorithm, at its core, is designed to be a *Target seeking* algorithm. It seeks Targets to intersect them. While there are many methods of path generation in use today, this thesis only views three basic methods of path generation to begin to discover the overall trend in the dynamics of multi-objective Joint Inference and Control (JIC). The project that the research presented in this thesis comes from is still in the early stage of development. The goal of this thesis is to present a **basic** foundation of JIC and its effectiveness using basic algorithms. Allowing for future work to involve adding additional methods of path generation.

With the overall goal of managing the resource of Range, as distance available per flight, the first approach is to generate an ideal shortest path. This starts with a well-known method of path generation known as Dubins path [2]. The final method of the first approach is called Optimized Order, where all permutations of Target orders are calculated and the resulting order correlating to the shortest overall distance required to travel is chosen. The second approach to path generation is the method of seeking the closest Target first, where the closest Target to the ISR vehicle becomes the Target the ISR vehicle seeks. Closest Target approach is added to allow for a wider range of dynamics of JIC to be presented in the results section.

4.1 Dubins Path

Dubins path has been a widely adapted method of generating the shortest path between two points in both two-dimensional [15] and three-dimensional space [16]. While work has been made to develop Dubins paths in three-dimensional space, a two-dimensional Dubins path was seen to be an efficient method of path generation with less computational cost than the three-dimensional counterpart. As well the two-dimensional path generation can be translated to three-dimensional space when applying a restriction that states: when the ISR vehicle intersects a Target, it must first match the Target's altitude. When the ISR vehicle is at the same altitude as a Target, a two-dimensional Dubins path can be directly applied. To create two-dimensional Dubins paths, the following conditions must be met:

- The path curvature is defined as the minimum in-flight turning radius of the ISR vehicle.
- The altitude of the ISR vehicle and the Targets is the same during the arc of intersection portion of the flight path.
- The ground speed of the ISR vehicle is set constant.

Given the minimum turning radius, starting position and course as well as the ending position and course between two points, four Dubins path cases are generated [3].

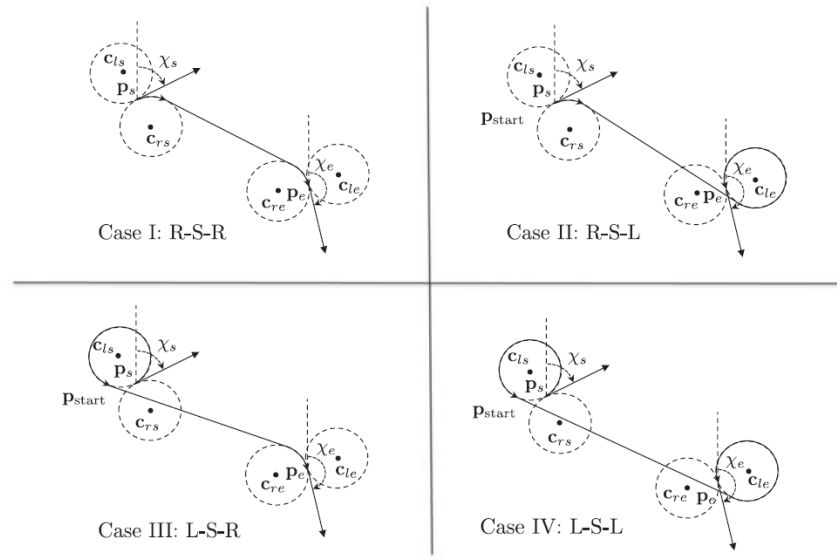


Figure 4.1: Dubins path

Figure 4.1 illustrates the four possible cases of generating a path between two points around a specified curvature, where $\mathbf{p}_s$ is the starting position, χ_s is the starting course, $\mathbf{C}_{ls}$ is the center of the left starting arc, and $\mathbf{C}_{rs}$ is the center of the right starting arc. $\mathbf{p}_e$ is the ending position, χ_e is the ending course, $\mathbf{C}_{le}$ is the center of the left ending arc, and $\mathbf{C}_{re}$ is the center of the right ending arc.

Case I: R-S-R, or right-straight-right, illustrates a path that begins with a right turn, goes straight, then makes a right turn ending at the second point aiming at the predetermined ending course angle.

Case II: R-S-L, right-straight-left, starts the path in the same way as Case I, however, the straight-line path connects to the arc required to make a left turn ending at the same predetermined ending point and ending course angle.

Case III: L-S-R, or left-straight-right, illustrates a path that begins with a left turn, goes straight, then makes a right turn ending at the second point aiming at the predetermined ending course angle.

Case IV: L-S-L, or left-straight-left, illustrates a path that begins with a left turn, goes straight, then makes a left turn ending at the second point aiming at the predetermined

ending course angle.

4.1.1 3D translation of 2D Dubins path

Two-dimensional Dubins path can be directly applied to find the shortest three-dimensional path when:

- The ISR vehicle is at the same altitude as the Target during the arc of intersection.
- The distance used to create the straight-line paths is the three-dimensional distances between the intersecting arcs.

This allows for using a two-dimensional method to find the shortest three-dimensional path. The three-dimensional coordinates are then used in the simulation environment for the ISR vehicle trajectory.

4.1.2 Control with Dubins Path

For every n number of Targets Control begins with Dubins path by calculating every permutation of Target order starting with its current location and ending with a specified ending location. For example, assuming there are three Targets, the Control algorithm then generates the possible flight permutations of the three Targets seen as:

1. Start-1-2-3-End
2. Start-1-3-2-End
3. Start-2-1-3-End
4. Start-2-3-1-End
5. Start-3-1-2-End
6. Start-3-2-1-End.

The number of permutations can be seen as

$$\text{number of permutations} = n! \quad (4.1)$$

for n number of Targets.

After the list of permutations is created, the four cases of Dubins path are then generated and the path with the shortest total distance is chosen to be the path the Control algorithm follows. When connecting multiple Dubins paths to generate the path to intersect all Targets it is possible to only need to connect every 2^{nd} and 3^{rd} location by a straight line. That is, there is no need to generate the four cases of Dubins path connecting the 2^{nd} and 3^{rd} location. The ending heading from 2^{nd} location simply becomes the starting heading for the 3^{rd} location. From there the four cases are then needed to go from the 3^{rd} location to the 4^{th} .

This translates to

$$\text{number of Dubins paths} = 4\left(\frac{m}{2}\right) \quad (4.2)$$

for m number of even locations and

$$\text{number of Dubins paths} = 4\left(\frac{m-1}{2}\right) \quad (4.3)$$

for m number of odd locations. Combining (4.1) and (4.2) we get

$$\text{Total number of Dubins paths} = (n!)4\left(\frac{m}{2}\right) \quad (4.4)$$

for n number of Targets and m number of even locations. Combining (4.1) and (4.3) we get

$$\text{Total number of Dubins paths} = (n!)4\left(\frac{m-1}{2}\right) \quad (4.5)$$

for n number of Targets and m number of odd locations. With the assumption that every flight will have a starting and ending location (4.4) and (4.5) can be further simplified to

$$\text{Total number of Dubins paths} = (n!)4\left(\frac{n+2}{2}\right) \quad (4.6)$$

for n number of even Targets and

$$\text{Total number of Dubins paths} = (n!)4\left(\frac{n+1}{2}\right) \quad (4.7)$$

for n number of odd Targets.

4.2 Optimized Order

Optimized Order is an array of Targets in the order that results in the lowest amount of overall distance traveled. Optimized Order starts by calculating all permutations of the Target order. It then calculates the total path distance for each permutation beginning at the starting location and terminating at the ending location. The resulting Optimized Order is the permutation correlating to the shortest total distance.

4.2.1 Control with Optimized Order Path

Control with Optimized Order works by feeding the Optimized Order into the simulation environment, using the kinematic guidance equations to create the path. Control with Optimized Order starts with the first Target in the Optimized Order array and guiding the ISR

vehicle to intersect the Target. After a Target is successfully intersected, the intersected Target is removed from the Optimized Order array and Control with Optimized Order now seeks the next Target in the array. This process continues until all Targets have been intersected and the ISR vehicle has reached the ending location.

4.3 Closest Target Path

The last approach to path generation is the approach of Closest Target, where the closest Target to the ISR vehicle becomes the Target the ISR vehicle seeks. This approach is added to allow for a wider range of dynamics of JIC to be presented in the results section.

4.3.1 Control with Closest Target Path

As the name suggests, the path is generated by always seeking the closest Target, where the location of the closest Target is fed into the simulation environment, using the kinematic guidance equations to create the path. After a Target has been intersected it is no longer evaluated when calculating the next closest target. This process continues until all Targets have been intersected, where the ISR vehicle will then travel to the ending location.

JOINT INFERENCE and CONTROL

(JIC)

Joint Inference and Control (JIC) combines the Control and Inference algorithms with a single weighting term γ and is defined as

$$J_{IC} = \gamma \hat{C} + (1 - \gamma) \hat{I} \quad (5.1)$$

where γ carries a value from $[0,1]$ and allows for adjusting the amount of weight on Control and Inference, $\hat{I}$ is the unit vector result of the Inference algorithm, and $\hat{C}$ is the unit vector result of the Control algorithm. When $\gamma=0$ is the case of Joint Inference and Control, only using Inference and the case of $\gamma=1$ resulting in Joint Inference and Control only uses Control.

5.1 Results

Assumptions made for the simulations are as follows:

- The ISR vehicle is a fixed-wing UAV with constant velocity set at 370 km per hour.
- All in-flight turns are made at the minimum turning radius of 0.270km.

- All Targets are at fixed in-flight locations.
- The variance in received sensor data is a function of distance where $variance = t_d$ and t_d is the distance to the Target from the ISR vehicle.

Note: The ISR vehicle velocity and minimum turning radius as well as the definition of variance can take on any value. These values were chosen to allow for simulation results.

5.1.1 γ equals 0

The case when $\gamma = 0$ illustrates the behavior of the Inference algorithm alone. The Inference algorithm, when used without the addition of the Control algorithm, is incapable of completing its information-seeking objective. Looking at equation (3.1) the Inference value of an individual Target,

$$I_{t_n} = \hat{t}_n \left(\frac{V_{t_n} d_{t_n}}{1 + c_{t_n}} \right)$$

it can be seen that when the ISR vehicle is unable to enter sensor range with a Target, the values for Value and Confidence remain at their default lowest setting of 1 and 0 respectively, an individual Targets Inference value becoming a function of distance. This results in an individual Target's Inference value being defined as

$$I_{t_n} = \hat{t}_n d_{t_n}, \tag{5.2}$$

where the only driving component of equation (5.2) is the distance from the Target to the ISR vehicle. Figure 5.1 illustrates this case.

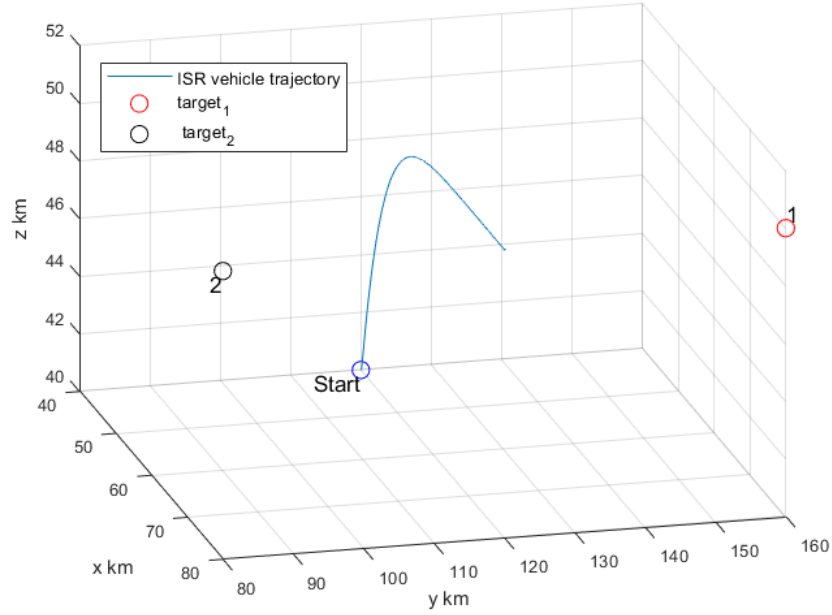


Figure 5.1: Simulated flight trajectory with two Targets where $\gamma = 0$ (Inference only)

Figure 5.1 shows the case where the starting location is $[40 \ 120 \ 40]$, $target_1$'s location is $[80 \ 160 \ 50]$ and $target_2$'s location is $[80 \ 80 \ 50]$. The locations were chosen to illustrate equation (5.2) Since the Inference algorithm is the sum of individual Target's Inference values, the Inference algorithm is driven to find the mid-point between the two Targets. As it reaches the mid-point it begins to fly in a circle, as the ISR vehicle gets closer to $target_1$, the distance increases to $target_2$ causing the Inference algorithm to enter an endless loop of changing which Target it wants to seek. This is illustrated in the figure 5.2.

Figure 5.2 shows the influence of individual Targets on the Inference algorithm when $\gamma=0$. When the Inference algorithm goes into the JIC algorithm, the unit vector value of Inference is used. This means the total magnitude of all Targets summed together equals one.

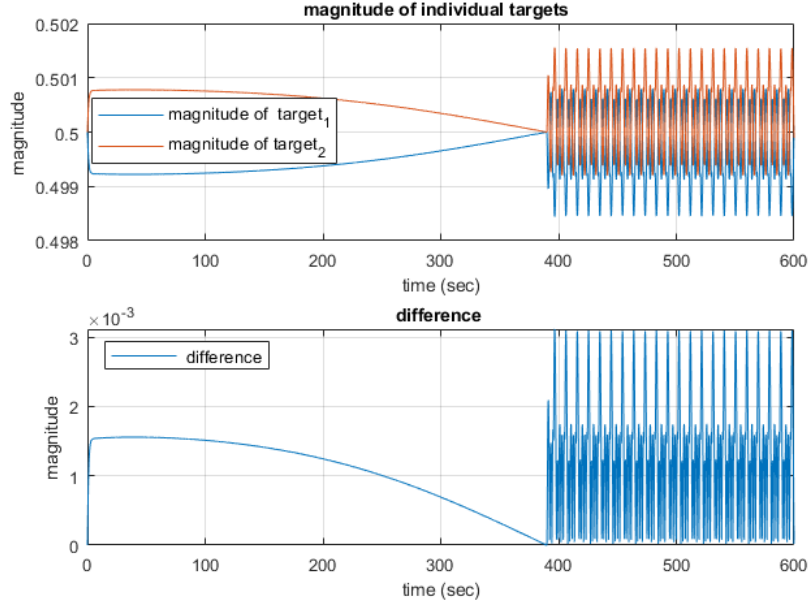


Figure 5.2: Influence of individual Targets on Inference Algorithm when $\gamma = 0$

It can be seen in figure 5.2, as the ISR vehicle reaches a point of equal distance to both $target_1$ and $target_2$ the difference in the magnitude of $target_1$ and $target_2$ reaches near-zero (note the y-axis scale on the difference plot being 10^{-3}). At this point, the Inference algorithm begins to rapidly change between which Target it wants to seek. Since the ISR vehicle cannot hover in place, it begins to travel in a circular trajectory attempting to settle at a mid-point. While figure 5.2 is an ideal scenario to show the dynamics of the Inference algorithm by having the Targets be equidistant from the starting position when reviewing equation (3.1) once more

$$I_{t_n} = \hat{t}_n \left(\frac{V_{t_n} d_{t_n}}{1 + c_{t_n}} \right)$$

it can be seen that whenever the ISR vehicle is not in sensor Range of a Target, even when its value has been discovered, it still becomes a function of distance. When the Targets are far enough away, the Inference algorithm still seeks a point between Targets where it begins to take on a circular trajectory as seen in figure 5.3.

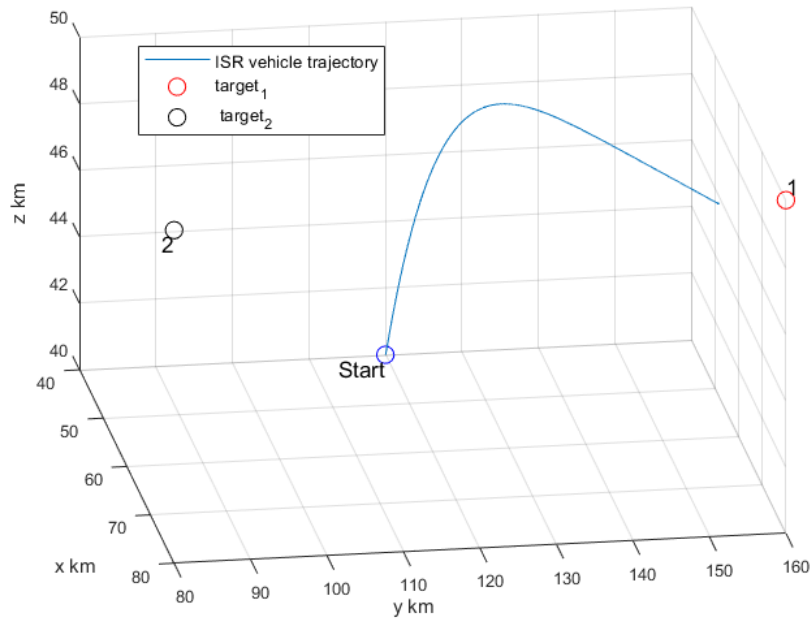


Figure 5.3: Simulated flight trajectory with two Targets where $\gamma = 0$ (Inference only) where $target'_1$'s Value=8 and $target'_2$'s Value=1

Figure 5.3 shows the case where the starting location is [40 120 40], $target'_1$'s location is [80 160 50] and $target'_2$'s location is [80 80 50], the same as figure 5.1. This time $target_1$ has a known Value of 8 and $target_2$ has a Value of 1. This illustrates the nature of the Inference algorithm. Unless all Targets are grouped where their distance to each other is no greater than the sensor Range, Inference will find a minima point, flying a circular trajectory. Figure 5.4 shows the influence of individual Targets on the Inference algorithm where $target_1$ has a known Value of 8 and $target_2$ has a default unknown Value of 1.

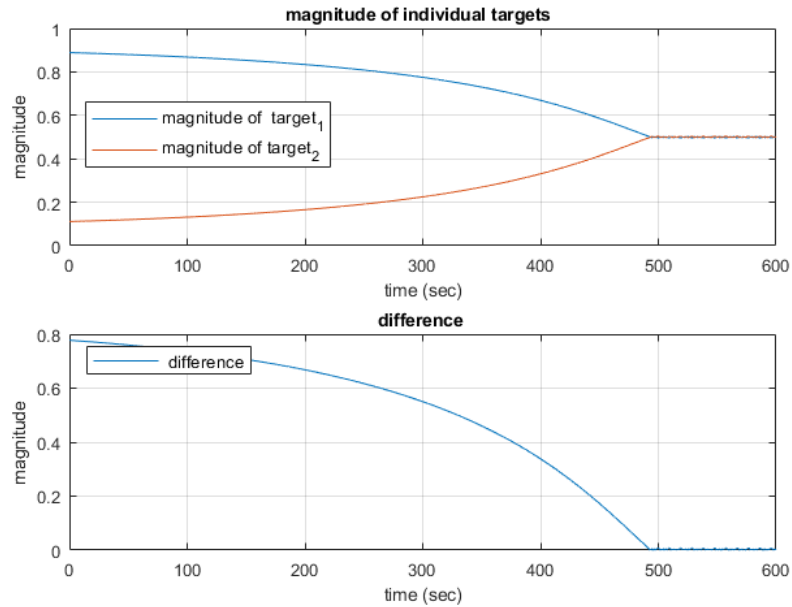


Figure 5.4: Influence of individual Targets on Inference Algorithm when $\gamma = 0$ and $target_1$ has known Value of 8

Figure 5.4 shows the influence of individual Targets on the Inference algorithm, where $target'_1$'s Value = 8 and $target'_2$'s Value is the default lowest value of 1. The start position and Target positions are the same as in figure 5.2 however, when comparing the two figures it can be seen when $target_1$ has a Value of 8 it takes approximately 85 seconds longer for the magnitude of the individual Targets to converge on 0.5. This is due to $target_1$ having a Value of 8 driving the Inference algorithm to seek $target_1$, yet $target_1$ is still significantly far enough away causing the ISR vehicle to begin a circular trajectory, this time much closer to $target_1$ than in the case of $target_1$ and $target_2$ both have Values equal to 1.

5.1.2 γ equals 1

For the case of $\gamma = 1$, the flight simulation illustrates the behavior of the Control algorithm alone. Where, unlike the Inference algorithm, when Control is used without the addition of Inference, it can complete its target-seeking objective.

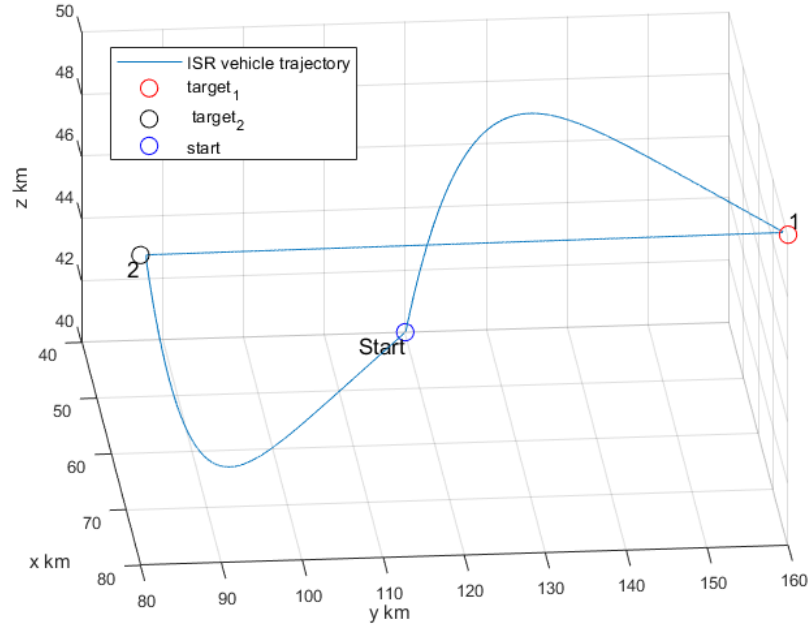


Figure 5.5: Simulated flight trajectory with two Targets where $\gamma = 1$ (Control only)

Figure 5.5 shows a simulated flight with two Targets where $\gamma = 1$, illustrating the behavior of the Control algorithm. The starting position is the same as the ending position set at [40 120 40], $target_1$'s location is [80 160 50] and $target_2$'s location is [80 80 50]. The Control algorithm is designed to seek Targets and intersect them and that is exactly what it is doing. The ISR vehicle flies to $target_1$ and intersects it, followed by flying to $target_2$ intersecting it and finishing with flying to the end location.

5.1.3 Target Value calculation

Figure 5.6 shows the calculation of $target_2^t$ s Value for the flight simulated in figure 5.5. Where the actual Value of $target_2$ is 8. It can be seen at about a time of 1296 seconds into the simulation the ISR vehicle enters sensor Range with $target_2$ and begins to calculate the Value. It can be seen that within seven seconds the ISR vehicle arrives at a steady-state value for $target_2^t$ s Value. The steady state value error is 0.125 in which it is rounded to the nearest integer, resulting in the calculated Value of 8.

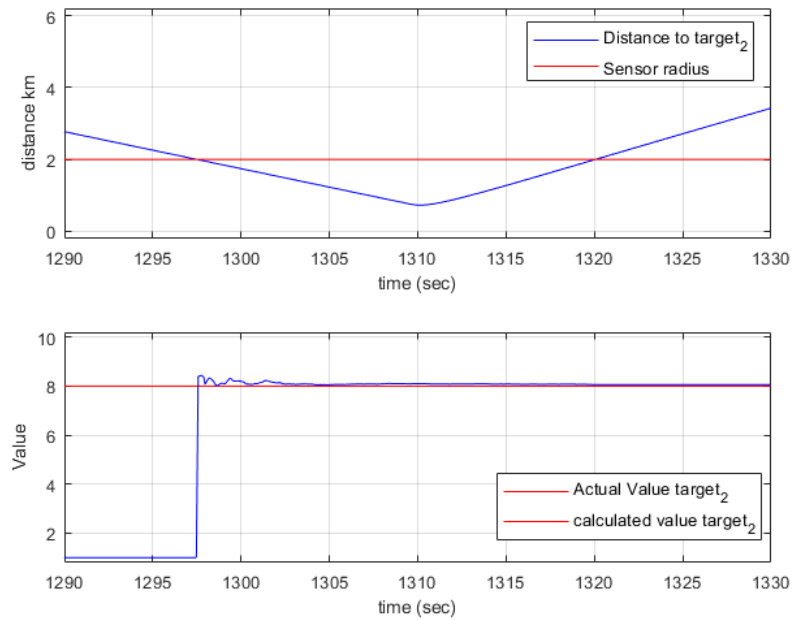


Figure 5.6: Value calculation during simulation

5.1.4 JIC Two Targets

This section looks at simulated flights with two Targets, where the starting location is $[0,0,0]$ and the ending location is $[25, 25, 25]$. The results of JIC are evaluated at γ equals $[0.66, 0.71, 0.77, 0.82, 0.87]$. The two Targets are placed at random in a restricted air space limited to 5 km, 10 km, 20 km, 50 km, 100 km, and 200 km per axis. For example, in the 5 km case the Targets can take on locations on the x-axis from 0 km to 5 km, the y-axis from 0 km to 5 km, and the z-axis from 0 km to 5 km.

Three hundred simulations per γ and restricted air space are used for this analysis. One hundred using Control with Dubins Path, one hundred using Control with Optimized order Path, and one hundred using Control with Closest Target Path. Results from all three methods are combined to allow for analysis of the dynamics of the Joint Inference Control algorithm. The amount of savings shown, is the amount of distance traveled that is reduced from using JIC with compared to using the Control algorithm alone.

Two Targets: axis are bound from 0 km to 5 km

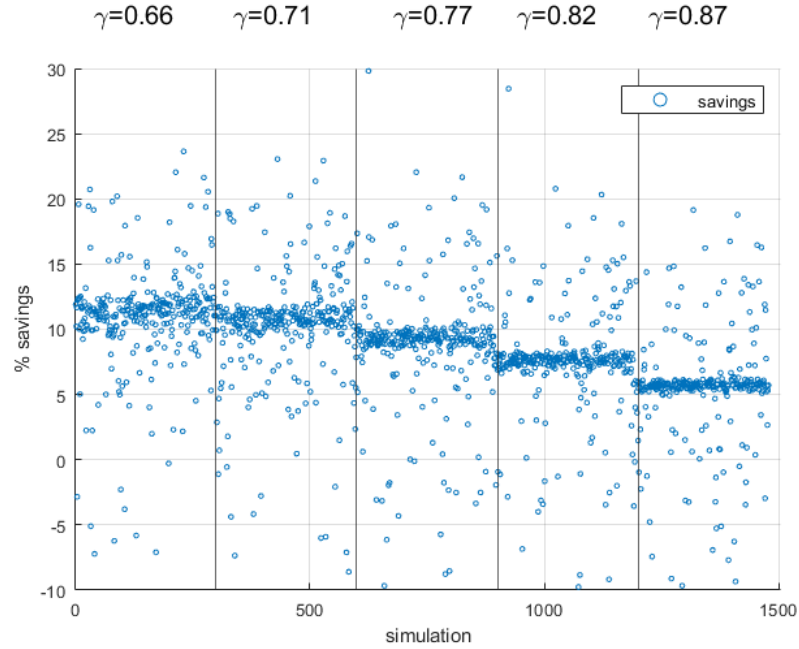


Figure 5.7: Simulated flight trajectory with two Targets when all axis are bound from 0 km to 5 km

Figure 5.7 shows the simulated flights of Two targets where γ is varied as [0.66, 0.71, 0.77, 0.82, 0.87]. The targets are placed at random in the bound space of x-axis equals a range of [0 5] km, y-axis equals a range of [0 5] km, and the z-axis equals a range of [0 5] km. Three hundred simulations are ran for each value of γ . Statistical analysis of this simulation can be seen in table 5.1 below.

Table 5.1: Statistical analysis of two Targets where all axis are bound from 0 km to 5 km

	$\gamma = 0.66$	$\gamma = 0.71$	$\gamma = 0.77$	$\gamma = 0.82$	$\gamma = 0.87$
Peak savings (%)	23.64	23.05	29.80	28.45	26.85
mean savings (%)	10.93	10.29	8.90	7.60	5.60
standard deviation	4.11	4.38	4.56	4.14	4.32
standard error	0.24	0.25	0.26	0.24	0.25

Two Targets: axis are bound from 0 km to 10 km

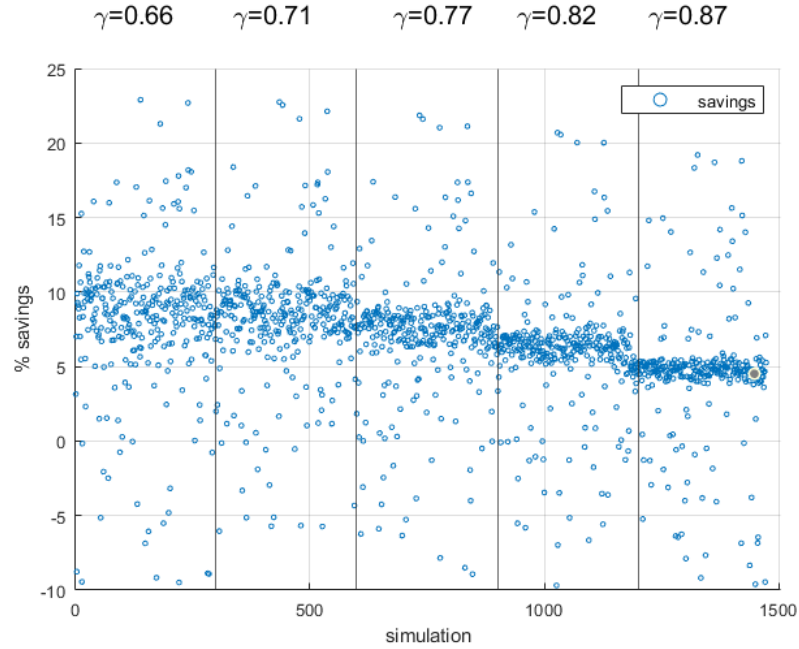


Figure 5.8: Simulated flight trajectory with two Targets when all axis are bound from 0 km to 10 km

Figure 5.8 shows the simulated flights of Two targets where γ is varied as [0.66, 0.71, 0.77, 0.82, 0.87]. The targets are placed at random in the bound space of x-axis equals a range of [0 10] km, y-axis equals a range of [0 10] km, and the z-axis equals a range of [0 10] km. Three hundred simulations are ran for each value of γ . Statistical analysis of this simulation can be seen in table 5.2 below.

Table 5.2: Statistical analysis of two Targets where all axis are bound from 0 km to 10 km

	$\gamma = 0.66$	$\gamma = 0.71$	$\gamma = 0.77$	$\gamma = 0.82$	$\gamma = 0.87$
Peak savings (%)	22.90	22.75	21.85	20.69	19.20
mean savings (%)	8.14	8.25	7.11	6.00	4.52
standard deviation	4.79	4.02	4.27	3.83	4.13
standard error	0.29	0.24	0.25	0.23	0.24

Two Targets: axis are bound from 0 km to 20 km

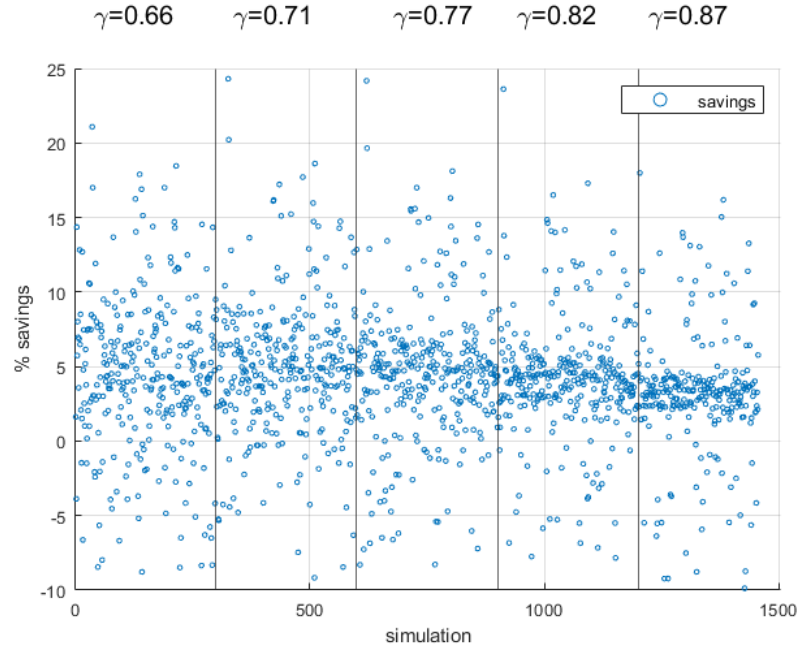


Figure 5.9: Simulated flight trajectory with two Targets when all axis are bound from 0 km to 20 km

Figure 5.9 shows the simulated flights of Two targets where γ is varied as [0.66, 0.71, 0.77, 0.82, 0.87]. The targets are placed at random in the bound space of x-axis equals a range of [0 20] km, y-axis equals a range of [0 20] km, and the z-axis equals a range of [0 20] km. Three hundred simulations are ran for each value of γ . Statistical analysis of this simulation can be seen in table 5.3 below.

Table 5.3: Statistical analysis of two Targets where all axis are bound from 0 km to 20 km

	$\gamma = 0.66$	$\gamma = 0.71$	$\gamma = 0.77$	$\gamma = 0.82$	$\gamma = 0.87$
Peak savings (%)	21.09	24.30	24.7	23.61	22.70
mean savings (%)	4.41	4.99	4.60	4.21	3.40
standard deviation	5.09	4.65	4.75	4.05	4.12
standard error	0.30	0.27	0.28	0.24	0.24

Two Targets: axis are bound from 0 km to 50 km

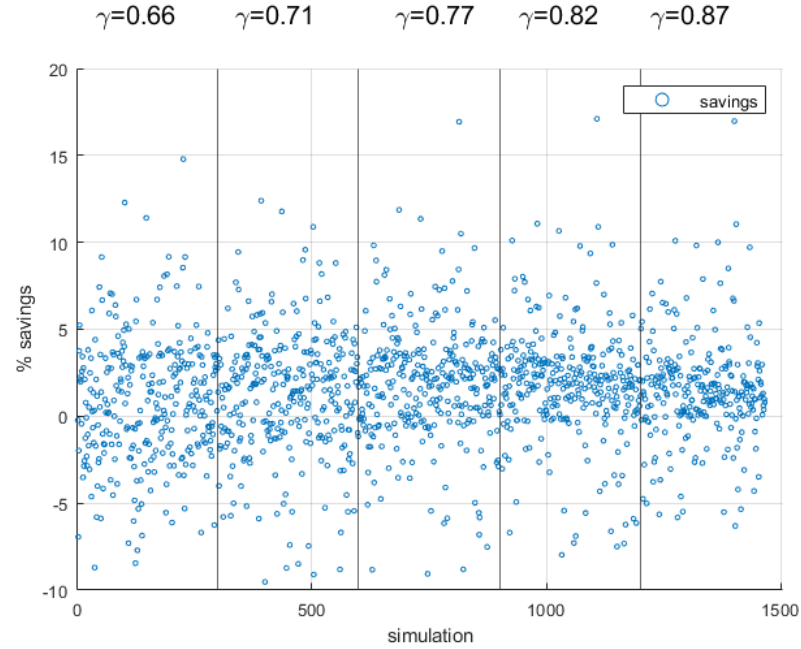


Figure 5.10: Simulated flight trajectory with two Targets when all axis are bound from 0 km to 50 km

Figure 5.10 shows the simulated flights of Two targets where γ is varied as [0.66, 0.71, 0.77, 0.82, 0.87]. The targets are placed at random in the bound space of x-axis equals a range of [0 50] km, y-axis equals a range of [0 50] km, and the z-axis equals a range of [0 50] km. Three hundred simulations are ran for each value of γ . Statistical analysis of this simulation can be seen in table 5.4 below.

Table 5.4: Statistical analysis of two Targets where all axis are bound from 0 km to 50 km

	$\gamma = 0.66$	$\gamma = 0.71$	$\gamma = 0.77$	$\gamma = 0.82$	$\gamma = 0.87$
Peak savings (%)	14.78	12.40	16.93	17.10	16.97
mean savings (%)	0.80	1.29	1.80	1.88	1.50
standard deviation	3.56	3.51	3.33	3.13	2.89
standard error	0.21	0.20	0.19	0.18	0.17

Two Targets: axis are bound from 0 km to 100 km

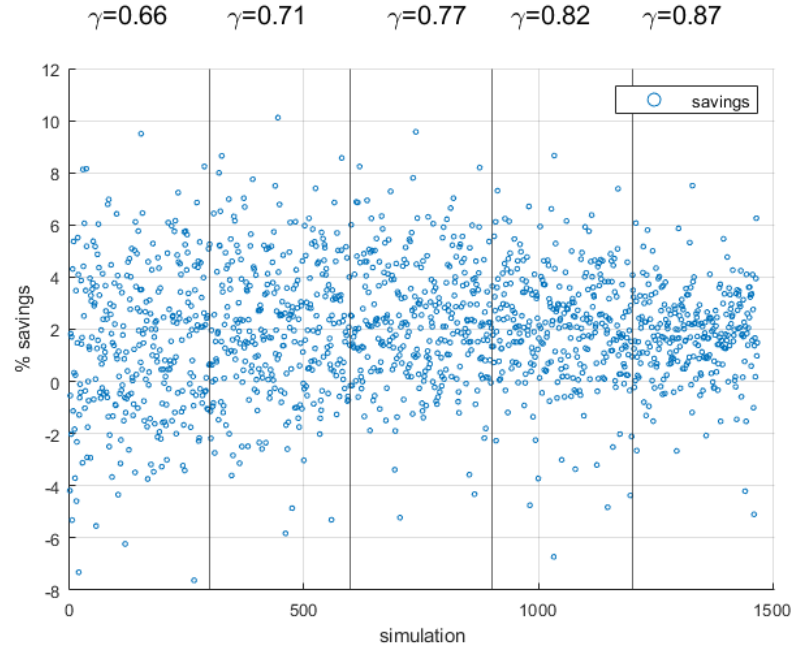


Figure 5.11: Simulated flight trajectory with two Targets when all axis are bound from 0 km to 100 km

Figure 5.11 shows the simulated flights of Two targets where γ is varied as [0.66, 0.71, 0.77, 0.82, 0.87]. The targets are placed at random in the bound space of x-axis equals a range of [0 100] km, y-axis equals a range of [0 100] km, and the z-axis equals a range of [0 100] km. Three hundred simulations are ran for each value of γ . Statistical analysis of this simulation can be seen in table 5.5 below.

Table 5.5: Statistical analysis of two Targets where all axis are bound from 0 km to 100 km

	$\gamma = 0.66$	$\gamma = 0.71$	$\gamma = 0.77$	$\gamma = 0.82$	$\gamma = 0.87$
Peak savings (%)	9.50	10.12	9.57	8.66	7.50
mean savings (%)	1.40	2.09	2.36	2.17	1.76
standard deviation	2.90	2.60	2.27	2.11	1.68
standard error	0.17	0.15	0.13	0.12	0.09

Two Targets: axis are bound from 0 km to 200 km

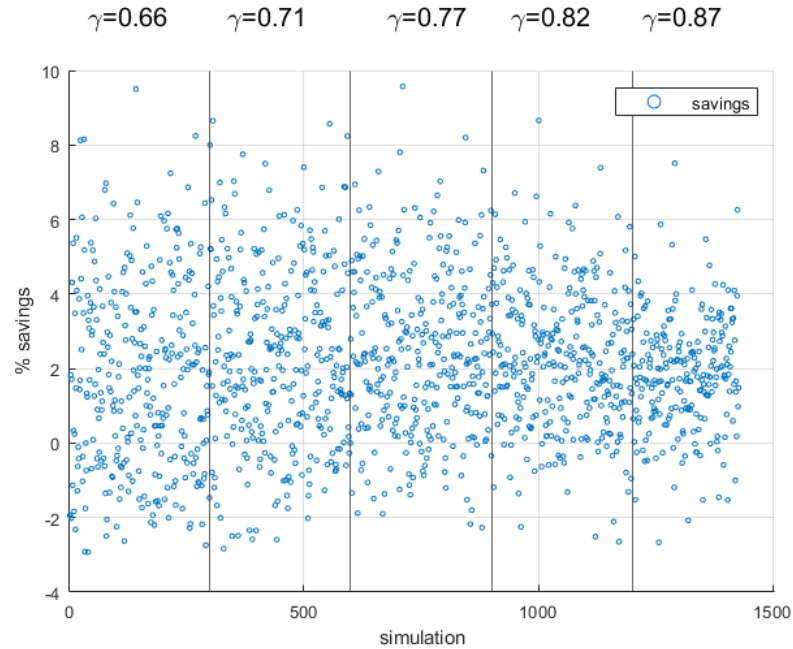


Figure 5.12: Simulated flight trajectory with two Targets when all axis are bound from 0 km to 200 km

Figure 5.12 shows the simulated flights of Two targets where γ is varied as [0.66, 0.71, 0.77, 0.82, 0.87]. The targets are placed at random in the bound space of x-axis equals a range of [0 200] km, y-axis equals a range of [0 200] km, and the z-axis equals a range of [0 200] km. Three hundred simulations are ran for each value of γ . Statistical analysis of this simulation can be seen in table 5.6 below.

Table 5.6: Statistical analysis of two Targets where all axis are bound from 0 km to 200 km

	$\gamma = 0.66$	$\gamma = 0.71$	$\gamma = 0.77$	$\gamma = 0.82$	$\gamma = 0.87$
Peak savings (%)	9.50	8.65	9.57	8.66	7.50
mean savings (%)	1.81	2.19	2.45	2.32	1.83
standard deviation	2.53	2.42	2.15	1.88	1.55
standard error	0.15	0.14	0.13	0.11	0.09

Two Targets: mean comparison

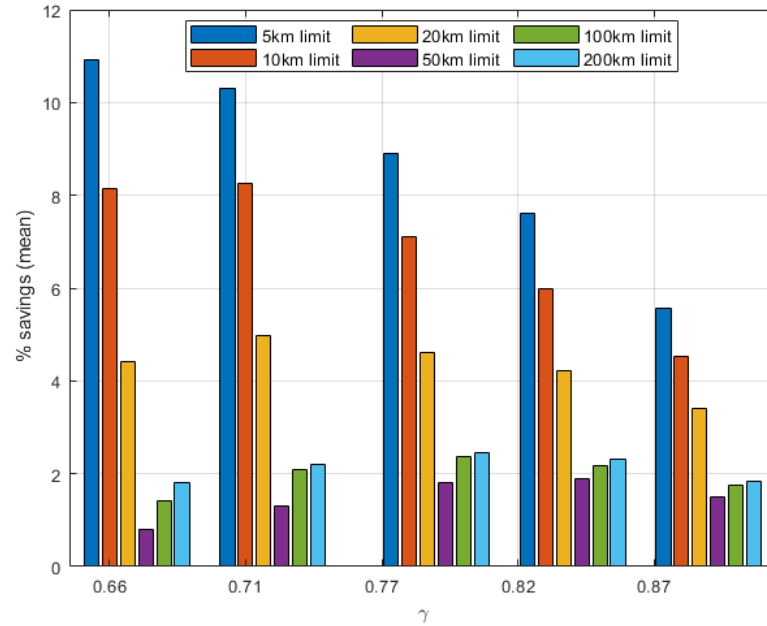


Figure 5.13: Mean comparison of two Target simulations

Figure 5.13 shows the simulated flights of Two targets where, γ is varied as [0.66, 0.71, 0.77, 0.82, 0.87]. Statistical analysis of figure 5.13 can be seen in table 5.7.

Table 5.7: Statistical analysis of two Targets, mean savings

Flight Simulation	Optimal γ	Optimal mean savings (%)
0 km - 5 km	0.66	10.93
0 km - 10 km	0.71	8.25
0 km - 20 km	0.71	4.99
0 km - 50 km	0.82	1.88
0 km - 100 km	0.77	2.36
0 km - 200 km	0.77	2.45

Two Targets: peak savings comparison

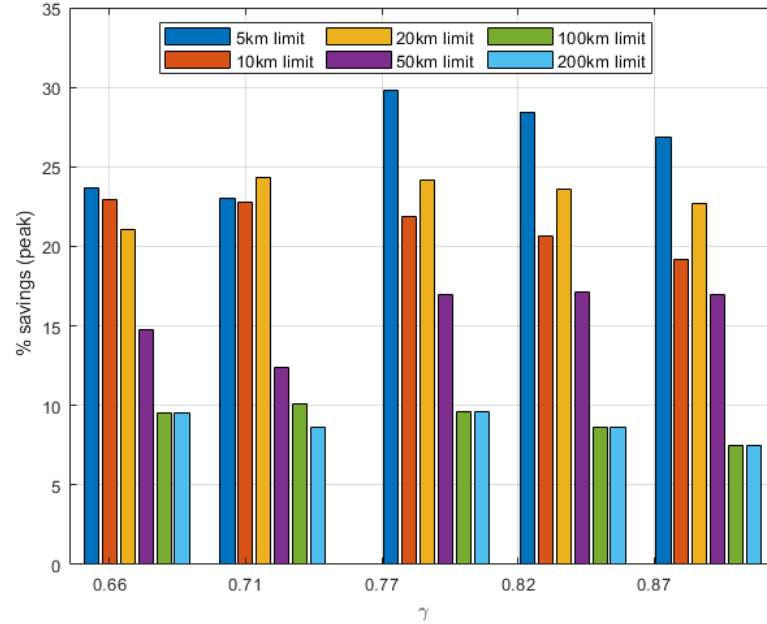


Figure 5.14: Peak savings comparison of two Target simulations

Figure 5.14 shows the simulated flights of Two targets where, γ is varied as [0.66, 0.71, 0.77, 0.82, 0.87]. Statistical analysis of figure 5.14 can be seen in table 5.8.

Table 5.8: Statistical analysis of two Targets, peak savings

Flight Simulation	Optimal γ	Optimal peak savings (%)
0 km - 5 km	0.77	29.80
0 km - 10 km	0.66	22.90
0 km - 20 km	0.77	24.70
0 km - 50 km	0.82	17.10
0 km - 100 km	0.71	10.12
0 km - 200 km	0.77	9.57

5.1.5 JIC Three Targets

This section looks at simulated flights with three Targets, where the starting location is $[0,0,0]$ and the ending location is $[25,25,25]$. The results of JIC are evaluated at γ equals $[0.66, 0.71, 0.77, 0.82, 0.87]$. The three Targets are placed at random in a restricted air space limited to 5 km, 10 km, 20 km, 50 km, 100 km, and 200 km per axis. For example, in the 5km case the Targets can take on locations on the x-axis from 0 km to 5 km, the y-axis from 0 km to 5 km, and the z-axis from 0 km to 5km.

Three hundred simulations per γ and restricted air space are used for this analysis. One hundred using Control with Dubins Path, one hundred using Control with Optimized order Path, and one hundred using Control with Closest Target Path. Results from all three methods are combined to allow for analysis of the dynamics of the Joint Inference Control algorithm. The amount of savings shown, is the amount of distance traveled that is reduced from using JIC with compared to using the Control algorithm alone.

Three Targets: axis are bound from 0 km to 5 km

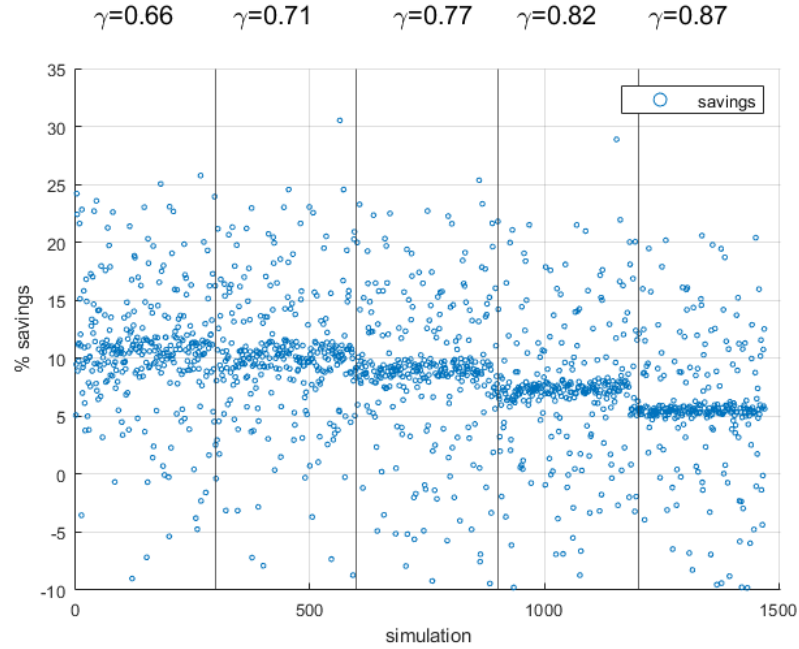


Figure 5.15: Simulated flight trajectory with three Targets when all axis are bound from 0 km to 5 km

Figure 5.15 shows the simulated flights of three targets where γ is varied as [0.66, 0.71, 0.77, 0.82, 0.87]. The targets are placed at random in the bound space of x-axis equals a range of [0 5] km, y-axis equals a range of [0 5] km, and the z-axis equals a range of [0 5] km. Three hundred simulations are ran for each value of γ . Statistical analysis of this simulation can be seen in table 5.9 below.

Table 5.9: Statistical analysis of three Targets where all axis are bound from 0 km to 5 km

	$\gamma = 0.66$	$\gamma = 0.71$	$\gamma = 0.77$	$\gamma = 0.82$	$\gamma = 0.87$
Peak savings (%)	25.76	30.51	25.35	28.88	20.56
mean savings (%)	10.86	10.39	8.96	7.47	6.00
standard deviation	5.24	5.24	5.85	5.29	5.32
standard error	0.31	0.30	0.34	0.31	0.31

Three Targets: axis are bound from 0 km to 10 km

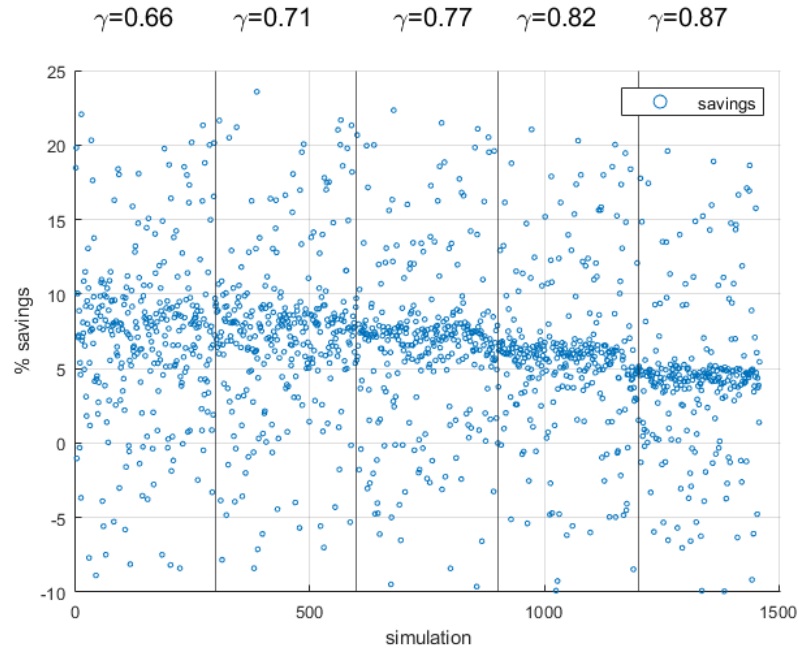


Figure 5.16: Simulated flight trajectory with three Targets when all axis are bound from 0 km to 10 km

Figure 5.16 shows the simulated flights of three targets where γ is varied as [0.66, 0.71, 0.77, 0.82, 0.87]. The targets are placed at random in the bound space of x-axis equals a range of [0 10] km, y-axis equals a range of [0 10] km, and the z-axis equals a range of [0 10] km. Three hundred simulations are ran for each value of γ . Statistical analysis of this simulation can be seen in table 5.10 below.

Table 5.10: Statistical analysis of three Targets where all axis are bound from 0 km to 10 km

	$\gamma = 0.66$	$\gamma = 0.71$	$\gamma = 0.77$	$\gamma = 0.82$	$\gamma = 0.87$
Peak savings (%)	22.06	23.57	22.33	21.04	19.59
mean savings (%)	7.10	7.43	6.81	6.12	4.34
standard deviation	5.53	5.30	5.18	4.93	5.04
standard error	0.322	0.31	0.31	0.29	0.30

Three Targets: axis are bound from 0 km to 20 km

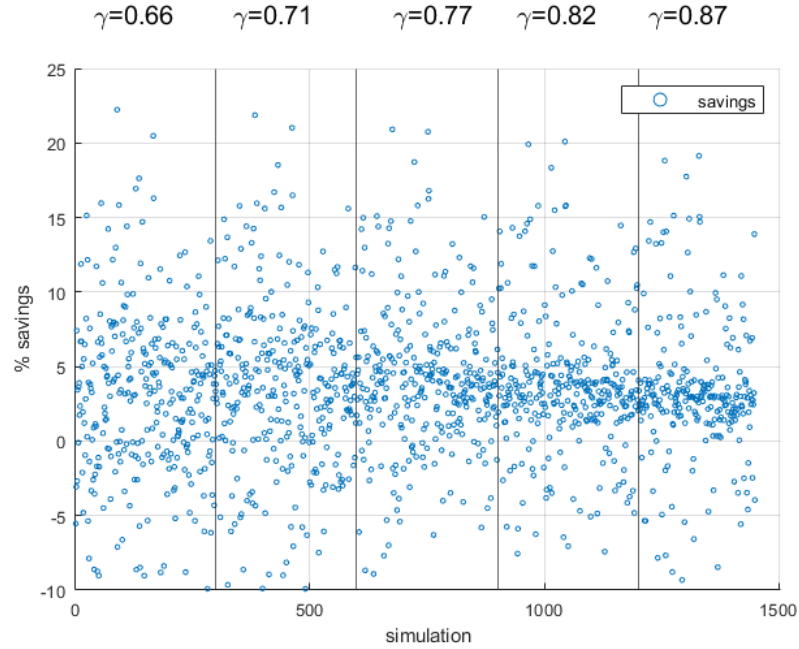


Figure 5.17: Simulated flight trajectory with three Targets when all axis are bound from 0 km to 20 km

Figure 5.17 shows the simulated flights of three targets where γ is varied as [0.66, 0.71, 0.77, 0.82, 0.87]. The targets are placed at random in the bound space of x-axis equals a range of [0 20] km, y-axis equals a range of [0 20] km, and the z-axis equals a range of [0 20] km. Three hundred simulations are ran for each value of γ . Statistical analysis of this simulation can be seen in table 5.11 below.

Table 5.11: Statistical analysis of three Targets where all axis are bound from 0 km to 20 km

	$\gamma = 0.66$	$\gamma = 0.71$	$\gamma = 0.77$	$\gamma = 0.82$	$\gamma = 0.87$
Peak savings (%)	22.24	21.87	20.92	20.09	19.13
mean savings (%)	3.12	3.71	4.16	3.74	3.03
standard deviation	5.48	5.43	4.71	4.47	4.48
standard error	0.32	0.32	0.28	0.26	0.27

Three Targets: axis are bound from 0 km to 50 km

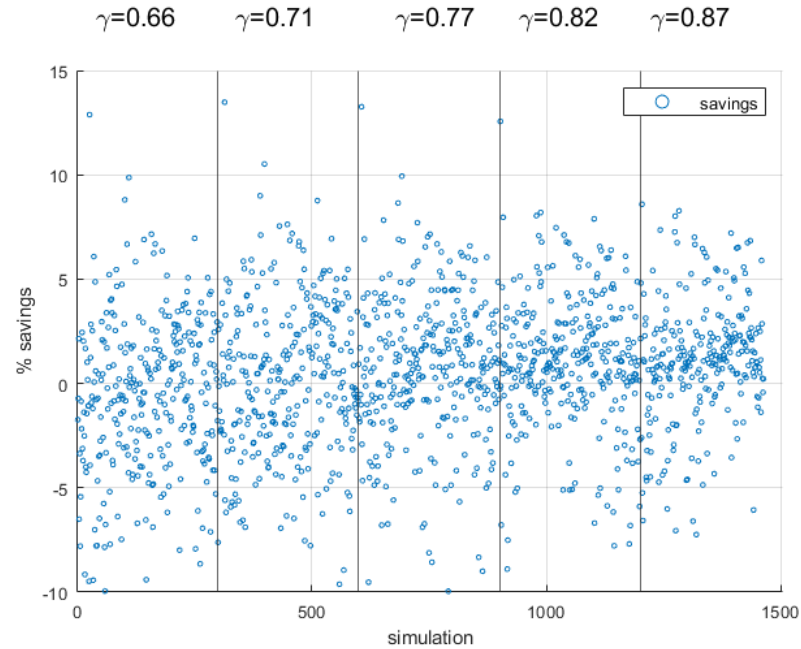


Figure 5.18: Simulated flight trajectory with three Targets when all axis are bound from 0 km to 50 km

Figure 5.18 shows the simulated flights of three targets where γ is varied as [0.66, 0.71, 0.77, 0.82, 0.87]. The targets are placed at random in the bound space of x-axis equals a range of [0 50] km, y-axis equals a range of [0 50] km, and the z-axis equals a range of [0 50] km. Three hundred simulations are ran for each value of γ . Statistical analysis of this simulation can be seen in table 5.12 below.

Table 5.12: Statistical analysis of three Targets where all axis are bound from 0 km to 50 km

	$\gamma = 0.66$	$\gamma = 0.71$	$\gamma = 0.77$	$\gamma = 0.82$	$\gamma = 0.87$
Peak savings (%)	12.88	13.48	13.26	12.56	8.59
mean savings (%)	0.58	0.21	0.72	1.12	0.97
standard deviation	3.70	3.70	3.41	3.04	2.93
standard error	0.22	0.22	0.20	0.18	0.17

Three Targets: axis are bound from 0 km to 100 km

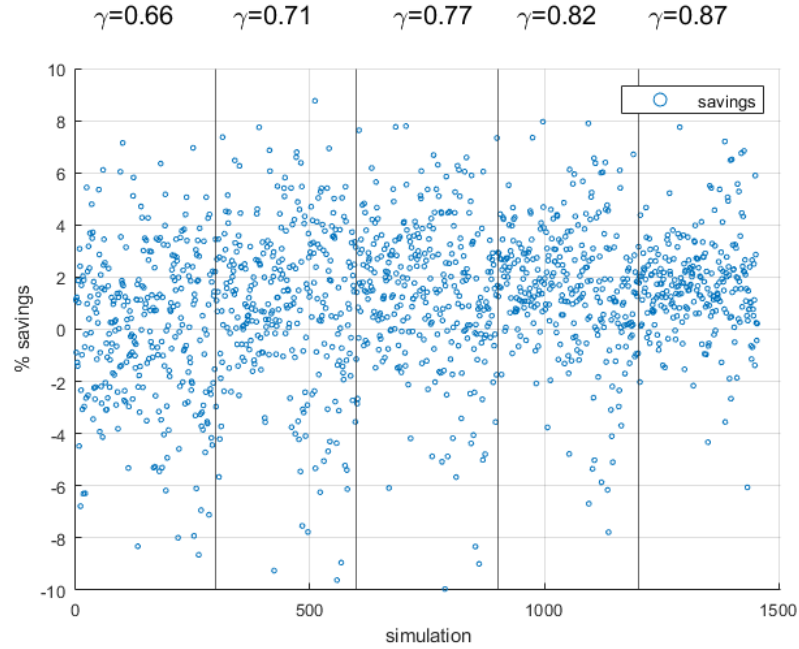


Figure 5.19: Simulated flight trajectory with three Targets when all axis are bound from 0 km to 100 km

Figure 5.19 shows the simulated flights of three targets where γ is varied as [0.66, 0.71, 0.77, 0.82, 0.87]. The targets are placed at random in the bound space of x-axis equals a range of [0 100] km, y-axis equals a range of [0 100] km, and the z-axis equals a range of [0 100] km. Three hundred simulations are ran for each value of γ . Statistical analysis of this simulation can be seen in table 5.13 below.

Table 5.13: Statistical analysis of three Targets where all axis are bound from 0 km to 100 km

	$\gamma = 0.66$	$\gamma = 0.71$	$\gamma = 0.77$	$\gamma = 0.82$	$\gamma = 0.87$
Peak savings (%)	7.15	8.76	7.79	7.96	7.45
mean savings (%)	0.03	1.02	1.45	1.62	1.62
standard deviation	2.93	3.06	2.74	2.42	1.83
standard error	0.17	0.18	0.16	0.14	0.10

Three Targets: axis are bound from 0 km to 200 km

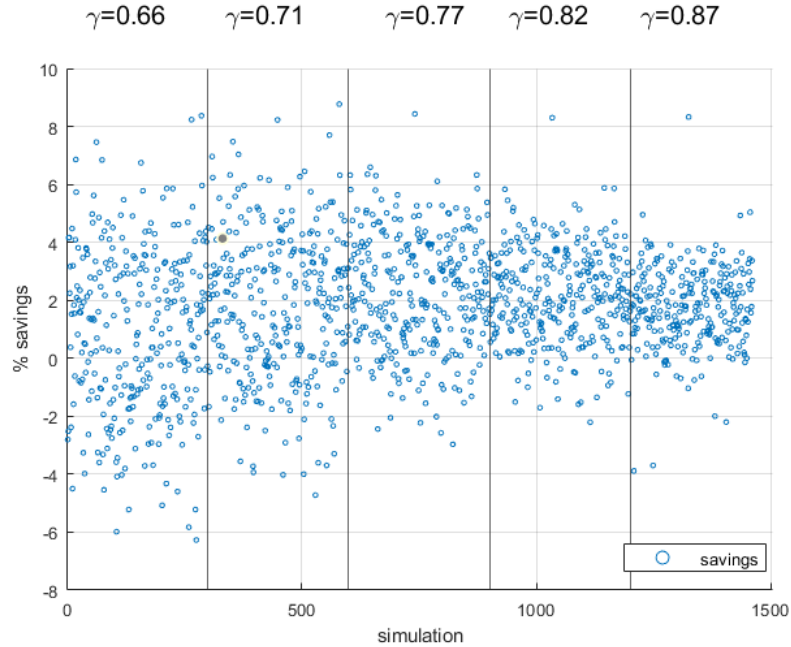


Figure 5.20: Simulated flight trajectory with three Targets when all axis are bound from 0 km to 200 km

Figure 5.20 shows the simulated flights of three targets where γ is varied as [0.66, 0.71, 0.77, 0.82, 0.87]. The targets are placed at random in the bound space of x-axis equals a range of [0 200] km, y-axis equals a range of [0 200] km, and the z-axis equals a range of [0 200] km. Three hundred simulations are ran for each value of γ . Statistical analysis of this simulation can be seen in table 5.14 below.

Table 5.14: Statistical analysis of three Targets where all axis are bound from 0 km to 200 km

	$\gamma = 0.66$	$\gamma = 0.71$	$\gamma = 0.77$	$\gamma = 0.82$	$\gamma = 0.87$
Peak savings (%)	8.37	8.77	8.43	8.30	8.33
mean savings (%)	0.855	1.83	2.26	2.17	1.76
standard deviation	2.84	2.51	1.99	1.62	1.34
standard error	0.16	0.15	0.12	0.09	0.08

0.

Three Targets: mean comparison

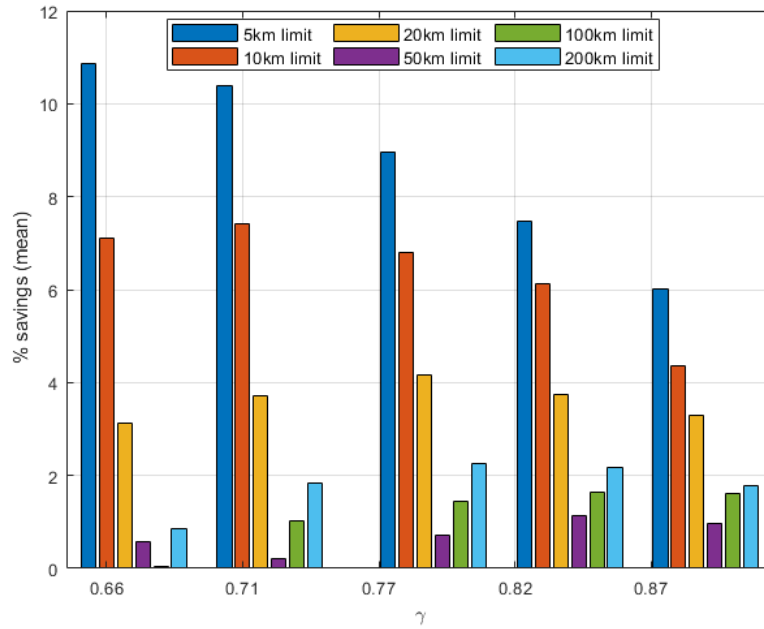


Figure 5.21: Mean comparison of three Target simulations

Figure 5.21 shows the simulated flights of three targets, where γ is varied as [0.66, 0.71, 0.77, 0.82, 0.87]. Statistical analysis of figure 5.21 can be seen in table 5.15.

Table 5.15: Statistical analysis of three Targets, mean savings

Flight Simulation	Optimal γ	Optimal mean savings (%)
0 km - 5 km	0.66	10.86
0 km - 10 km	0.71	7.43
0 km - 20 km	0.77	4.16
0 km - 50 km	0.82	1.12
0 km - 100 km	0.82	1.62
0 km - 200 km	0.77	2.26

Three Targets: peak savings comparison

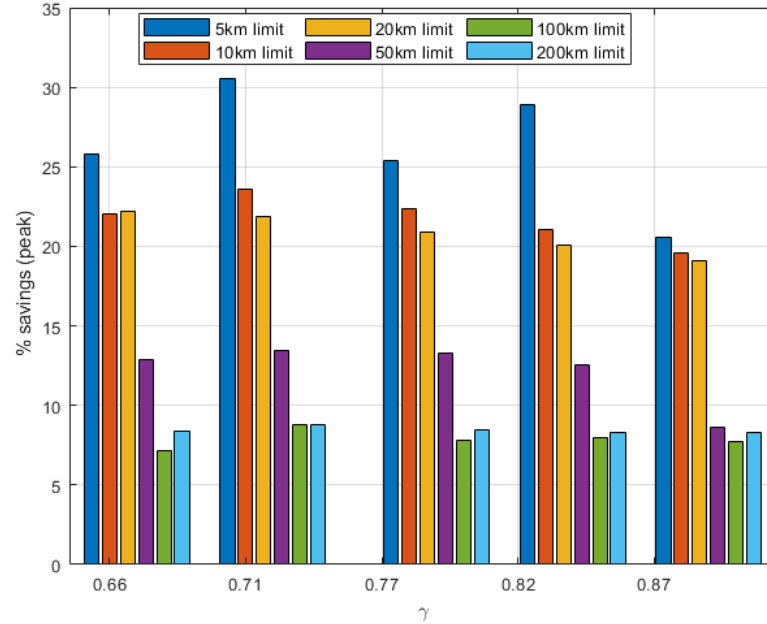


Figure 5.22: Peak savings comparison of three Target simulations

Figure 5.22 shows the simulated flights of three Targets, where γ is varied as [0.66, 0.71, 0.77, 0.82, 0.87]. Statistical analysis of figure 5.22 can be seen in table 5.16.

Table 5.16: Statistical analysis of three Targets, peak savings

Flight Simulation	Optimal γ	Optimal peak savings (%)
0 km - 5 km	0.71	30.51
0 km - 10 km	0.71	23.57
0 km - 20 km	0.66	22.24
0 km - 50 km	0.71	13.48
0 km - 100 km	0.71	8.76
0 km - 200 km	0.71	8.77

5.1.6 JIC Four Targets

This section looks at simulated flights with four Targets, where the starting location is $[0,0,0]$ and the ending location is $[25,25,25]$. The results of JIC are evaluated at γ equals $[0.71, 0.77, 0.82, 0.87]$. The four Targets are placed at random in a restricted air space limited to 5 km, 10 km, 20 km, 50 km, 100 km, and 200 km per axis. For example, in the 5 km case the Targets can take on locations on the x-axis from 0 km to 5 km, the y-axis from 0 km to 5 km, and the z-axis from 0 km to 5 km.

Three hundred simulations per γ and restricted air space are used for this analysis. One hundred using Control with Dubins Path, one hundred using Control with Optimized order Path, and one hundred using Control with Closest Target Path. Results from all three methods are combined to allow for analysis of the dynamics of the Joint Inference Control algorithm. The amount of savings shown, is the amount of distance traveled that is reduced from using JIC with compared to using the Control algorithm alone.

Four Targets: axis are bound from 0 km to 5 km

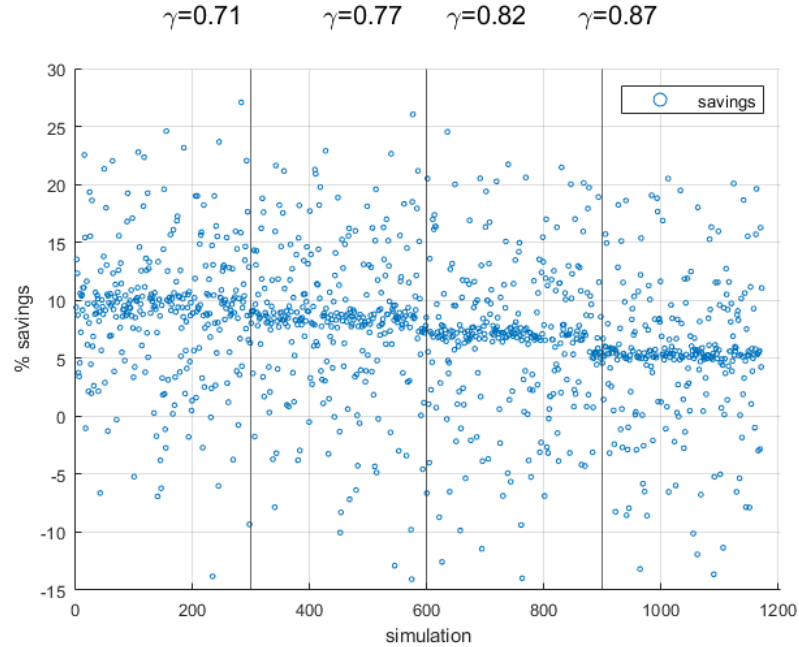


Figure 5.23: Simulated flight trajectory with four Targets when all axis are bound from 0 km to 5 km

Figure 5.23 shows the simulated flights of four targets where γ is varied as [0.71, 0.77, 0.82, 0.87]. The targets are placed at random in the bound space of x-axis equals a range of [0 5] km, y-axis equals a range of [0 5] km, and the z-axis equals a range of [0 5] km. Three hundred simulations are ran for each value of γ . Statistical analysis of this simulation can be seen in table 5.17 below.

Table 5.17: Statistical analysis of four Targets where all axis are bound from 0 km to 5 km

	$\gamma = 0.71$	$\gamma = 0.77$	$\gamma = 0.82$	$\gamma = 0.87$
Peak savings (%)	27.07	26.31	24.56	20.51
mean savings (%)	9.23	8.36	6.80	5.32
standard deviation	5.68	6.09	6.21	5.83
standard error	0.33	0.36	0.36	0.34

Four Targets: axis are bound from 0 km to 10 km

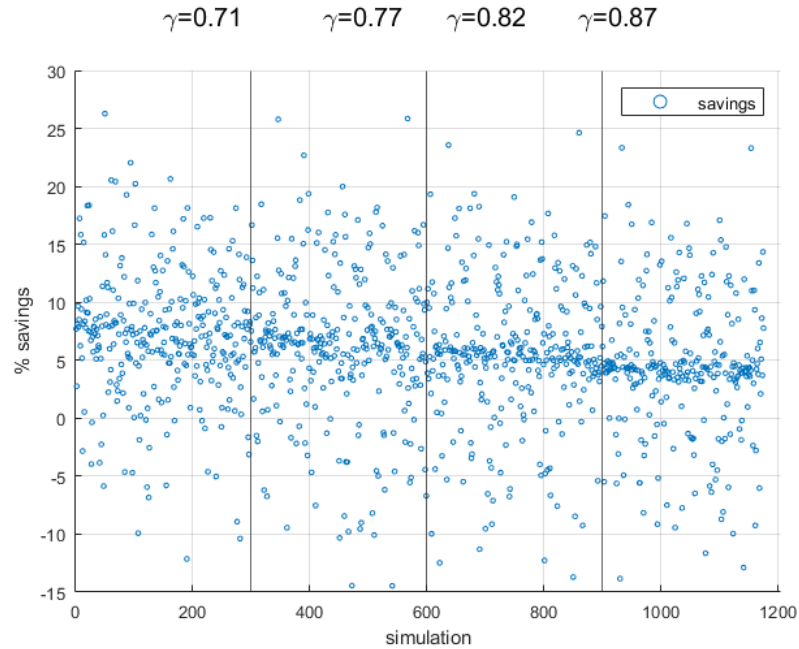


Figure 5.24: Simulated flight trajectory with four Targets when all axis are bound from 0 km to 10 km

Figure 5.24 shows the simulated flights of four targets where γ is varied as [0.71, 0.77, 0.82, 0.87]. The targets are placed at random in the bound space of x-axis equals a range of [0 10] km, y-axis equals a range of [0 10] km, and the z-axis equals a range of [0 10] km. Three hundred simulations are ran for each value of γ . Statistical analysis of this simulation can be seen in table 5.18 below.

Table 5.18: Statistical analysis of four Targets where all axis are bound from 0 km to 10 km

	$\gamma = 0.71$	$\gamma = 0.77$	$\gamma = 0.82$	$\gamma = 0.87$
Peak savings (%)	26.27	25.84	24.62	23.33
mean savings (%)	7.18	6.36	5.60	4.41
standard deviation	5.62	6.33	6.21	5.81
standard error	0.33	0.37	0.36	0.34

Four Targets: axis are bound from 0 km to 20 km

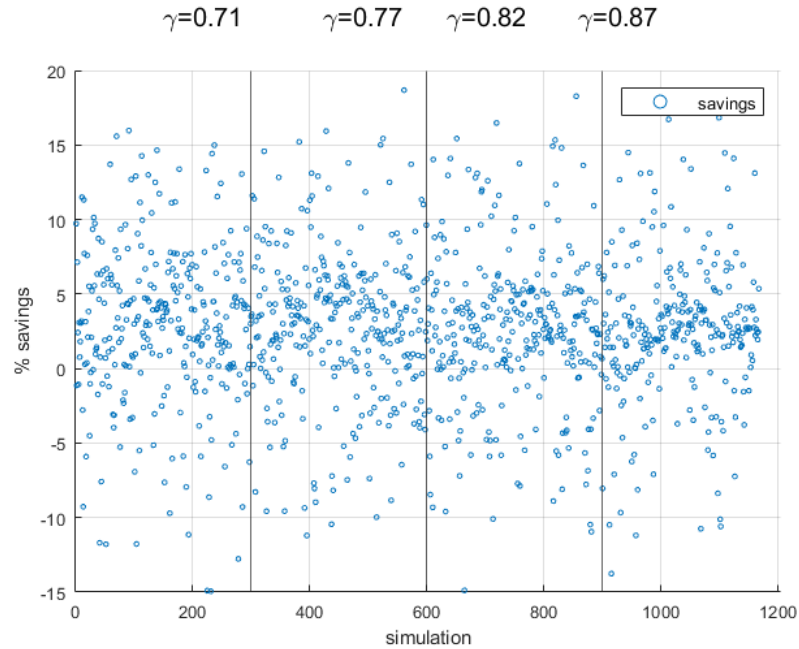


Figure 5.25: Simulated flight trajectory with four Targets when all axis are bound from 0 km to 20 km

Figure 5.25 shows the simulated flights of four targets where γ is varied as [0.71, 0.77, 0.82, 0.87]. The targets are placed at random in the bound space of x-axis equals a range of [0 20] km, y-axis equals a range of [0 20] km, and the z-axis equals a range of [0 20] km. Three hundred simulations are ran for each value of γ . Statistical analysis of this simulation can be seen in table 5.19 below.

Table 5.19: Statistical analysis of four Targets where all axis are bound from 0 km to 20 km

	$\gamma = 0.71$	$\gamma = 0.77$	$\gamma = 0.82$	$\gamma = 0.87$
Peak savings (%)	15.96	18.68	18.27	17.43
mean savings (%)	2.91	2.93	2.74	2.40
standard deviation	5.39	5.08	5.13	4.88
standard error	0.32	0.30	0.30	0.29

Four Targets: axis are bound from 0 km to 50 km

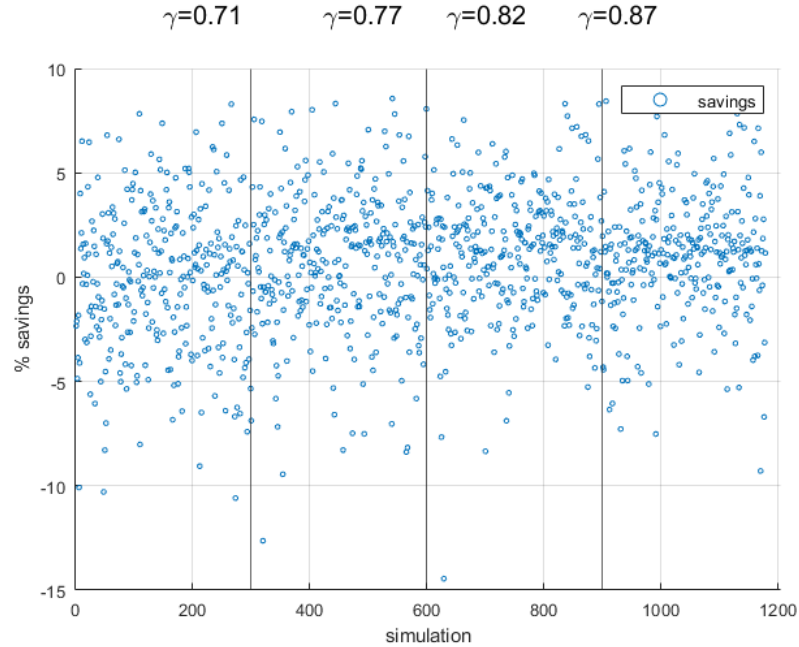


Figure 5.26: Simulated flight trajectory with four Targets when all axis are bound from 0 km to 50 km

Figure 5.26 shows the simulated flights of four targets where γ is varied as [0.71, 0.77, 0.82, 0.87]. The targets are placed at random in the bound space of x-axis equals a range of [0 50] km, y-axis equals a range of [0 50] km, and the z-axis equals a range of [0 50] km. Three hundred simulations are ran for each value of γ . Statistical analysis of this simulation can be seen in table 5.20 below.

Table 5.20: Statistical analysis of four Targets where all axis are bound from 0 km to 50 km

	$\gamma = 0.71$	$\gamma = 0.77$	$\gamma = 0.82$	$\gamma = 0.87$
Peak savings (%)	8.29	8.56	8.31	8.68
mean savings (%)	0.13	0.63	1.15	0.99
standard deviation	3.48	3.32	2.87	2.90
standard error	0.20	0.19	0.17	0.17

Four Targets: axis are bound from 0 km to 100 km

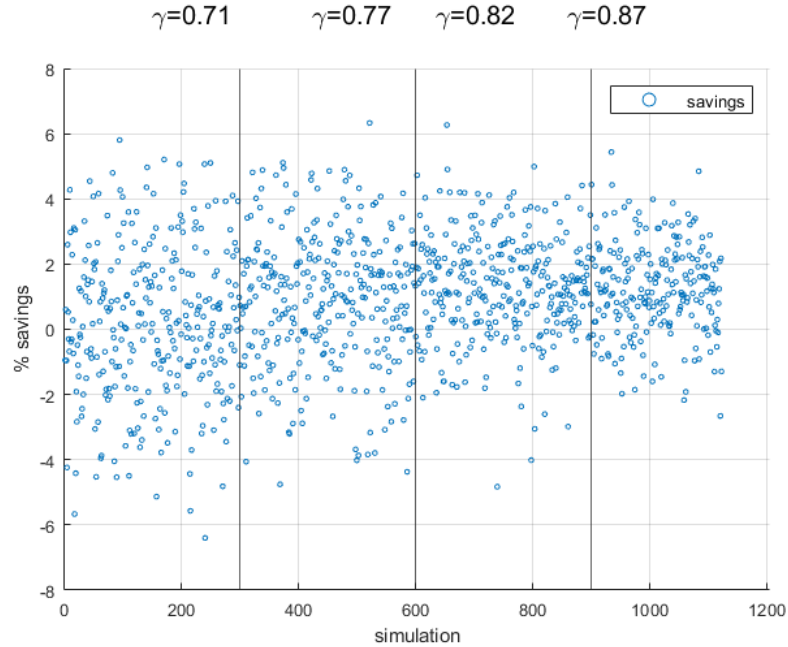


Figure 5.27: Simulated flight trajectory with four Targets when all axis are bound from 0 km to 100 km

Figure 5.27 shows the simulated flights of four targets where γ is varied as [0.71, 0.77, 0.82, 0.87]. The targets are placed at random in the bound space of x-axis equals a range of [0 100] km, y-axis equals a range of [0 100] km, and the z-axis equals a range of [0 100] km. Three hundred simulations are ran for each value of γ . Statistical analysis of this simulation can be seen in table 5.21 below.

Table 5.21: Statistical analysis of four Targets where all axis are bound from 0 km to 100 km

	$\gamma = 0.71$	$\gamma = 0.77$	$\gamma = 0.82$	$\gamma = 0.87$
Peak savings (%)	5.80	6.32	3.26	5.44
mean savings (%)	0.12	0.93	1.19	1.23
standard deviation	2.36	2.04	1.69	1.36
standard error	0.14	0.12	0.10	0.08

Four Targets: axis are bound from 0 km to 200 km

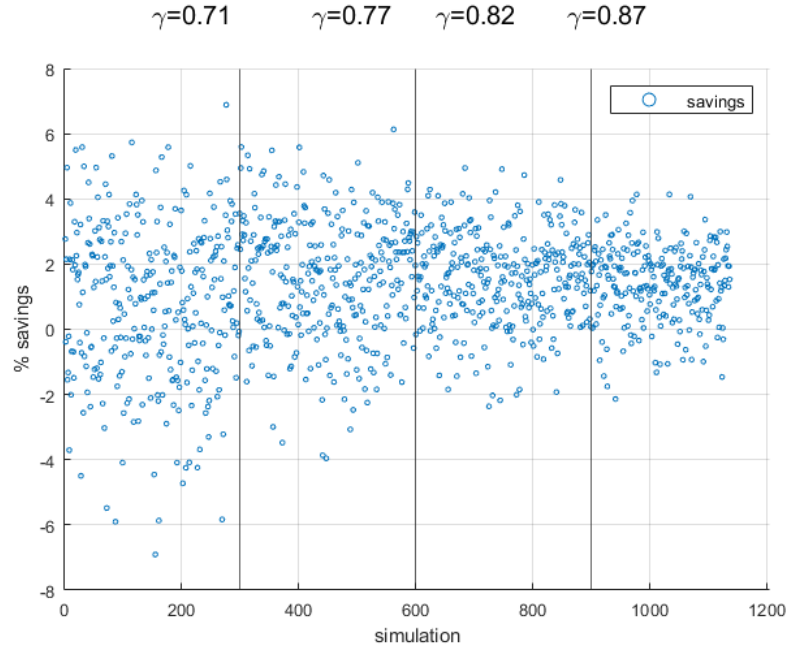


Figure 5.28: Simulated flight trajectory with four Targets when all axis are bound from 0 km to 200 km

Figure 5.28 shows the simulated flights of four targets where γ is varied as [0.71, 0.77, 0.82, 0.87]. The targets are placed at random in the bound space of x-axis equals a range of [0 200] km, y-axis equals a range of [0 200] km, and the z-axis equals a range of [0 200] km. Three hundred simulations are ran for each value of γ . Statistical analysis of this simulation can be seen in table 5.22 below.

Table 5.22: Statistical analysis of four Targets where all axis are bound from 0 km to 200 km

	$\gamma = 0.71$	$\gamma = 0.77$	$\gamma = 0.82$	$\gamma = 0.87$
Peak savings (%)	6.88	6.13	4.95	4.14
mean savings (%)	0.72	1.50	1.59	1.45
standard deviation	2.54	1.88	1.45	1.13
standard error	0.15	0.11	0.08	0.06

Four Targets: mean comparison

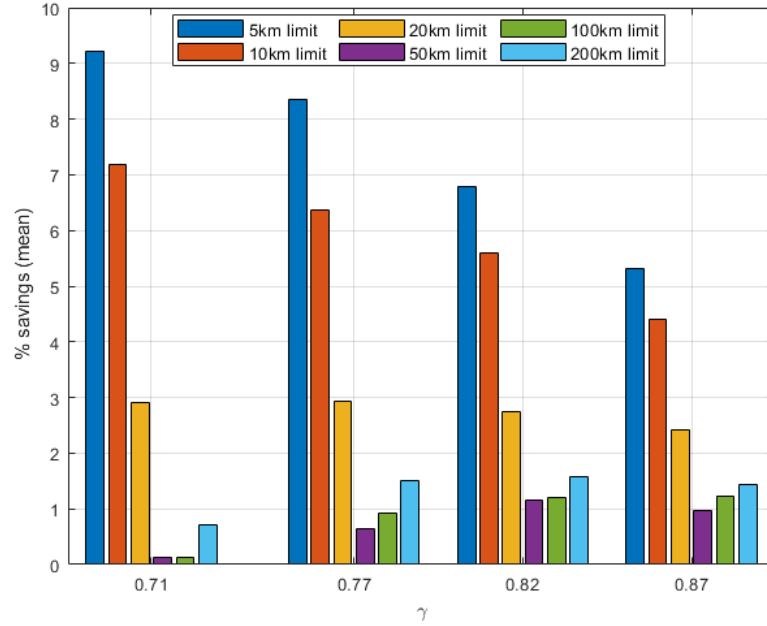


Figure 5.29: Mean comparison of four Target simulations

Figure 5.29 shows the simulated flights of four targets, where γ is varied as [0.71, 0.77, 0.82, 0.87]. Statistical analysis of figure 5.29 can be seen in table 5.23.

Table 5.23: Statistical analysis of four Targets, Optimal mean savings

Flight Simulation	Optimal γ	Optimal mean savings (%)
0 km - 5 km	0.71	9.23
0 km - 10 km	0.71	7.18
0 km - 20 km	0.77	2.93
0 km - 50 km	0.82	1.15
0 km - 100 km	0.82	1.19
0 km - 200 km	0.82	1.59

Four Targets: peak savings comparison

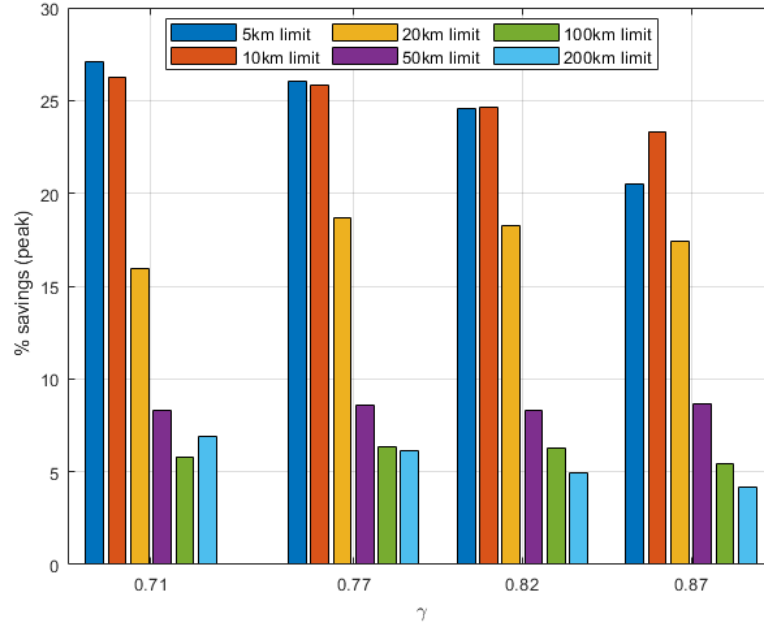


Figure 5.30: Peak savings comparison of four Target simulations

Figure 5.30 shows the simulated flights of four targets, where γ is varied as [0.71, 0.77, 0.82, 0.87]. Statistical analysis of figure 5.30 can be seen in table 5.24.

Table 5.24: Statistical analysis of four Targets, peak savings

Flight Simulation	Optimal γ	Optimal peak savings (%)
0 km - 5 km	0.71	27.07
0 km - 10 km	0.71	26.27
0 km - 20 km	0.77	18.68
0 km - 50 km	0.87	8.68
0 km - 100 km	0.77	6.32
0 km - 200 km	0.71	6.88

CONCLUSION AND FUTURE WORK

With Unmanned Aerial Vehicle (UAV) uses and applications continuing to increase, the need to optimize the resources available to UAVs has become a significant challenge. My research addresses this challenge from the viewpoint of Intelligence Surveillance and Reconnaissance (ISR) while managing the resource of Range available to a fixed-wing UAV. Managing the resource of Range available by reducing the amount of distance needed to travel.

Research has taken on the challenge of resource management by developing several tune-able algorithms to optimize flight trajectory. Thus effectively reducing the required amount of distance needed to travel. Research has allowed for the use of a varying-variance which allows for tuning the Inference algorithm to replicate the Signal to Noise Ratio (SNR) seen in any sensor. Providing an approach to calculating Target Confidence and a method of calculating Target Values, which is tune-able to counter the effects of SNR.

Specifically, research also shows when the number of Targets present increases as well as when the distance between Targets increases, the Joint Inference and Control (JIC) algorithm's effectiveness is reduced. This can be seen as, when there are two Targets in the air space of x-axis equals a range of [0 5] km, the y-axis equals a range of [0 5] km, and the z-axis equals a range of [0 5] km. The resulting mean savings was 10.93% with a peak savings of 29.80%. With the presence of four Targets in the air space of the x-axis equals a range of [0 200] km, the y-axis equals a range of [0 200] km, and the z-axis equals a range of [0 200] km. The resulting mean savings was 1.59% with a peak savings of 6.88%. This

brings several potential directions for future work.

Future work from this research can involve increasing the efficiency of the JIC algorithm as well as increasing the complexity of the problem. One direction this research is heading to solve the reduction of effectiveness is to group one large number of Targets into several smaller clusters of Targets. Where, when the ISR vehicle is outside of a cluster, the JIC algorithm sees the entire cluster as one Target, then once inside the cluster the JIC algorithm only focuses on the Targets present in that specific cluster. This direction is aimed at creating more time in-flight seeing larger amounts of savings.

Future complexity of the problem can also be increased by simulating moving Targets as well as developing additional JIC algorithms that will focus on managing resources additional to the range available to a fixed-wing UAV.

Bibliography

- [1] R. P. Malhotra, M. J. Pribilski, P. A. Toole, and C. Agate, “Decentralized asset management for collaborative sensing,” in *Micro- and Nanotechnology Sensors, Systems, and Applications IX* (T. George, A. K. Dutta, and M. S. Islam, eds.), vol. 10194, pp. 403 – 414, International Society for Optics and Photonics, SPIE, 2017.
- [2] L. E. Dubins, “On curves of minimal length with a constraint on average curvature, and with prescribed initial and terminal positions and tangents,” *American Journal of Mathematics*, vol. 79, no. 3, pp. 497–516, 1957.
- [3] R. W. Beard and T. W. McLain, *Small Unmanned Aircraft Theory and Practice*. Princeton, New Jersey: Princeton University Press, 2012.
- [4] C. Clapham and J. Nicholson, *The Concise Oxford Dictionary of Mathematics (4 ed.)*. Oxford, England: Oxford University PressPrint, 2009.
- [5] A. Wald, *Sequential Analysis*. Hoboken, New Jersey: John Wiley and Sons Inc., 1947.
- [6] D. Siegmund, *Sequential Analysis: Test and Confidence Intervals*. New York City, New York: Springer-Verlag, 1985.
- [7] D. Siegmund, “Boundary crossing probabilities and statistical applications,” *The Annals of Statistics*, vol. 14, 06 1986.

- [8] T. Lai, “Sequential analysis: Some classical problems and new challenges,” *Statistica Sinica*, vol. 11, pp. 303–408, 04 2001.
- [9] H.-J. Kim, *Change point problems in regression*. PhD thesis, Stanford University, 1988.
- [10] L. Wasserman, *All of Statistics A Concise Course in Statistical Inference*. New York City, New York: Springer-Verlag, 2004.
- [11] T. Adão, J. Hruška, L. Pádua, J. Bessa, E. Peres, R. Morais, and J. Sousa, “Hyperspectral imaging: A review on uav-based sensors, data processing and applications for agriculture and forestry,” *Remote Sensing*, vol. 2017, p. 1110, 10 2017.
- [12] V. Konev and S. Pergamenshchikov, “On guaranteed estimation of the mean of an autoregressive process,” *Annals of Statistics*, vol. 25, 10 1997.
- [13] Mesiar and J. Špírková, “Weighted means and weighting functions,” *Kybernetika - Praha-*, vol. 42, 01 2006.
- [14] S. W. Smith, *The Scientist and Engineer’s Guide to Digital Signal Processing*. Poway, California: California Technical Publishing, 1999.
- [15] I. Lugo-Cárdenas, G. Flores, S. Salazar, and R. Lozano, “Dubins path generation for a fixed wing uav,” *International Conference on Unmanned Aircraft Systems*, no. 10.1109/ICUAS.2014.6842272, pp. 339–346, 2014.
- [16] H. Marino, M. Bonizzato, R. Bartalucci, P. Salaris, and L. Pallottino, “Motion planning for two 3d-dubins vehicles with distance constraint,” *IEEE/RSJ International Conference on Intelligent Robots and Systems*, no. doi: 10.1109/IROS.2012.6385866, pp. 4702–4707, 2012.