



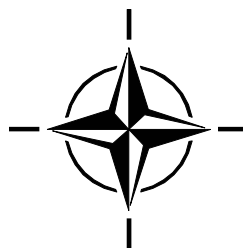
RTO TECHNICAL REPORT

TR-SCI-144

Integration of Systems with Varying Levels of Autonomy

(Intégration de systèmes à niveau
d'autonomie variable)

This Report was prepared by Task Group SCI-144 on “System-Level
Integration of Control plus Automation” and has been sponsored
by the Systems Concepts and Integration Panel.



Published September 2008





RTO TECHNICAL REPORT

TR-SCI-144

Integration of Systems with Varying Levels of Autonomy

(Intégration de systèmes à niveau
d'autonomie variable)

This Report was prepared by Task Group SCI-144 on “System-Level
Integration of Control plus Automation” and has been sponsored
by the Systems Concepts and Integration Panel.

The Research and Technology Organisation (RTO) of NATO

RTO is the single focus in NATO for Defence Research and Technology activities. Its mission is to conduct and promote co-operative research and information exchange. The objective is to support the development and effective use of national defence research and technology and to meet the military needs of the Alliance, to maintain a technological lead, and to provide advice to NATO and national decision makers. The RTO performs its mission with the support of an extensive network of national experts. It also ensures effective co-ordination with other NATO bodies involved in R&T activities.

RTO reports both to the Military Committee of NATO and to the Conference of National Armament Directors. It comprises a Research and Technology Board (RTB) as the highest level of national representation and the Research and Technology Agency (RTA), a dedicated staff with its headquarters in Neuilly, near Paris, France. In order to facilitate contacts with the military users and other NATO activities, a small part of the RTA staff is located in NATO Headquarters in Brussels. The Brussels staff also co-ordinates RTO's co-operation with nations in Middle and Eastern Europe, to which RTO attaches particular importance especially as working together in the field of research is one of the more promising areas of co-operation.

The total spectrum of R&T activities is covered by the following 7 bodies:

- AVT Applied Vehicle Technology Panel
- HFM Human Factors and Medicine Panel
- IST Information Systems Technology Panel
- NMSG NATO Modelling and Simulation Group
- SAS System Analysis and Studies Panel
- SCI Systems Concepts and Integration Panel
- SET Sensors and Electronics Technology Panel

These bodies are made up of national representatives as well as generally recognised 'world class' scientists. They also provide a communication link to military users and other NATO bodies. RTO's scientific and technological work is carried out by Technical Teams, created for specific activities and with a specific duration. Such Technical Teams can organise workshops, symposia, field trials, lecture series and training courses. An important function of these Technical Teams is to ensure the continuity of the expert networks.

RTO builds upon earlier co-operation in defence research and technology as set-up under the Advisory Group for Aerospace Research and Development (AGARD) and the Defence Research Group (DRG). AGARD and the DRG share common roots in that they were both established at the initiative of Dr Theodore von Kármán, a leading aerospace scientist, who early on recognised the importance of scientific support for the Allied Armed Forces. RTO is capitalising on these common roots in order to provide the Alliance and the NATO nations with a strong scientific and technological basis that will guarantee a solid base for the future.

The content of this publication has been reproduced directly from material supplied by RTO or the authors.

Published September 2008

Copyright © RTO/NATO 2008
All Rights Reserved

ISBN 978-92-837-0054-8

Single copies of this publication or of a part of it may be made for individual use only. The approval of the RTA Information Management Systems Branch is required for more than one copy to be made or an extract included in another publication. Requests to do so should be sent to the address on the back cover.

Table of Contents

| | Page |
|--|-------------|
| List of Figures | vi |
| List of Tables | ix |
| Programme Committee | x |
| | |
| Executive Summary and Synthèse | ES-1 |
| | |
| Chapter 1 – Introduction | 1-1 |
| 1.1 Terms of Reference and Definition of Scope | 1-1 |
| 1.2 System Complexity and the Need for a New Approach | 1-2 |
| | |
| Chapter 2 – Background: Evolution of System Integration | 2-1 |
| 2.1 Human-Machine Integration | 2-1 |
| 2.1.1 Transfer of Control to a Human Operator | 2-5 |
| 2.2 Case Studies | 2-7 |
| 2.2.1 Land | 2-7 |
| 2.2.1.1 Background | 2-7 |
| 2.2.1.2 UGV Case Studies | 2-9 |
| 2.2.2 Sea (DH) | 2-14 |
| 2.2.2.1 Theseus AUV | 2-15 |
| 2.2.3 Air and Space | 2-18 |
| 2.2.3.1 F-22A Flying Qualities Development | 2-18 |
| 2.2.3.2 Apollo and Space Shuttle | 2-19 |
| 2.2.3.3 Development of Fly-By-Wire Integration | 2-20 |
| | |
| Chapter 3 – Best Practices for System Integration | 3-1 |
| 3.1 Problems of Systems Engineering | 3-1 |
| 3.2 Systems Engineering Best Practices | 3-1 |
| 3.2.1 Land Systems | 3-4 |
| | |
| Chapter 4 – Complexity, Automation and Autonomy | 4-1 |
| 4.1 Introduction | 4-1 |
| 4.2 Complexity | 4-1 |
| 4.3 Man-Machine Systems | 4-3 |
| 4.3.1 Supervisory Control in Man-Machine System | 4-3 |
| 4.3.2 Adaptive and Intelligent Systems Grounding on Human Operator Experience as a Basis for System-Level Integration of Control | 4-6 |

| | | |
|--|---|------------|
| 4.3.2.1 | Intelligent Control Systems as a Reflection of Experienced Human Pilot Skills | 4-6 |
| 4.3.2.2 | Control Problems for Advanced Manned and Unmanned Aircraft – General Description | 4-6 |
| 4.3.2.3 | Semi-Soft Computing as a Basis for Implementation of Intelligent Control Systems | 4-7 |
| 4.3.2.4 | Adaptive and Intelligent Systems as a Basis for System-Level Integration of Control | 4-8 |
| 4.3.2.5 | Intelligent Control Techniques Based on Human Operator Experience | 4-14 |
| 4.4 | Autonomous Vehicles | 4-17 |
| 4.4.1 | Automation and Autonomy | 4-17 |
| 4.4.2 | Autonomy for Robotics Systems | 4-17 |
| 4.4.3 | Complexity for Autonomous Systems | 4-18 |
| 4.4.3.1 | Environmental Complexity | 4-18 |
| 4.4.4 | World Representation and Understanding | 4-19 |
| 4.4.4.1 | A Ground Vehicle Perspective | 4-20 |
| 4.4.4.2 | Understanding from the Machine or Computer Vision Perspective | 4-20 |
| 4.4.5 | Single and Multiple Platform Systems | 4-20 |
| 4.4.6 | Cooperative Control Operations | 4-21 |
| 4.4.6.1 | A Taxonomy of Teams | 4-21 |
| 4.4.6.2 | Team Coordination | 4-22 |
| 4.4.6.3 | Team Cooperation | 4-22 |
| 4.4.6.4 | Team Collaboration | 4-22 |
| 4.4.6.5 | Goal-Seeking Team Action | 4-23 |
| 4.4.6.6 | Non-Cooperative Behavior | 4-23 |
| 4.4.6.7 | Conflict of Interest | 4-23 |
| 4.4.6.8 | Distributed Decision and Control Systems | 4-24 |
| 4.4.6.9 | Complexity in Cooperative Teams | 4-26 |
| 4.4.6.10 | Task Coupling | 4-27 |
| 4.4.6.11 | Uncertainty | 4-27 |
| 4.4.6.12 | Communication | 4-27 |
| 4.4.6.13 | Partial Information | 4-28 |
| 4.4.6.14 | Operator | 4-29 |
| 4.4.6.15 | Adversary Action | 4-29 |
| 4.4.6.16 | Algorithms for Cooperative Control | 4-30 |
| 4.4.6.17 | Coupling of Tasks vs. Decentralization vs. Communication | 4-31 |
| 4.5 | Artificial Intelligence | 4-32 |
| 4.5.1 | Architecture | 4-32 |
| 4.5.2 | Uncertainty | 4-32 |
| 4.5.3 | Deliberative Architectures | 4-33 |
| 4.5.4 | Reactive Architectures | 4-33 |
| 4.5.5 | Three-Layer Architectures | 4-33 |
| 4.5.6 | Cognitive Architectures | 4-33 |
| Chapter 5 – Mission Management and Robustness | | 5-1 |
| 5.1 | Introduction | 5-1 |
| 5.1.1 | Risk Management | 5-1 |

| | | |
|--|--|------------|
| 5.1.2 | Uncertainty Management | 5-2 |
| 5.1.2.1 | Definitions of Uncertainty | 5-3 |
| 5.1.2.2 | Sources of Uncertainty | 5-5 |
| 5.2 | Review of Current Approaches | 5-5 |
| 5.3 | Fault Detection and Isolation | 5-8 |
| 5.3.1 | Review on Fault Tolerance | 5-9 |
| 5.3.1.1 | Fault Classification | 5-9 |
| 5.3.1.2 | How to Defeat Faults | 5-10 |
| 5.4 | Fault and Failure Accommodation | 5-13 |
| 5.4.1 | Robustness and Redundancy | 5-14 |
| 5.4.2 | Fault-Tolerant Control and Reconfiguration | 5-14 |
| 5.5 | Prognostics | 5-15 |
| 5.5.1 | Prognostic Assessment | 5-16 |
| 5.5.2 | Development of Diagnostic Systems as a Precursor for Prognostics | 5-17 |
| 5.5.3 | US Air Force CBM+ | 5-21 |
| 5.5.4 | Integration with the Logistics Infrastructure | 5-22 |
| 5.6 | Information Fusion | 5-23 |
| 5.6.1 | Definition of the Fusion Process | 5-24 |
| 5.6.2 | Data Types | 5-25 |
| 5.6.3 | Categories of Fusion Concepts | 5-26 |
| 5.6.4 | Data Acquisition Problems | 5-26 |
| 5.6.5 | Data Management Policy | 5-28 |
| 5.6.6 | AFRL GCSS Research Environment | 5-28 |
| 5.7 | Applications in Human-Machine Interface | 5-29 |
| 5.8 | System Health Management | 5-30 |
| 5.9 | Integrated Systems Health Management | 5-34 |
| 5.9.1 | ISHM Research Requirements | 5-35 |
| 5.9.2 | AFRL ISHM Programs | 5-37 |
| 5.9.3 | Obstacles to Health Management Research | 5-38 |
| 5.10 | System Test, Verification and Validation and Certification Issues | 5-38 |
| 5.10.1 | Basic Definitions | 5-38 |
| 5.10.2 | Verification Techniques | 5-38 |
| 5.10.3 | Validation Techniques | 5-39 |
| 5.10.4 | Certification Process | 5-40 |
| 5.11 | Summary | 5-41 |
| Chapter 6 – Conclusions and Recommendations | | 6-1 |
| 6.1 | Background: Achievements in Flight Control System Design and Their Use for Future Aircraft | 6-1 |
| 6.2 | Issues and Challenges | 6-2 |
| 6.2.1 | Systems Level Challenges | 6-3 |
| 6.3 | Conclusions | 6-4 |
| Chapter 7 – References | | 7-1 |

List of Figures

| Figure | | Page |
|---------------|--|-------------|
| Figure 1.1 | Pictorial Representation of Multiple Vehicles (Land, Sea and Air) Communicating and Coordinating | 1-2 |
| Figure 1.2 | Typical Structural Hierarchies for Multiple Autonomous Vehicles | 1-3 |
| Figure 1.3 | Time Scale vs. Mass of Some Representative Vehicle Types in the Land, Air and Sea Environments | 1-4 |
| Figure 2.1 | Man Machine System | 2-1 |
| Figure 2.2 | The Human Operator Subsystems | 2-2 |
| Figure 2.3 | Driver Vehicle System | 2-4 |
| Figure 2.4 | The Internal Helmsman Model | 2-5 |
| Figure 2.5 | Carnegie Mellon's Task Control Architecture for the Ambler Hexapod | 2-8 |
| Figure 2.6 | Carnegie Mellon's Distributed Architecture for Mobile Navigation for the NAVLAB | 2-8 |
| Figure 2.7 | The Mars Exploration Rover, Courtesy NASA/JPL-Caltech | 2-9 |
| Figure 2.8 | C.M.U.s Sandstorm Unmanned Ground Vehicle | 2-11 |
| Figure 2.9 | Stanford's Stanley Unmanned Ground Vehicle | 2-11 |
| Figure 2.10 | DRDC's Tele-Operated Multi-Agent Tactical Vehicle | 2-12 |
| Figure 2.11 | DRDC's Raptor UGV | 2-13 |
| Figure 2.12 | Raptor UGV Systems | 2-13 |
| Figure 2.13 | Two Examples of Autonomous Underwater Vehicles | 2-14 |
| Figure 2.14 | The Multi-Mission Spartan Vehicle and the Remote Minehunting Vehicle Dorado | 2-15 |
| Figure 2.15 | Theseus Schematic | 2-16 |
| Figure 2.16 | The First Fly-By-Wire Aircraft, T-4 | 2-22 |
| Figure 2.17 | The First Mass Production Fly-By-Wire Aircraft, F-16 | 2-22 |
| Figure 2.18 | Fly-By-Wire Aircraft, Su 27 | 2-23 |
| Figure 2.19 | Prefilter for Fly-By-Wire System | 2-23 |
| Figure 2.20 | Confirmation of Fly-By-Wire and Mechanical Systems | 2-24 |
| Figure 2.21 | Enlargement of Control Surfaces and Modes | 2-24 |
| Figure 2.22 | Integration of Flight Control System and Stall Warning System | 2-25 |
| Figure 2.23 | Su 30 MK Aircraft | 2-25 |
| Figure 2.24 | The Buran Prefilter and its Modification | 2-26 |
| Figure 2.25 | Self-Adaptive Actuator | 2-26 |
| Figure 4.1 | Embedded Cognitive Control Engineering Discipline | 4-2 |
| Figure 4.2 | Man-Machine System in Supervisory Control | 4-4 |
| Figure 4.3 | Human Operator Schematic | 4-4 |

| | | |
|-------------|---|------|
| Figure 4.4 | Detected Time vs. Normalized Workload | 4-5 |
| Figure 4.5 | Relationships between Soft Computing (SC), Extended Soft Computing (ESC) and Semi-Soft Computing (SSC) | 4-7 |
| Figure 4.6 | Intelligent and Conventional Control Systems versus Soft and Semi-Soft Computing | 4-8 |
| Figure 4.7 | Generalized Structure of Autonomous Agent and its Interaction with Environment | 4-12 |
| Figure 4.8a | Multiagent Architectures Based on Direct Interaction With the Environment and Without any Coordination Between Agents | 4-12 |
| Figure 4.8b | Multiagent Architecture Involving Agent Cooperation via Blackboard | 4-13 |
| Figure 4.9 | A Low Complexity Environment | 4-18 |
| Figure 4.10 | A Medium Complexity Environment | 4-19 |
| Figure 4.11 | A High Complexity Environment | 4-19 |
| Figure 4.12 | Team Decision and Control Metrics | 4-25 |
| Figure 4.13 | Hierarchical Decomposition | 4-25 |
| Figure 4.14 | Notional Coordination Metric | 4-28 |
| Figure 4.15 | Family of Receiver Operating Characteristic (ROC) | 4-29 |
| Figure 4.16 | Coupling–Decentralization–Communication Trade Space | 4-31 |
| Figure 5.1 | General DoD Risk Assessment Process | 5-2 |
| Figure 5.2 | From Data to Knowledge | 5-4 |
| Figure 5.3 | Location of Potential Fault in the Control System | 5-9 |
| Figure 5.4 | Fault Classifications | 5-10 |
| Figure 5.5 | Taxonomy of System-Failure Response Strategies | 5-11 |
| Figure 5.6 | Graphical Presentation of Fault Detection and Identification | 5-11 |
| Figure 5.7 | Conceptual Structure of Model-Based Fault Diagnostics | 5-12 |
| Figure 5.8 | General Scheme of Model-Based Fault Detection | 5-12 |
| Figure 5.9 | One Scheme of Active Fault Tolerant Control | 5-13 |
| Figure 5.10 | Block Structure of Reconfigurable Fault Tolerant Control System | 5-15 |
| Figure 5.11 | System Health Management | 5-16 |
| Figure 5.12 | Prognostic Regions | 5-17 |
| Figure 5.13 | Wiring Diagnostic Module | 5-18 |
| Figure 5.14 | DoD CBM+ Initiative Listed CBM+ Capabilities or Programs Being Targeted under the DoD's FLE | 5-21 |
| Figure 5.15 | GCSS AF Data Services Capability Concept | 5-23 |
| Figure 5.16 | Methods of Information Fusion | 5-24 |
| Figure 5.17 | Information Fusion Levels | 5-25 |
| Figure 5.18 | Acquisition Data Process | 5-25 |
| Figure 5.19 | Legacy Propulsion IT System Interfaces: Initial GCSS Implementation of Legacy Central Data Systems Data | 5-27 |
| Figure 5.20 | Integrated Vehicle Health Management (IVHM) and Condition Based Maintenance (CBM) | 5-29 |

| | | |
|-------------|---|------|
| Figure 5.21 | Human-Machine Interface | 5-30 |
| Figure 5.22 | IVHM System and Advanced Diagnostics/Prognostics | 5-32 |
| Figure 5.23 | Health Monitoring of an Aircraft Actuator | 5-33 |
| Figure 5.24 | Health Monitoring System Example – Motor Pump Setup | 5-34 |
| Figure 5.25 | AFRL ISHM Program Initiative: Depicts AFRL ISHM CBM+ Technology Programs by Capability Development through Fiscal Year 2017 | 5-35 |
| Figure 5.26 | Vision for an Autonomic Information Infrastructure that Fully Integrates and Automates Logistics Functions | 5-37 |
| Figure 5.27 | Generic Verification and Validation Process shown as “V” | 5-40 |
| Figure 6.1 | Automation Functionalities vs. Time | 6-2 |
| Figure 6.2 | Cooperation is at the Intersection of Disciplines | 6-3 |

List of Tables

| Table | | Page |
|--------------|---|-------------|
| Table 2.1 | Thrust and Pitch Attitude Before the Accident | 2-6 |
| Table 2.2 | Theseus Vehicle Characteristics | 2-17 |
| Table 5.1 | Risk Assessment, Abatement and Management | 5-1 |
| Table 5.2 | Types of Uncertainty | 5-3 |
| Table 5.3 | Fault Classification | 5-9 |

Programme Committee

The members of Task Group SCI-144 were:

| | | |
|-----------------------------------|--------------|-----------------|
| Dr. David J. Moorhouse (Chairman) | AFRL | USA |
| Dr. Siva Banda | AFRL | USA |
| Dr. Al Behbahani | AFRL | USA |
| Mr. Tom Black | ASC | USA |
| Mr. Bruce Digney | DRDC | CAN |
| Mr. Alexander Efremov | MAI | RUS |
| Mr. Kris Ellis | DRDC | CAN |
| Mr. Louis Erkelens | NLR | NLD (part-time) |
| Dr. Mario Innocenti | U. of Pisa | ITA |
| Mr. Frank Thielecke | DLR | DEU (part-time) |
| Mr. David Hopkin | DRDC | CAN |
| Dr. Brian White | Cranfield U. | GBR |

Additional contributions to this report were made by (in alphabetical order):

| | | |
|-----------------------|--------------|-----|
| Mr. Greg Broten | DRDC | CAN |
| Dr. Phil Chandler | AFRL | USA |
| Ms. Cheryl Holtz | AFRL | USA |
| Dr. Mark Mears | AFRL | USA |
| Mr. Gary Smith | AFRL | USA |
| Dr. Tiumentsev | MAI | RUS |
| Dr. Antonios Tsourdos | Cranfield U. | GBR |

Integration of Systems with Varying Levels of Autonomy

(RTO-TR-SCI-144)

Executive Summary

This Task Group was formally initiated in 2004, to address higher-level issues of controlling vehicles in a system of systems. The subject area of vehicle control continues to expand rapidly. New techniques have been developed, plus major improvements in how this expertise can be applied to solve new problems. Continued development of this core competency is essential to harness these technological advancements. Even greater benefits can be achieved if we integrate the technology developments across all vehicle classes and domains, including the cooperative operation of dissimilar vehicles. It is no longer appropriate to design military vehicles as individual entities, they must operate in the total system environment. The ultimate payoff is design processes to achieve equal or better system capabilities at more affordable cost.

The first part of the report begins with a historical background and the evolution of systems engineering. There is a discussion of case studies for land, sea and air vehicles, detailing lessons learned from various programs. There is a chapter discussing both negative and positive aspects on systems engineering, followed by a presentation of recommended best practices.

The second part of the report continues with a chapter discussing complexity and automation, considering man/machine interface, single vs. multiple vehicles, etc. Then there is discussion of mission management, especially robust design of autonomous systems. This leads to addressing certification issues, such as verification and validation of non-deterministic systems, followed by discussion of issues and challenges.

The Task Group members originally laid out this report to present an assessment of design methods, but no correlation has been found between the method used and the problems of the past, or the successes. The benefits of advanced methods are primarily in terms of greater efficiency, with a shorter design cycle translating into cost savings. No method will guarantee success by itself, since it still needs the correct design criteria, development rigor and disciplined application of the other components of the best practices process as defined in this report.

The report ends with conclusions and suggestions for required future research.

Intégration de systèmes à niveau d'autonomie variable

(RTO-TR-SCI-144)

Synthèse

Ce Groupe de travail a été officiellement créé en 2004, afin d'aborder les questions de niveau supérieur relatives au contrôle des véhicules dans un système de systèmes. Le domaine du contrôle des véhicules continue de croître rapidement. De nouvelles techniques ont été développées, et d'importantes améliorations ont été apportées quant à la manière d'appliquer cette expertise à la résolution de nouveaux problèmes. Le développement continu de cette compétence de base est indispensable à l'utilisation de ces avancées technologiques. De plus grands avantages encore peuvent être obtenus si l'on intègre les développements technologiques dans tous les domaines et toutes les classes de véhicules, y compris l'exploitation conjointe de véhicules dissemblables. Il n'est désormais plus approprié de concevoir les véhicules militaires comme des entités individuelles : ils doivent fonctionner dans l'environnement du système global. Le résultat ultime est l'élaboration de processus de conception permettant d'obtenir des capacités égales ou supérieures pour le système, à un coût plus abordable.

La première partie du rapport débute par un rappel historique et par l'évolution de l'ingénierie des systèmes. Elle comprend ensuite un débat sur des études de cas de véhicules terrestres, maritimes et aériens, détaillant les leçons tirées de divers programmes. Un chapitre expose les aspects positifs et négatifs de l'ingénierie des systèmes, et se conclut sur la présentation des meilleures pratiques recommandées.

La seconde partie du rapport commence par un chapitre débattant de la complexité et de l'automatisation, traitant de l'interface homme/machine, des véhicules uniques opposés aux véhicules multiples, etc. S'ensuit un débat sur la gestion de mission, en particulier sur la conception robuste de systèmes autonomes. Cela conduit à aborder les questions de certification – telles que la vérification et la validation de systèmes non déterministes – et à évoquer les problèmes et les défis.

Les membres du Groupe de travail ont initialement préparé ce rapport en vue de présenter une évaluation des méthodes de conception, mais aucune corrélation n'a été découverte entre la méthode utilisée et les problèmes – ou succès – du passé. Les avantages des méthodes de pointe résident principalement dans une plus grande efficacité, doublée d'un cycle de conception plus court, ce qui se traduit par une diminution des coûts. Aucune méthode en elle-même ne peut garantir le succès, dans la mesure où les bons critères de conception, la rigueur de développement et l'application disciplinée des autres composants du processus de meilleures pratiques sont toujours nécessaires, comme défini dans ce rapport.

Le rapport s'achève sur la présentation des conclusions et des suggestions pour les futures recherches à mener.

Chapter 1 – INTRODUCTION

This Working Group was formally initiated in 2004, with the genesis and rationale contained in a Pilot Paper written in September 2002. It is appropriate, therefore, to start with some thoughts from that document. The document cited a need for stressing the system engineering process. As the focus changed to the current report title, it became more obvious that unmanned vehicles would be a primary focus. With the increased interest and emphasis on unmanned systems over the last decade, the question arises as to whether totally new design and development processes are required. While there are competing views, the authors of this report believe that while unmanned vehicles open new areas of “design space”, the systematic engineering disciplines associated with their development remain largely unchanged and may be extended from those already existent for comparable manned systems. The area which has opened for new development tools and techniques is the integration of individual vehicles and (where applicable) control stations into a system of systems. In many cases, the individual elements of this system of systems are dissimilar from each other and other components of a broader set (such as a force structure) with which they must be integrated. In some cases, the individual elements (vehicles) may even operate in dissimilar media – land vehicles may be integrated with airborne vehicles and water (or even submarine) vehicles. The authors did attempt to address the similarities or differences between air, land, and sea systems. However, in spite of those differences, the authors assert that the classic systems engineering processes, properly understood and applied, may be used as the “jumping off point” for the larger systems of systems development and integration challenge.

It is intended that this report should be a tool for scientists and engineers working both in research and development and in systems acquisition. For the former, it should denote areas where knowledge, technology or tools are poorly developed or even lacking. These areas become candidate areas for research. For the latter group, it is intended that this report represent a compendium of best practices and state of the art from the NATO technical community.

1.1 TERMS OF REFERENCE AND DEFINITION OF SCOPE

There have been a variety of recent activities within RTO devoted to control issues. Task Group SCI-026 focused on design issues of air vehicle control and produced a Best Practices Guide for design of flight control systems (Anon 2000). In May 2000, the AVT Panel held a symposium devoted to ‘Active Control Technology’ (Reference Anon 2001). In this symposium, papers discussed various extensions to classical flight control. As the Technical Evaluation Report said: “Within the realm of aircraft technology, this symposium was of very high quality”. But then a further comment was: “It is suggested that a symposium would be possible in another five years and show similar progress. At that time we should include the technologies of all vehicle classes”.

The most recent effort was Task Group SCI-053, “Vehicle Dynamics, Modelling and System ID, Control and Handling Qualities”. This Task Group produced a Technical Report in 2002 (Reference Anon 2002) with a significant advance by identifying similarities and differences in those subjects between land, air and sea domains. In addition, a symposium in May 2002 (Reference Anon 2003) with papers in the four TG subjects addressing land, air, sea and (minimal) space aspects. That report contains a very good coverage of vehicle related issues. It was suggested that there be a requirement to address higher-level issues of controlling vehicles in a system of systems.

The subject area of vehicle control continues to expand rapidly. New techniques have been developed, plus major improvements in how this expertise can be applied to solve new problems. Continued development of this core competency is essential to harness these technological advancements. Even greater benefits can be achieved if we integrate the technology developments across all vehicle

INTRODUCTION

classes and domains, including the cooperative operation of dissimilar vehicles. It is no longer appropriate to design military vehicles as individual entities, they must operate in the total system environment. The ultimate payoff is design processes to achieve equal or better system capabilities at more affordable cost.

The emergence of RTO as the organisation to co-ordinate research and technology activities over the breadth of all the NATO military environments (land, air, sea and space) presents a unique challenge. The core competencies embodied in the technical areas of vehicle control, command and control, and automation are at the heart of the mission performance of all military vehicles. The experts currently active in RTO activities and in the process of solving aircraft issues and performing research in these areas have counterparts addressing analogous issues for land, sea and space vehicles. These experts all face similar challenges, and the co-ordination of the different activities will have benefits that can be applied to all environments.

The contents of this technical report start with a background of matured technologies such as man-machine integration. It was decided to consider a variety of subjects and current efforts: Swarms of Unmanned Vehicles, Safe Mixing of Manned and Unmanned Vehicles, Automated Mission Performance of Unmanned Vehicles, and Optimizing Vehicle Convoy Performance. There is then a discussion of future potential benefits from system-level controls integration, automation and optimization, e.g.: Fully Intelligent Control, and others. An objective of the report is a complete interchange of expertise/information between land/air/sea/space communities considering ‘control’ as only a component in a system-level approach to the war fighter needs.

1.2 SYSTEM COMPLEXITY AND THE NEED FOR A NEW APPROACH

Complex systems consisting of single vehicles or multiple vehicles have to operate at several levels. These levels describe the capability of the systems within the appropriate domains. To illustrate this, consider Figure 1.1.

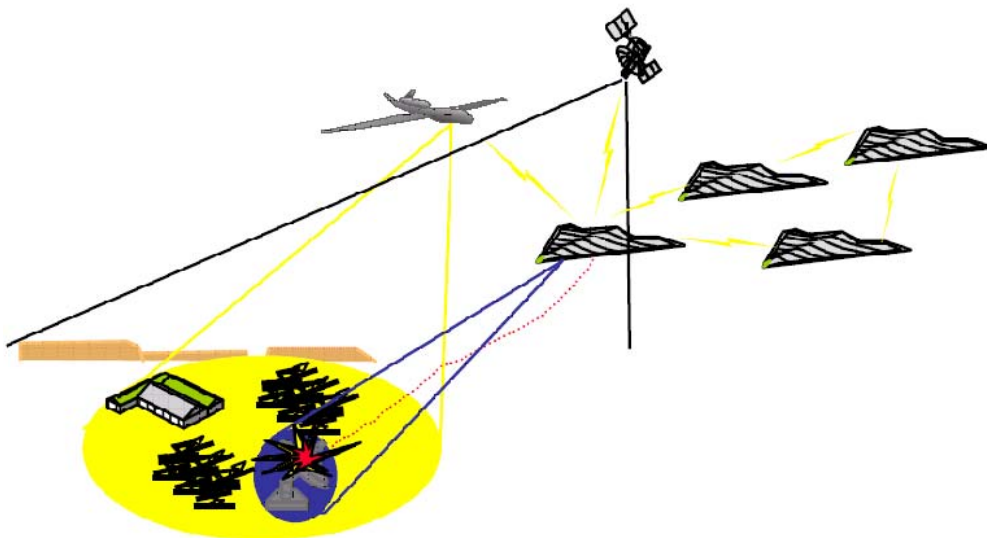


Figure 1.1: Pictorial Representation of Multiple Vehicles (Land, Sea and Air) Communicating and Coordinating.

This shows the operation of a vehicle or collection of vehicles which is mobile in an uncertain, unstructured environment with a set of sensors that inform the system about the vehicle state and the

external environment. The information generated within the individual vehicles and that shared over the totality of vehicles is inherently different in quality, accuracy and latency. Hence such a multi-vehicle system becomes actually a system of systems. There are several layers of complexity within each system and sub-system that need to work effectively together in a well behaved and predictable manner from the design perspective. Additionally, if this system has some degree of autonomy, it will have to perform deliberate actions as well as reacting to the environment in a capable manner. Hence the design of these complex systems is that of designing capability such that the system will perform and react in a controlled recognizable manner. The overriding requirement is to design the system such that no unexpected behaviour should result. If this system is controlled and operated by a human, its behaviour should be recognizable and enable the operator to recognize capability and be able to use and exploit it. This principle can also be applied to the structure of the complex system itself.

To illustrate the above concepts, consider a single autonomous vehicle within an autonomous multi-vehicle system. In order to function, several layers of activity are required, each layer having a technical domain in which problems have been posed and solved mainly in isolation in the literature. A typical structure is shown in Figure 1.2.

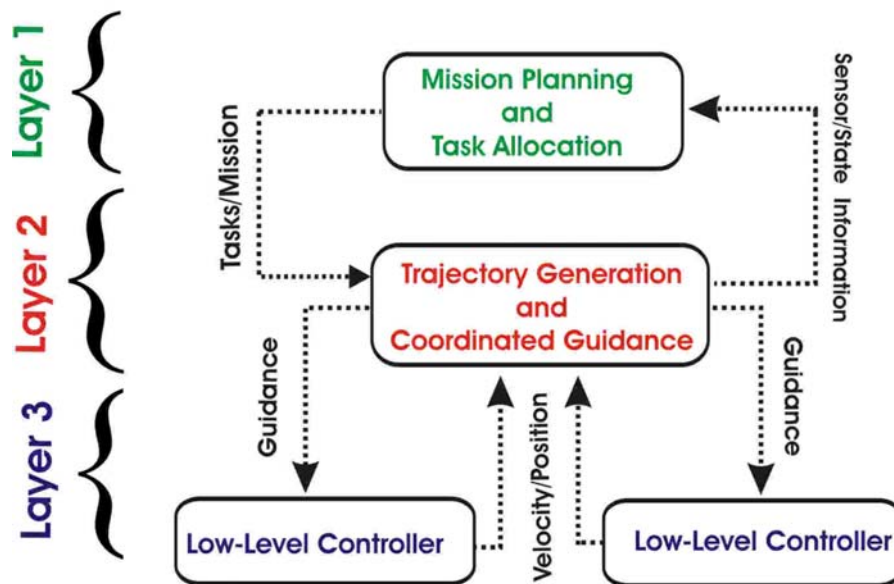


Figure 1.2: Typical Structural Hierarchies for Multiple Autonomous Vehicles.

To function well as an autonomous system each level in the figure should be able to recognize, use and exploit the capability of each level below it. The lower layers should in turn, have knowledge of the upper levels, sufficient to describe their own capability in terms that the upper layers can exploit. Hence the interface that translates between levels requires special attention, in that a common view of the capability exists across the boundaries. Hence control at each level needs to be integrated with levels both above and below its own level.

Each level will have its own language, whether that is a mathematical framework or an experiential description that will need to be integrated sufficiently to enable the overall system capability to be understood. The mathematical tools for each level are fundamentally different and it is unlikely that a grand unifying approach will capture the system design process. Hence continuous differential equation representations are useful at level 1, where autopilots and parametric robustness, parametric identification techniques are used. Level 2 is primarily kinematic in that planning routes through uncertain environments, re-planning as pop-up threats and other obstacles are detected and collision avoidance is performed. The top

INTRODUCTION

level deals with decision making and mission planning, where event driven reactive behaviours are dealt with, together with deliberative behaviours in fulfilling tasks. The mathematical framework for each of these levels is fundamentally different, but assumptions about the behaviour of each level are made in each framework. For example, for route planning assumptions about manoeuvre capability of the airframe enter at level 1 in the form of limits on lateral acceleration or incidence. At level 2 this will be translated into the kinematic domain as a constraint on trajectory curvature. This will promulgate into level 3 as a constraint on immediately reachable areas. It seems important that these different descriptions are consistent, and easily used and exploited at each level. Essentially, the structure is the same as for a single vehicle, with the difference being in the abstraction level of the information, together with the latency.

One issue that will be important for these multiple vehicle systems is the integration, i.e. command and control, of totally dissimilar vehicles. Representative time scales (defined as L/U , where L is typically length, and U is forward speed) and masses for some vehicle categories of interest are compared in Figure 1.3, (Reference Anon 2002). Since this is a variation of the classic transport efficiency diagram, the two variables are reasonably well correlated for each vehicle category. However, there is some distinction between the ranges for military and civilian fixed-wing aircraft because of speed, and between military and civilian surface ships because of mass. Note that the time scale covers several orders of magnitude. In addition, if L/U represents a medium length time scale for a particular vehicle, then the system design problem includes a number of other different time scales to consider, such as:

- Time to complete a representative manoeuvre – typically long.
- Vehicle responses – roll period, etc.: typically medium scale.
- Time history (memory) effects – trailing vortex interactions, etc.: typically medium.
- Vehicle subsystem responses – land vehicle suspension period, etc.: typically short scale, but can vary considerably.
- Sensing and control system responses – short to very short.

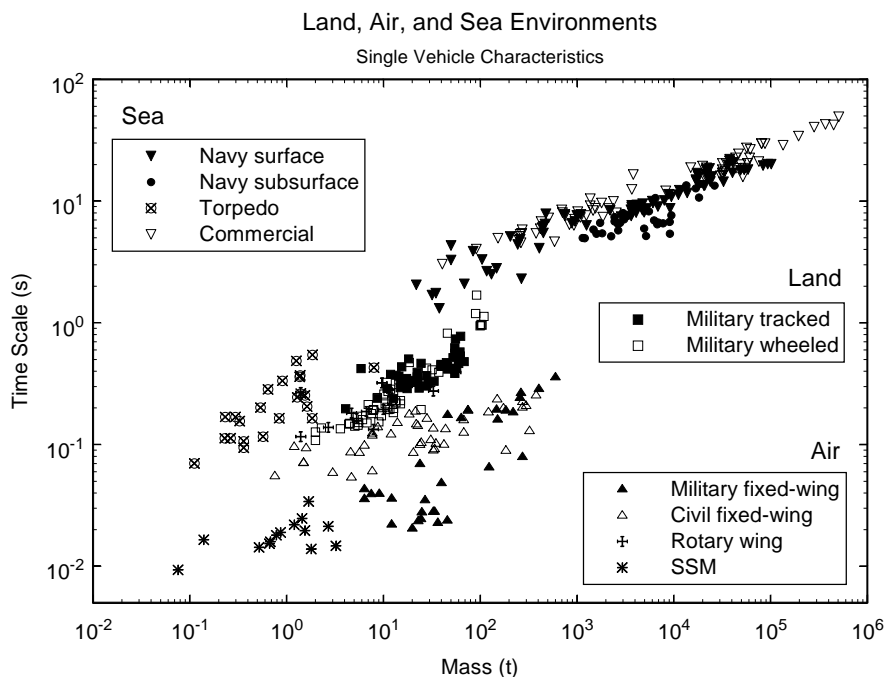


Figure 1.3: Time Scale (Sec) vs. Mass (Tonnes) of Some Representative Vehicle Types in the Land, Air, and Sea Environments.

In a few cases, subsystem characteristic time scales e.g., that of a tow cable, may be much longer than that of the vehicle(s) in the system.

Including the full control bandwidth will generally add one or two lower orders of magnitude to time scales in a simulation. Such a variation, or even the variation in fundamental time scales in a multi-vehicle simulation, will result in “stiff” equations of motion that require relatively inefficient and complex implicit methods to integrate.

For multiple vehicle systems there will also be issues related to effective communication between vehicles. For a single vehicle the design structure will be fixed and usually fit for its purpose. Hence, each layer will have a communication speed and protocol that will integrate by design. For multiple vehicle systems communication will be more problematic in that there will be bandwidth limitations, availability of channels and possible interference from natural hazards such as weather, buildings, terrain or deliberate jamming and spoofing. Such multiple vehicle systems must therefore have integrated capability under all conditions. This will range from each vehicle operating as an independent autonomous system to full linked capability when coordinated control and actions give more effective performance. The multiple vehicle system must be capable of defined performance and capability over the complete spectrum of integration conditions.

The design challenge is to review existing techniques and to propose new approaches to the design of autonomous control systems in such an unstructured and uncertain operating environment, and this will be presented in the next chapters.



Chapter 2 – BACKGROUND: EVOLUTION OF SYSTEM INTEGRATION

The system of systems consideration and integration of different systems in vehicle design have a history. There are discussed the historical evaluation of systems integration including man-machine systems, fly-by-wire systems and other examples.

2.1 HUMAN-MACHINE INTEGRATION

One of the bright examples of system integration experience was integration of man and machine in manual control tasks. The main components of this system (see Figure 2.1) are: display, human-operator (pilot, helmsman, driver), manipulator, controlled element.

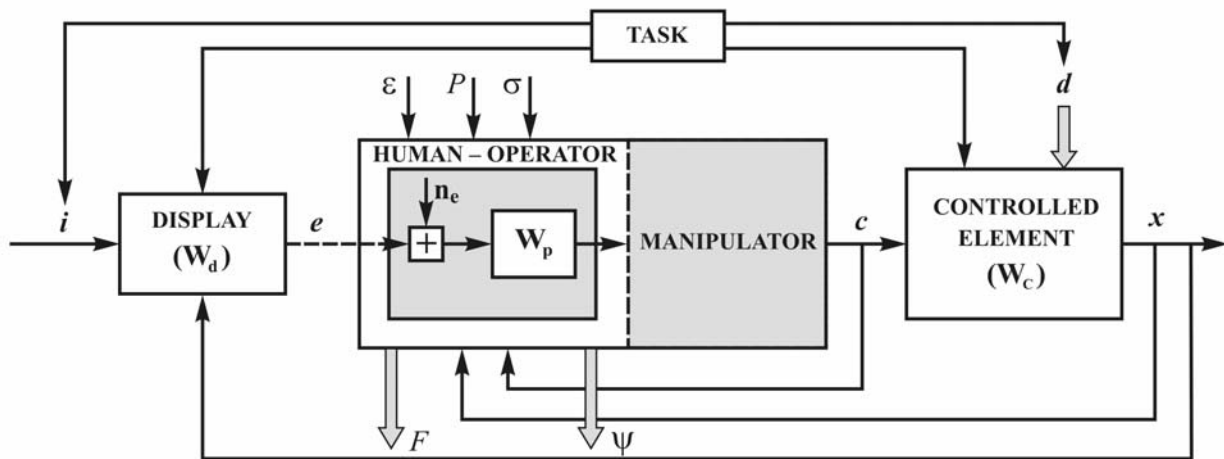


Figure 2.1: Man Machine System.

The specific peculiarities of this system are the following: any man-machine-system is a system of the systems.

Any component of it is the complex system. One of the more complicated components is the human-operator, characterized by three types of responses: control, psychological and psychophysiological characteristics. The operator model characterized by his control response on visual, vestibular and kinesthetic cues is shown on Figure 2.2. This model reflects the major processes taking place in perception, motor and central-nervous systems. The other component of this system is the controlled element consisted of the vehicle and control system (computer, filters, feedback, prefilters, actuators etc.).

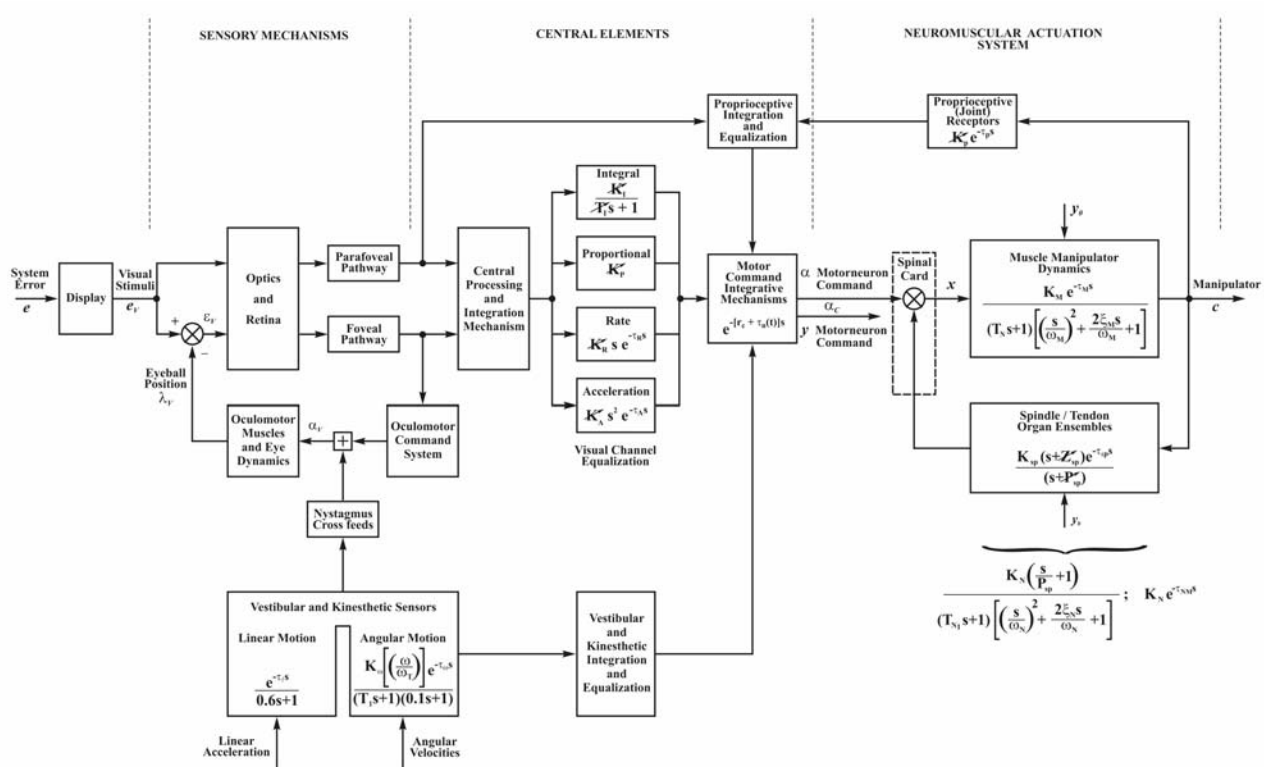


Figure 2.2: The Human Operator Subsystems.

Display is also complex system, demonstrated the different phase coordinates, director signals generated by the computer.

- The change of task (piloting task) causes the change of so-called man-machine system variables. Two of them influence on the system characteristics considerably – task variables including display, controlled element dynamics and input signal and man’s center inner variables. The motivation or levels of training goals of the mission are related to the last.
- **Adaptation of human-operator behavior to man-machine variables.** If the parameters of the variables will be changed the human-operator demonstrates the change of all his pilot response characteristics too.

There are considered three types of pilot adaptation:

- **Parametrical Adaptation.** A change of any man-machine system variable parameter causes the change of human-operator control response characteristics (his describing function, special density of remnant).
- **Structural Adaptation.** For the different task or task variables (for examples vehicle dynamics) human-operator can closes different loops or/and choices the best type of behavior (compensatory, pursuit etc.).
- **Goal Adaptation.** A change of task accompanies by change of the goals (requirements to the accuracy, for example).

The adaptation is the more remarkable feature and it was investigated attentively with goal to decide the task of integration of all component of human-operator-vehicle system allowed to get the highest efficiency of the mission and its safety.

The first investigations on measurement of human-operator control response characteristics were fulfilled in World War II, [Tustin 1944 and 1947].

It became possible to do in that period, because of the theory of control began to develop and got the practical usage. Except it the progress in computers allowed to extend considerably the researches on man-machine system, to define the main principles of human operator behavior, to expose his main feature – adaptation to task variables associated with operator’s attempt to keep approximately the same man-machine close-loop system characteristics. All these fundamental knowledge received by D. McRuer and his colleagues from System technology Inc. led to creation of mathematical model of control response characteristics. All these models were linear and based on classical theory of control. The definition of models parameters was fulfilled with use of so-called «adjustment rules» [McRuer et al, 1965]. These classical models differed by level of complexity describe the main regularities in pilot describing function in crossover frequency range where they have good agreement with experimental results. Therefore they call these models as “crossover models”.

The pioneer stage on human-operator behavior investigations was finalized to the middle of sixties in the last century, when these models were created. The results of those researches were generalized initially in [McRuer et al 1965 and 1967] and then later in [McRuer 1973] too. To that period the developed models and exposed regularities were used widely to the different applied manual control tasks. One of them was the definition of controlled element dynamics provided the simplest type of human behavior. It was determine that in compensatory tasks human operator demonstrated the “proportional (simplest) type”,

($W_p = K_p e^{-s\tau}$) in crossover frequency range when he control the rate type of the vehicle dynamics

($W_c = K / s$). Such standard controlled element dynamics was used in aviation widely: in flight control system design for choice of laws allowed to approach the vehicle dynamics to such standard in crossover frequency range [Ashkenas, Hoh, McRuer et al 1988 and Efremov]. The same ideology was used for display indicator law development [McRuer et al 1968, Weir et al and Klein et al]. The idea of the best integration of pilot’s action and flight control system potentialities was used in several researches in development of prefilters [Bushgens et al] and new types of manipulators with changeable stick stiffness [Efremov 1992]. The accuracy in precise tracking tasks and stability of closed loop system depends on integration of aircraft flying qualities with the pilot activities. Such peculiarity is the reason of researches on development of criteria for prediction of flying qualities and pilot-induced oscillation tendency. At least several of them were proposed in the last several decades (see Neal and Smith, Hess and Efremov 1996).

At the end of sixties the new approach to human-operator mathematical modeling based on modern control theory was developed [Baron]. It was modified several times [Levison, Thompson and Davidson] and used widely for different manual control tasks: prediction of flying qualities [Bacon], display design [Kleinman], flight control system [Schmidt, Garg]. However the problems in definition of cost function weighting coefficients and disagreement of model and experimental data in low frequency range limited the usage of this approach for prediction of results in applied investigations. In [Efremov et al 1988] it was offered the modification of approach allowed to improve the agreement between mathematical modeling and experimental data and accuracy in prediction of flying qualities for superaugmented aircraft.

The modification of classic approach to description of pilot behavior is the structural model developed [Hess 1978] at the end of seventies last century. This model takes into account pilot ability to use kinesthetic cues for generation pilot control response. It has high potentiality in improvement of agreement with experimental data [Hess 1979 and 1984]. The modified version of this model differed by the procedure of determination of model parameters and some changes in structure were offered. This modified model was used for development of criteria for prediction of flying qualities in pitch control tracking task.

BACKGROUND: EVOLUTION OF SYSTEM INTEGRATION

There were used different criteria for definition of the best way in integration of pilot with aircraft. One of them is the minimum of pilot opinion rating (POR), proposed by Anderson and Dillow. They developed the technique for parameter optimization based on classic model of pilot describing function. This procedure was modified later by use optimal control model of pilot behavior.

The main experience in integration of human-operator and machine was obtained in aviation because of many piloting tasks are characterized by pilot-aircraft closed loop system. The general ideology of such integration was optimization of all technical elements of man machine system: controlled element dynamics, display manipulator, provided necessary accuracy, and flight safety with minimum human operator workload. The success of such ideology in aviation defines the interest and its usage for integration of other types of human operator (driver, helmsman) with the other vehicles.

These types of human-operator-vehicle systems have some peculiarities. The driver-vehicle systems are characterized by more pursuit (and preview) type in majority driven tasks then in comparison with the compensatory type of system, which is more typical to pilot aircraft system. It takes place for turning, ramp entry and exit, precise course control, overtaking and passing. In some of driving tasks or maneuvers (lane change, evasive steering) the precognitive control presents as one of the driver control modes. In general driver-vehicle system is shown on Figure 2.3.

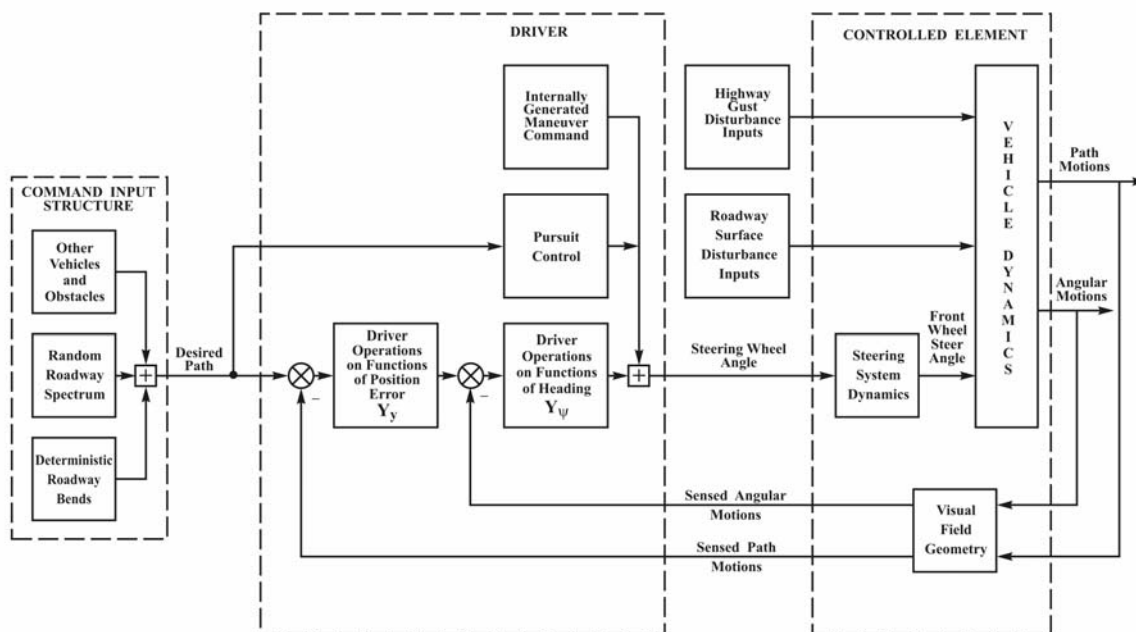


Figure 2.3: Driver Vehicle System.

Driver as an operative element in the system adapts and manipulates his dynamic characteristics to satisfy the key guidance and control requirements for the driver/vehicle system. Stated verbally, the guidance and control requirements for lateral position (path control) are:

- Establish and maintain the automobile on a specified spatial pathway;
- Reduce path error to zero in a stable, well damped and rapidly responding manner;
- Establish an equilibrant driving conditions; and
- Maintain the establish path in the presence of disturbance such as gusts crosswinds, roadway fluctuations, etc.

The driver models and technique for experimental investigations were used or modified from investigations developed before, from pilot-aircraft investigation [McRuer and Klein 1974, McRuer et al 1973, and McRuer and Klein 1976]. As the results were defined in terms of the steering characteristics, parameters of visibility and road [Weir 1970, and Allen] allowed integration of the system driver-vehicle-road by the best way.

Because of very slow processes taking place in helmsman-ship system and limited number of manual control task this system was investigated in several researches [Veldhuyzen and Stassen, and Veldhuyzen].

The model takes into account pilot ability to predict ship response and consists of two blocks-internal models of ship dynamics and decision making element (Figure 2.4).

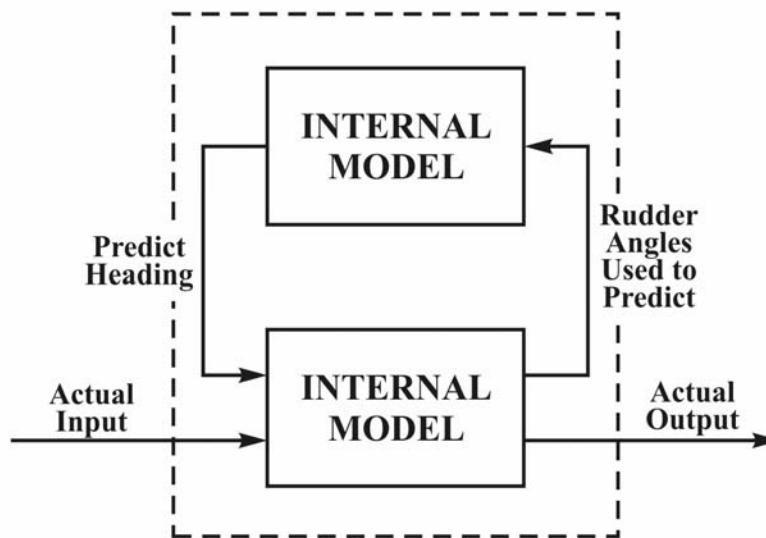


Figure 2.4: The Internal Helmsman Model.

In spite of wide use of considered above pilot behavior mathematical models all of them demonstrate different level of disagreement with experimental results. It is noticeable in low frequency range and in crossover frequency range too. The last peculiarity takes place for cases when controlled element dynamics has a considerable time delay or zero slop of amplitude ratio $|W_c(j\omega)|$ in the crossover frequency range. Therefore there is a necessity to find the new technologies for the mathematical modeling. One of them might be the artificial neural networks including feedforward and recurrent ones. In spite of there are many papers published in this area no one was dedicated to application of neural network technique to the pilot modeling. Nevertheless our analysis and knowledge of results received in the considered area are some experience in approximation of experimental results of pilot-aircraft system investigations demonstrates high ability of this technique. The more perspective kind of it is the semi soft computing technique discussed in Chapter 3.

2.1.1 Transfer of Control to a Human Operator

In this Technical Report we are addressing varying levels of autonomy, which includes consideration of a human operator assuming control of a vehicle. This can obviously be a part of the planned mission, but we should also address unplanned events. The purpose of this discussion is to review an accident related to this topic in which wind shear was cited as a primary cause, although other factors were also important. The winds calculated from the recorded airplane flight parameters, [see Anon 1975], show that both the

BACKGROUND: EVOLUTION OF SYSTEM INTEGRATION

tailwind and crosswind were greater than 20 kts above 500 ft altitude, with relatively little variation. Between 500 ft and 200 ft there was an extreme wind shear, such that the cross-wind reduced to less than 5 kts and the tailwind decreased to zero, becoming a small headwind from 250 ft to the ground. Thus, the winds by themselves presented a complex piloting task. It can also be seen that the surface winds given by the control tower (equivalent to a 4 kt headwind and a 2 kt crosswind) give absolutely no indication of potential problems. This is what the operator (the pilot in this case) would know ahead of time.

The final approach was made with autopilot coupled and auto throttle engaged, i.e. can be considered as autonomous operation for the purposes of this discussion. Because of conditions peculiar to the airfield the autopilot was disengaged at 184 ft altitude with the runway partially in sight, and a manual landing was attempted. Above 500 ft the automatic flight control system (AFCS) established off-nominal trim conditions of a higher rate of descent, reduced thrust and reduced pitch attitude in order to maintain the glideslope. As the airplane descended through approximately 500 ft the tailwind and crosswind began to decrease. With a decrease in tailwind, the momentum of the aircraft caused an initial increase in airspeed and consequent rise above the initial glideslope. This is discussed in the reference, but not pointed out, is that without any control input the aircraft would decelerate to approximately the original airspeed and descend below the original glideslope. The AFCS responded to the initial perturbation, however, by reducing thrust and decreasing pitch attitude, i.e., the opposite of the long-term corrections required. As the aircraft started to descend below the nominal glideslope, the AFCS would normally start to reverse the previous inputs and reacquire the glideslope. A further decrease in tailwind, however, would tend to produce another transient increase in airspeed and rise, causing further reduction in thrust and pitch attitude. Since the winds for the accident showed a continuous wind shear down to 200 ft altitude, it is probable that the AFCS was continually correcting the “initial transient” by reducing thrust and pitch attitude until the point at which it was disengaged. Also starting at about 600 ft altitude the left cross wind began to decrease, causing the aircraft to move left of the localizer. Although the autopilot input appropriate corrective control commands, the aircraft was still left of the localizer (but close to the glideslope) when the autopilot was disconnected. Thus, we consider that the autonomous system was in a continual dynamic control mode.

Now, we consider the effect of handing off the control to a human operator, in this case a pilot actually in the vehicle. It has been said that “we need autonomous systems to operate with human efficiency”. It is also true, however, that human operators can be disoriented. With the available visual cues the pilot judged his primary task to be aligning with the runway. At this point, unfortunately, we have already pointed out that the system was in a dynamic mode and he would have been given the ground conditions as only light winds. The following list of events Table 2.1, from the discussion in the reference, illustrates the conditions at various points including autopilot disconnect which occurred twelve seconds before ‘touchdown’. Engine rpm and pitch attitude are shown to allow comparison with nominal no wind conditions.

Table 2.1: Thrust and Pitch Attitude Before the Accident

| TIME | EVENT | ENGINE RPM % | ATTITUDE (degs) |
|----------------------------|----------------|--------------------|--------------------|
| t-15 | Middle marker | 56 | 0.9 |
| t-12 | A/P Disconnect | 54/48.5 | 0 |
| t-9 | | | begins to increase |
| t-3 | | begins to increase | |
| t | | 77 | 5.4 |
| NOMINAL NO WIND CONDITIONS | | 76.2 | 4.2 |

Because of the low-level headwind the thrust and pitch attitude required would be higher than the no-wind values. The preceding table shows that the pilot did make those necessary corrections, but pitch attitude control came three seconds after A/P disconnect and thrust increase started six seconds after disconnect, i.e. too late to prevent a short landing.

It may be of interest to consider the vehicle response to the type of airspeed perturbations under consideration. A reduction in tailwind (equivalent to an increase in headwind) is felt as an instantaneous increase in airspeed. The primary effects are increases in both lift and drag. For a conventional aircraft it is common to assume separation of the longitudinal dynamic response into a well-damped short period mode (variations of pitch attitude and angle of attack at constant airspeed) and a lightly-damped phugoid mode (variations in airspeed and pitch attitude at constant angle of attack). The response to an airspeed perturbation is dominated by the phugoid characteristics since it is primarily this mode that is excited, and any small angle of attack perturbations are quickly damped out.

Now, if the pitch attitude perturbations are suppressed the classical modes become an approximate “angle of attack” mode and an approximate “airspeed” mode. In this example the phugoid mode becomes aperiodic, see Moorhouse 1977. Even with a reduced time constant the aperiodic mode does not give the pilot any appearance of airspeed stability, whereas the oscillatory mode shows a significant reduction in the airspeed perturbation within a few seconds. It may be surmised that the aperiodic mode would induce over control, i.e., larger power changes than necessary to correct the airspeed transient. Thus, in a dynamic control mode the response would appear significantly different from what the operator may be expecting.

The above is a simple example of an unplanned event. It would have been possible, however, to provide a warning to the pilot of the overall nature of the wind shear to be encountered on a landing. This would be done by means of a computational procedure which would give the nominal trim conditions for the glideslope being flown with the calculated values in the reported surface wind conditions.

This discussion concerned one particular accident in which a primary factor was the occurrence of large wind shears. It has been shown that airspeed response to wind shear is adversely affected by tight control of pitch attitude to maintain glideslope. It is only an example of how to address unplanned events from degrading the hand over of an autonomous system to a human operator, if that operator is assuming complete control of the flight path. Technically, it seems straightforward to sense if the autonomous system is in a dynamic control mode. This would be deviations from the planned flight settings due to winds, threat avoidance, etc. It would be possible to compare this activity with the conditions assumed by the operator, and provide the advance warning/information to assist the operator by avoiding total surprise.

2.2 CASE STUDIES

2.2.1 Land

2.2.1.1 Background

The complexity associated with land environments has imposed unique demands on unmanned ground vehicles (UGV). Unlike its air (UAV) and sea (AUV) cousins, a UGV cannot assume its environment is passive and largely obstacle free. A UGV must actively sense its environment, create a representation of its world, and analyze its internal representation for obstacles before it can safely traverse. Formally, this is known as the sense, model, plan and act paradigm (SMPA) [Brooks 1991]. Of these four steps, the modelling and planning steps are considered the most difficult, especially when faced with unstructured environments. Given that unstructured environments are intrinsically unpredictable, early deliberative SMPA focused on structured environments that simplified the modelling and planning tasks [Nilsson]. This early research revealed that the monolithic, deliberative SMPA approach was brittle, cumbersome and slow. Subsequently, robotics diverged towards reactive systems, which used the world

BACKGROUND: EVOLUTION OF SYSTEM INTEGRATION

itself as the model. Using the world as the model greatly reduced the reliance on modelling and planning, and produced high performance systems that were capable of robust performance in high complexity, unstructured environments [Brooks 1989a, Brooks 1989b, Connell and Arkin]. Although reactive robots exhibited interesting, insect like characteristics, their lack of predictability yielded few useful applications. Most current UGVs, using a network of complementary control threads, now exploit a hybrid implementation that is composed of both deliberative and reactive control strategies. These hybrid architectures are pragmatic systems involving multiple computing elements of varying scales, protocols and capabilities that are networked into a functional, engineered system. Numerous hybrid architectures have been proposed and developed and the following is a partial list of the best known architectures:

- 1) The Task Control Architecture (TCA), shown in Figure 2.5, was one of the first architectures that united both the deliberative and reactive approaches [Simmons].
- 2) The Distributed Architecture for Mobile Navigation (DAMN) uses an arbiter as a means of integrating deliberative planning with reactive control [Rosenblatt]. The DAMN architecture is shown in Figure 2.6.
- 3) The Three-tiered Architecture developed by Bonasso [Bonasso et al].

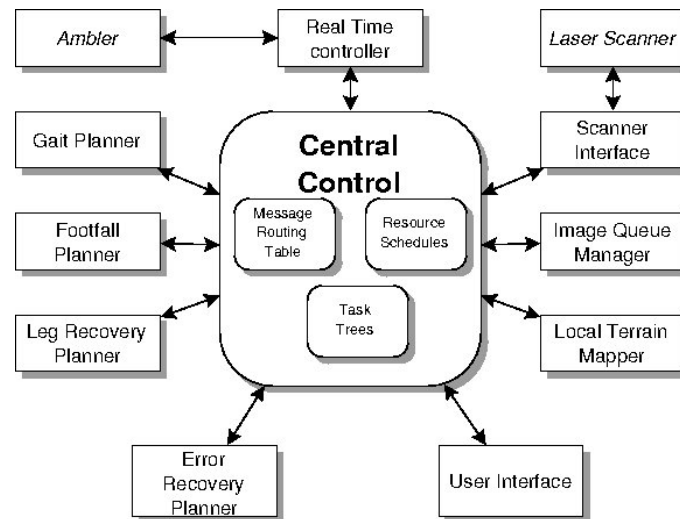


Figure 2.5: Carnegie Mellon's Task Control Architecture for the Ambler Hexapod.

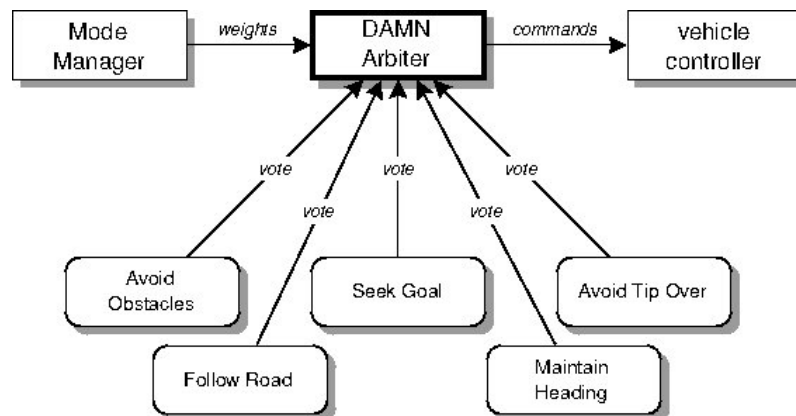


Figure 2.6: Carnegie Mellon's Distributed Architecture for Mobile Navigation for the NAVLAB.

All hybrid architectures attempt to interleave the hierarchical, deliberative paradigm and its sluggish response times, with the quick reactive behaviour based paradigm. This approach has yielded successful UGV implementations [Thrun et al 1998 and Brock et al], and is the dominant approach for the current generation of UGVs.

2.2.1.2 UGV Case Studies

2.2.1.2.1 Mars Rovers

Given Mars' orbital location and its distance from the Earth, it is impossible to drive Mars rovers in the traditional tele-operated manner. All Mars rovers exhibit a degree of autonomy, though their level of autonomy may be best described as semi-autonomous. The Pathfinder rover had a very limited degree of autonomy. Earth based engineers used the Rover Control Workstation [Cooper] to craft a detailed set of commands that described the sequence of operations to be performed by the rover. The Pathfinder rover was restricted to executing the command sequences as they were received. Most faults or anomalous situations required the rover to halt all activity and wait for the ground operations team to diagnose the problem and uplink a recovery plan [Washington et al].

The second generation of Mars exploration rovers (MER), called Spirit and Opportunity, are considerable more advanced than the first generation Pathfinder rover. These rovers include autonomous navigation (AutoNav) and visual odometry (VisOdom) that give the MERs considerable autonomous capabilities. Visual odometry significantly improves the MER's position estimation, as it is much more accurate than a position estimate that is basely solely upon wheel odometry. Under AutoNav, the MER, using its stereo vision cameras, acquires 3-D range data, uses the range data to construct a terrain map, analyses the terrain map for both positive and negative obstacles, and plans a route to avoid the detected obstacles [Biesiadecki]. Unfortunately, both MERs are computationally bound, and using its full autonomous capabilities seriously degrades the speed at which it can drive. Using directed drive commands a MER can cover distances of up to 124 meter/hr, AutoNav reduces the drive distance to 10 to 36 meter/hr, while AutoNav used in conjunction with VisOdom drops the traverse rate to 6 meter/hr. A MER, undergoing testing at JPL's Spacecraft Assembly Facility, is shown in Figure 2.7.

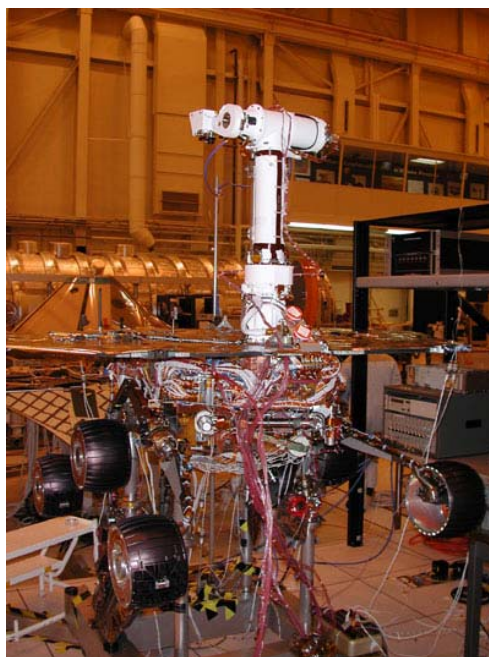


Figure 2.7: The Mars Exploration Rover, Courtesy NASA/JPL-Caltech.

2.2.1.2.2 *DARPA Grand Challenge*

In stark contrast to the very conservative autonomy implementation used on NASA's Mars rovers, the DARPA Grand Challenge demanded that autonomous ground vehicles (UGV) implement aggressive strategies. The DARPA Grand Challenge, initiated in July, 2002, challenged researchers to conceive and build autonomous vehicles that could traverse 140 miles (227 km) of outdoor terrain within a 10 hr time limit. The outdoor terrain featured difficult desert roads, global positioning drop-outs, sharp turns, narrow openings, bridges, railroad overpasses, long tunnels, and obstacles. DARPA offered a \$1,000,000 US prize to the team that completed the course in the shortest period of time. The teams travelled to Barstow, CA to compete in DARPA's first grand challenge. Of the 106 competitors, only 15 vehicles completed or partially completed the qualification round and competed in the actual race. The fact that none of the 15 finalists finished the course attests to the challenges associated with traversing outdoor terrain. Sandstorm, from Carnegie Mellon University, was the most successful entry and it only traversed 7.4 miles before immobilizing itself on a berm.

Traversing unstructured terrain requires the implementation of the full SMPA cycle, where sensors acquire data, a world representation is created and updated, planning is performed, and commands are issued. This process must be implemented in real-time as the vehicles are travelling at speeds near 20 mph (32 km/hr). To operate at such speeds, the vehicle must respond quickly to new inputs while maintaining a global strategy that allows it to pursue long-range goals.

The second DARPA Grand Challenge, held in Oct 2005, was a 132 mile (214 km) unpaved course near Primm, NV. One hundred and ninety-five teams applied to compete for this race's \$2,000,000 US prize and only 23 teams qualified. Using the lessons learned from the previous race, 5 teams devised UGVs that successfully completed the course. Of these 5 UGVs, the winner, Stanley from Stanford University, and the runner up, Sandstorm from C.M.U., illustrate the two distinctive philosophies of UGV implementations.

Researchers with a wealth of experience (Sandstorm competed in the first Grand Challenge) devised Sandstorm and, using this experience, they focused on simplifying the overall robotic system. This was accomplished via a balanced approach using mechanical and software solutions that leveraged existing technologies [Urmsen et al]. On the mechanical side, Sandstorm, shown in Figure 2.8, was derived from the HMMWV platform, which has a much larger ground clearance than a typical sport utility vehicle. Consequently, the onboard navigation software is less sensitive to terrain features as the vehicle can intrinsically surmount relatively large obstacles (Team TerraMax took this approach to an extreme by using the massive Oshkosh MTRV MK23 truck platform). From the software perspective, human input was used extensively at the preplanning stage, thus also reducing the complexity of the onboard navigation system. Grand Challenge competitors were given the course route, as a set of GPS waypoints, 2 hrs before the race start. During this 2 hr run up, human expertise was leveraged to supplement the route data and to account for dangerous terrain that might be difficult for the UGV to detect in real time. Hence, during a majority of the race, Sandstorm followed a predefined set of waypoints, at a predetermined velocity. Under this regime, the SMPA cycle is reduced to behavior that executes only under extenuating circumstances, such as the detection of an obstacle or hazard. In summary, Sandstorm can be viewed as a system with guarded autonomy. While embracing significantly more autonomy than delivered by the NASA Mars rovers, Sandstorm still relied heavily upon human expertise to choose its initial route.



Figure 2.8: C.M.U.s Sandstorm Unmanned Ground Vehicle.

Stanley, the robot that won DARPA’s second Grand Challenge, represents the second philosophy for UGV development. The Stanford Racing Team treated “autonomous navigation as a software problem”, hence the robot’s software relied predominantly on state-of-the-art artificial intelligence technologies such as machine learning and probabilistic reasoning [Thrun et al 2006]. Whereas Sandstorm relied on vehicle mobility and human preplanning, Stanley focused on software and autonomous capabilities. Following this line of reasoning, Stanley was built upon a modified Volkswagen Touareg R5 SUV, which remained in a street legal condition. Figure 2.9 shows the Stanley UGV.



Figure 2.9: Stanford’s Stanley Unmanned Ground Vehicle.

Both Stanley and Sandstorm devised real-time, hybrid, SMPA implementations that sensed and modelled the world, planned within the world model, and acted upon the chosen plan. Stanley’s SMPA implementation was significantly more complex than Sandstorm’s, as Stanley incorporated probabilistic reasoning that allowed for more reliable world representations, and machine learning allowed Stanley to learn from experience. The Grand Challenge racecourse included diverse terrain whose appearance changed with both locale and with time of day due to lighting conditions. Stanley’s ability to learn while driving allowed it to quickly adapt to these changing conditions and this ability was a key factor in Stanley’s victory.

2.2.1.2.3 Unmanned Ground Vehicles at DRDC – Suffield

DRDC – Suffield started its UGV program in the late 1980's with the development of tele-operated ground vehicles. Since its inception, DRDC's tele-operation program has developed numerous tele-operated vehicles, with various capabilities that were suited to differing applications. This vehicle stable includes: a DH6 Caterpillar, a Bobcat with a backhoe, a 6 wheel drive skid steer vehicle, and a number of 4 wheel drive vehicles. Figure 2.10 shows a recent addition to DRDC's tele-operated vehicle stable.



Figure 2.10: DRDC's Tele-Operated Multi-Agent Tactical Vehicle.

Beginning in the late 1990's, DRDC shifted its focus from pure tele-operation to autonomous vehicles. The DRDC Raptor UGV, shown in Figure 2.11, is capable of autonomous operation in unstructured, outdoor terrain. The Raptor UGV implements a real-time, hybrid SMPA cycle that allows it to:

- Sense terrain using nodding laser rangefinders [Brotten et al 2006] and stereo vision;
- Update the internal world representation [Brotten et al 2007];
- Determine the best obstacle free path [Giesbrecht] that progresses the Raptor towards its goal [Mackay]; and
- Send steering and velocity commands that drive the vehicle.



Figure 2.11: DRDC's Raptor UGV.

The Raptor UGV is a system of systems that, when operating together, allows the vehicle to operate autonomously without the need for human intervention. Figure 2.12 illustrates the individual systems that comprise the Raptor UGV.

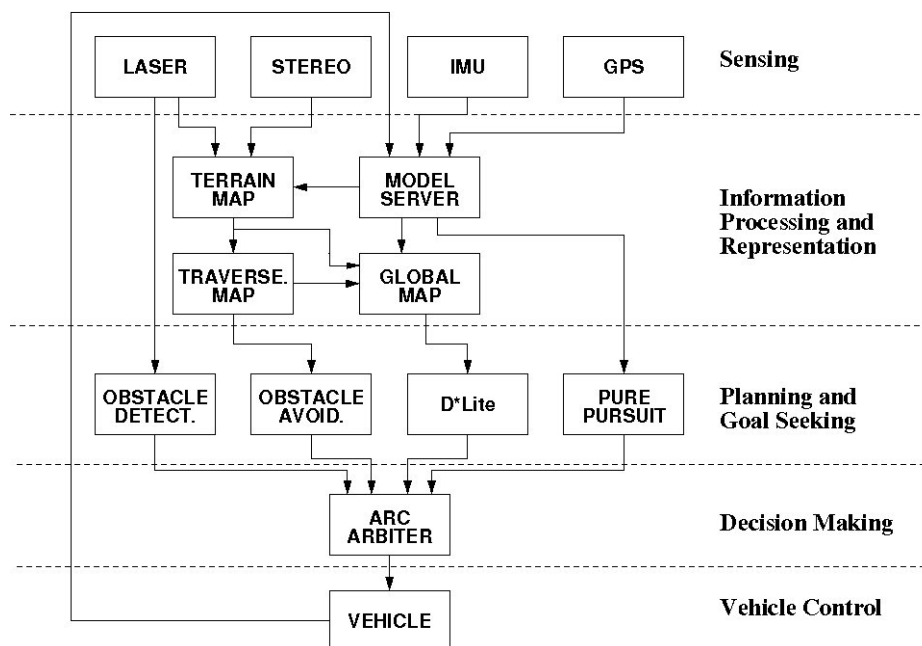


Figure 2.12: Raptor UGV Systems.

Each individual system is a separate entity that exports defined interfaces and is capable of acquiring data from other system interfaces. This modular approach, based upon components, is achieved using a software framework [Brotten et al 2006]. DRDC's systems approach allows researchers to easily modify existing components and to add new components into the existing structure. The Raptor UGV's autonomous capabilities have been exercised on numerous field trials.

2.2.2 Sea (DH)

Today there are many examples of autonomous marine vehicles that employ various levels of autonomy. The most numerous of this class of vehicle is usually referred to as autonomous underwater vehicles (AUVs), or sometimes unmanned underwater vehicles (UUVs), and can vary in size from something weighing less than 50 kg to vehicles with close to 9000 kg displacements. The vehicles can be designed for shallow water depths of 10 m or less, or deep water depths of over 6000 m. Similarly, their mission endurance can vary dramatically – from as little as several hours to a week or more. Figure 2.13 shows two vehicles that demonstrate this dramatic range of size and capability. The Remus vehicle, developed by Hydroid, is an example of a more compact AUV that is used in many application areas, ranging from general scientific sampling and mapping to more focused mine countermeasures. The Theseus vehicle, built by International Submarine Engineering for Defence Research and Development Canada, is an example of a vehicle that was developed for one specific mission – to lay 220 km of fiber optic cable under the Arctic ice.

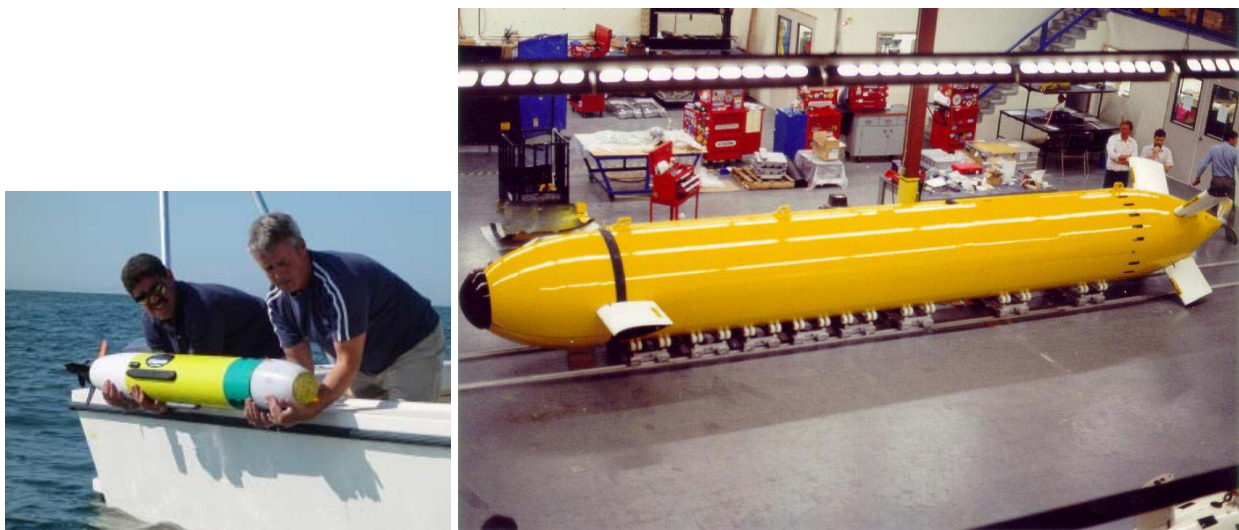


Figure 2.13: Two Examples of Autonomous Underwater Vehicles. The smaller vehicle Remus (left) is a compact, general purpose vehicle. The larger vehicle, Theseus (right) is designed for a specific mission.

In addition to AUVs, there is a second class of autonomous marine vehicles, known as unmanned surface vehicles (USVs). In general, USVs are built for very specific mission applications. Figure 2.14 shows two examples of USVs, the multi-mission Spartan vehicle (left) and the Remote Minehunting vehicle Dorado (right).



Figure 2.14: The Multi-Mission Spartan Vehicle (left) and the Remote Minehunting Vehicle Dorado (right) are Two Examples of Unmanned Surface Vehicles.

The Spartan is a modular vehicle built around a 7 m rigid inflatable vehicle (RIB). The payloads vary from a variety of sensor packages for minehunting and submarine detection to a variety of weapons payloads. The Dorado is a semi-submersible based vehicle that was initially designed to provide a stable platform for hydrographic surveys in high sea states. More recently, the third generation of the Dorado vehicle has since been developed specifically for remote minehunting missions.

Independent of size and mission, all of these unmanned systems have one very important thing in common – through varying degrees of autonomy, they have removed the human operator from a hostile operating environment, enabling missions that would be otherwise impossible to complete. In order to explore the degree of autonomy employed in each of the systems, and the impact this has on overall system complexity and mission reliability, the Dorado remote minehunting system and the Theseus AUV will be examined in more detail.

2.2.2.1 Theseus AUV

From 1992 to 1996, International Submarine Engineering Research and the Esquimalt Defence Research Detachment of Defence Research Establishment Atlantic worked together to develop a large autonomous underwater vehicle for laying fibre-optic cables in ice-covered waters. The vehicle, named Theseus, was designed to lay up to 220 km of fibre-optic cable. The water depth along the cable route varies from 50 m at the launch site to between 500 and 700 m at the array site.

Both the environment and the complexity of the mission imposed severe constraints on the vehicle design. In the operating area the ocean is completely ice covered, mostly by multi-year ice, 3.5 to 10 m thick, with ice keels that can extend to depths of 30 m within 10 km from the launch site, and 50 m further out; water currents vary from 0 cm/sec up to 50 cm/sec near the launch site, and up to approximately 10 cm/sec at the array site; air temperatures vary from -40 to -20° C during the only possible deployment period (late March to early May); and water temperatures vary from -2° C just under the ice to $+4^{\circ}$ C near the bottom at a depth of 600 m.

BACKGROUND: EVOLUTION OF SYSTEM INTEGRATION

To lay the cable, return to the launch site and allow a safety margin required a 450 km endurance and a 220 km cable capacity. The system required a navigational accuracy within 1% of distance travelled, and needed a terminal homing system for the final run-in to the array site. To minimize the amount of cable in the water column, the AUV was required to follow the bottom at an altitude of 20 to 50 m. To facilitate air transport to the launch site, a modular construction was required, with each section weighing under 1400 kg.

In addition, it was determined that an obstacle-avoidance sonar (OAS) system would be required to ensure that the vehicle would not crash into uncharted bottom features or into ice keels. Acoustic telemetry was also considered essential for occasional enroute communication with the vehicle. The vehicle needed a precise terminal guidance system to facilitate cable recovery. A provision was included to allow the vehicle to update its position at acoustic beacons located along the route and also at the cable delivery site. Details on the resulting vehicle are provided in the next section.

A cross-section of the Theseus AUV is shown in Figure 2.15. The principal characteristics of the vehicle are listed in Table 2.2.

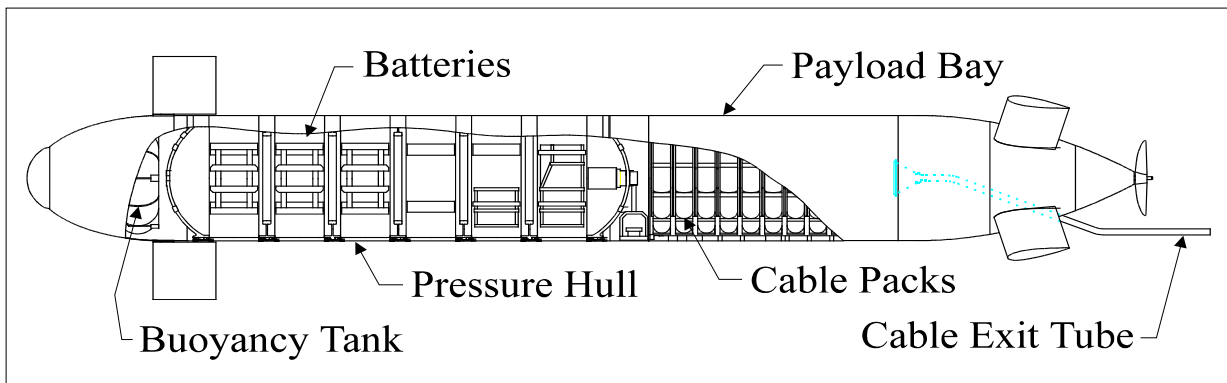


Figure 2.15: Theseus Schematic (Foreplanes, Exit Tube in Plan View, Remainder in Elevation View).

Table 2.2: Theseus Vehicle Characteristics

| | |
|--------------------------------|---|
| Length | 10.7 m (35 feet) |
| Diameter | 127 cm (50 inches) |
| Displacement | 8600 kg (19,000 lbs) |
| Speed | 2 m/s (4 knots) |
| Range | 700 km (380 nm) |
| Maximum Operating Depth | 425 m verified, 1000-m (3280-foot) design depth |
| Cable Capacity | 220 km |
| Navigational Accuracy | Achieved ~0.5% of distance travelled |
| Propulsion | 6 hp brushless dc motor and gearbox / single 61 cm diameter propeller |
| Power | 360kWh Silver Zinc battery pack consisting of 280 individual cells manufactured by Yardney. 450 km mission plus an additional 24 hours of hotel load with a safety factor of two. |
| Variable Ballast | ±95 kg (250 lbs) in each of 2 toroidal tanks, 1 fore and 1 aft |
| Controller | Proprietary real-time kernel running on MC68030 microprocessor |
| Navigation systems: Transit | Honeywell MAPS Inertial navigation unit EDO 3050 Doppler sonar (bottom tracking) |
| Terminal Homing | Datasonics ACU-206 acoustic homing system. Ranges up to 10 km in 500m-deep water. |
| Acoustic Telemetry | Datasonics Model ATM851 using Multiple Frequency Shift Keying (MFSK) plus error encoding operating in the 15 to 20 kHz band. |
| Fibre Optic Telemetry | Used on outbound leg of mission for vehicle status. Allows operator to assume control |
| Emergency Beacons | ORE 6702 acoustic transponder located in the tail section. Interrogated with ORE LXT ultrashort-base-line acoustic tracking system operating at 11kHz. |
| Obstacle Avoidance | Sonatech STA-013-1 forward-looking sonar. 5 by 4 beams. |
| Pressure Hull | 5 cm-thick Aluminum (7075), 4.5 m by 127 cm diameter in 5 sections plus end domes. Design depth 1000 m. |
| Payload Bay | Free-flooding fiberglass shell with syntactic foam lining, top half removable. Inner diam 114 cm, length 228 cm. Payload up to 1960 kg dry, 320 kg in water. |
| Current Payload | 11 packs of 20 km cable, each weighing 60 kg in water. 11 toroidal compensation tanks fill as cable paid out. Tank inner diam 76 cm (30 in). |
| Transportability | Modular construction in sections under 1400 kg each. |

In order to increase the fault tolerance, Theseus manages fault responses using a pre-defined fault table. This table allows the user to divide a mission into any number of phases, where a phase consists of one or more manoeuvres between waypoints. Each phase of a mission script has its own set of responses to each of the vehicle faults: a response is either stop up under the ice, stop down to the sea bed, change to another

BACKGROUND: EVOLUTION OF SYSTEM INTEGRATION

mission step, or ignore the fault. Therefore, a change of phase occurs when the desired response to some fault changes, such as when approaching a manned camp. At this point a new set of entries in the fault table takes effect. It was decided that 18 phases adequately provided for the changing circumstances during the Arctic mission.

Designing a navigation system to allow an AUV to navigate autonomously under-ice for more than 400 km was a challenge. The presence of a permanent ice cover requires that all sensors used to determine position had to be located below the ice cover but not necessarily on board the vehicle. The chosen solution for navigation was to use an onboard, medium-accuracy positioning system for outbound/inbound transits, and an external, but subsurface, terminal-guidance acoustic positioning system for cable delivery and vehicle recovery.

Theseus monitors its position by dead reckoning. It uses a Honeywell medium-accuracy inertial navigation unit (INU) and a Doppler sonar. The INU provides heading and attitude data, while the Doppler sonar measures forward and lateral ground speeds, as well as altitude above the seafloor. This combination provides positions with an error of approximately 0.5% of the distance travelled, well within the 1% design goal.

The cable is stored on a series of spools which are stacked longitudinally along the vehicle axis. Adjacent spools are spliced together prior to launch. The cable and splices wind off the spools from the inside-out, and exit through a tube in the stern. The tension on the cable (to keep it from free-spooling) is maintained through the use of a special glue applied to the cable during the spooling process. To keep the system simple and reliable, no active tensioning or dispensing devices are used.

As the cable leaves the vehicle, weight is lost. To prevent this from affecting vehicle trim, the loss in cable weight is counteracted by an automatic buoyancy compensation system. Surrounding each cable spool is a toroidal hard ballast tank which is filled with water as the cable is dispensed from its companion spool. This keeps the net buoyancy of each spool/tank assembly near neutral.

2.2.3 Air and Space

2.2.3.1 F-22A Flying Qualities Development

The development and integration of the flight control system on the F-22A is cited as an example of how the Systems Engineering process can be applied properly. The specifics are discussed in RTO Report 29 and in Harris. One of the hallmarks of this “Design for Flying Qualities” Process is that one of the flight dynamics simulations of the air vehicle is identified as the “truth model” of the air vehicle. This truth model is “pedigreed”; that is, each component and its integration into the whole is identified, an accurate simulation model of that component and its interfaces is developed, and that component model is shown to be traceable to “reality” (system design and test data and the physics associated with the component). In addition, positive and negative margins and tolerances in how the system will perform are identified and tracked, as well as limitations of the component model and the effect of those limitations on the total air vehicle model. Once identified, the entire flight dynamics simulation, including all of its components, is placed under strict configuration control, and a disciplined process for updating component models is implemented, allowing for the systematic updating of the simulation as new specifications, component test data, or air vehicle flight test data become available. As implemented, the F-22A “Design for Flying Qualities” Process was, in fact, a continual iterative system engineering process applied to the development of the flying qualities of the aircraft, and it explicitly addresses steps 3 – 6 of the Systems Engineering process (see Chapter 3).

As implemented, the simulation should be as accurate as possible, rather than being conservative at each interface and in each subsystem model. The result of such conservatism is the simulation equivalent of a

“negative tolerance stackup”. One of the authors of this report was associated with a flight development program where the actual aircraft displayed performance much better than the flight dynamics simulation indicated. While pleasantly surprising, the simulation model had in fact failed to accurately predict the behaviour of the flight vehicle. In examining the simulation to determine why this had happened it was found that such conservative tolerance stackup had occurred.

This is not to say that component performance bounds should not be tracked. When combined, the negative and positive tolerance stickups of predicted performance will form the bounds of sensitivity analyses. When these sensitivity analyses are performed, not only should the question of “what happens if everything is as bad as it could be and still within specification bounds” be examined, but also the question of “what happens if everything is as good as it can be” should also be examined.

One of the report authors was acting as consultant after one ‘major visible system problem’. The working level unanimously said there was a problem with inter-group communication and no integration. The first-level manager said that was not true. This is a red flag, when there is consensus on a working problem {not one disgruntled employee} but the manager denies it, then INVESTIGATE. In contrast, the cooperation and smooth team functioning exhibited by the F-22A Flying Qualities Working Group (FQWG) was cited by an independent evaluator as one of the major contributors to the success of the F-22A in achieving excellent flying qualities. Indeed, the FQWG achieved such a standard of excellence and cooperation that on more than one occasion simply saying, “the FQWG recommends that...” carried sufficient authority and credibility to obtain management cooperation or approval.

The Design for Flying Qualities process has subsequently been adopted as a Best Practice by development teams at the USAF’s Aeronautical systems Center and at the USN’s Naval Air Systems Command.

2.2.3.2 Apollo and the Space Shuttle

The United States’ Apollo spacecraft system (consisting of the Saturn launch vehicles, the Command, Service, and Lunar Modules, the Skylab space station (itself consisting of several modules), and the Apollo-Soyuz Docking Module) was designed to a system Probability of Loss of Aircraft (PLOA) of 1/1000 per flight, but was implemented with a high level of redundancy (including the “dissimilar redundancy” of using the Lunar Module as a “lifeboat” for a disabled Command/Service Module, evaluated in flight on Apollo 9 and used on Apollo 13). In fact, only for three mission phases (the Lunar Module’s ascent from the moon into initial parking orbit, the trans-earth injection to return to the earth, and the Command Module’s re-entry) did Apollo flight crews not have a hardware or procedural backup of some form available; in those cases the components required to function correctly for a successful mission were designed and tested to very high standards, and when possible the number of individual components without similar backups (specifically the combustion chamber/nozzle assemblies for the Lunar Module ascent stage and the Service Module) were minimized.

Flight Director Gene Kranz [Kranz] also credits a philosophical approach of treating any active combination of modules and launch vehicles as a single integrated entity as a major factor contributing to the success of Apollo. In this sense, Apollo’s managers and flight controllers presaged a “system of systems” approach. Another factor in Apollo’s success was an extensive test program, beginning with components and progressing through subassemblies to full system, full-scale tests to ensure the performance of the hardware was well understood and within (or better than) specifications. Finally, every Apollo flight (particularly after the Apollo 1 on-pad cabin fire) was philosophically approached as a high-risk test flight. The result was no losses of vehicles or crews in 6761.49 system manned flight hours and only one loss of mission (Apollo 13) with successful recovery of the flight crew. Given the design PLOA of 1/1000 per flight and an average flight duration of 450.76 hours for the 15 manned Apollo flights (including 3 long-duration Skylab flights), the equivalent design PLOA is actually 2.22e-6 per hour.

BACKGROUND: EVOLUTION OF SYSTEM INTEGRATION

NASA's ambition for the next spaceflight program after Apollo – the Space Transportation System, or Space Shuttle – was a vehicle which could be operated much more like a production public-use aircraft. Part of this was a desire to change the whole philosophical approach to manned space flight, while part (in retrospect, a significant part) was driven by greatly-reduced appropriations in the 1970s. The result was an attempt to apply design, test, and operating procedures and paradigms appropriate to a mature, production system operating in a well-defined and understood environment to what is arguably the most complex flight vehicle ever built. Even prior to its first flight in 1981, attempts were made to reduce costs by reducing component and subassembly testing. During development, multiple catastrophic failures of components of the Space Shuttle Main Engines (SSMEs) absorbed the projected cost and schedule savings and then some; other components proved equally troublesome.

NASA's travails in operating the Space Shuttle have been well documented (See, for example, the Columbia Accident Investigation Board Report, [Anon 2003b], and Mike Mullane's *Riding Rockets*, [Mullane]). From a design standpoint, the Shuttle system was meant to have reliability sufficiently high to allow this, which would require the PLOA to be of the same order of magnitude as that seen in airline operations, which ranged from $9.1e-7$ to $5.6e-8$ per flight hour during roughly the same time period (specifically 1983 – 2002 for United States commercial (14CFR121) operations). Instead, the Shuttle has suffered two well-publicised fatal losses in 25,076.64 flight hours (as of 2005), resulting in a demonstrated PLOA of $7.98e-5$ losses per flight hour (roughly $1e-4$). This is a rate much more analogous to that of an immature system in the flight test environment, backing the assertion of the flight crews and the Challenger and Columbia accident boards that the Space Shuttle is an experimental vehicle. Overemphasis on meeting schedules and the “normalization of deviance” (where undesired, unexpected or unexplained behaviour of a component of the system is progressively accepted as normal) have been well-documented as underlying causes of both Shuttle losses, [Anon 2003b, Deal, Feynman, and Vaughn]. Less immediately obvious is that given the underlying failures causing the losses of Challenger and Columbia did not originate with the Shuttle vehicles themselves, the question arises as to whether NASA abandoned the Apollo-era philosophy of treating any combination of “modules” as a single integrated system.

2.2.3.3 The Development of Fly-By-Wire Integration

The aircraft effectiveness and flight safety were always the main criteria in aircraft design. In particular it might be done by improvement of flying qualities and flight performances by different means including flight control system design (FCS). To the end sixties of the last century it was obvious that the shortcomings of mechanical linkage (increased weight, nonlinear effects, difficulties in realization of advanced control laws etc.) did not allow to realize new ideas. The new innovations were required to provide the new principles. Such new principles in improvement of flight performances are the following:

- Decrease of stability margin and use of unstable configuration even.
- Super maneuverability and high L/D ratio by use of specific aerodynamic and additional control surfaces.
- Integration of different aircraft systems (flight control and propulsive, for example).
- New principles in flying qualities, which were necessary to provide:
 - Optimization of flying qualities by use new algorithms adopted to the piloting tasks and pilot-aircraft system goals; and
 - Integration of flight control system and display.
- The problems in flight safety were solved by:
 - Development of automatic critical regimes warning and barrier system;
 - Creation of means for reduction of conflict between pilot actions and limited potentialities of flight control system;

- Redundancy of control surfaces and elements; and
- Reconfiguration of control surfaces and control laws.

Fly-by-wire (FBW) technology was one of the innovations allowed to solve these principles. It was developed to the end of sixties and realized in aircraft development in seventies.

The main features of FBW FCS are:

- Electrical linkage between pilot and actuators;
- Electrohydraulic actuators;
- Computers;
- Advanced control laws;
- Enlargement of FCS functions; and
- New manipulators.

The FBW technology allowed designers to integrate all mentioned above principles in improvement of flight performances, flying qualities and flight safety. The technology is based on achievements in design of flight control systems of previous generation. Some of them were the followings:

- Hydraulic actuators (The first experimental hydraulic mechanical actuator was developed and tested in Russia in 1949 [Bushgens et al 2001]. A MIG-15 was used as a test-bed for that purpose;
- Innovations in improvement of flight performances: e.g. location control system for reduction of stability margin (aircraft M-3 and M-50, Russia [G. Bushgens et al 2001]; new additional control surfaces located at the fuselage nose (Tupolev Tu-144, Russia [Bushgens et al 2001, Bushgens, 1990]);
- Redundancy of the systems (two hydraulic channels were used on MIG-19, and four – on Tu-144);
- Sectioning of control surfaces (eight sections of Tu-144 elevons); and
- Complicated algorithms for flight control systems improved considerably flying qualities (for example, RCAH (Rate Command Attitude Hold) type of system, tested in 1960 on aircraft M-3 [Bushgens et al 2001, Bushgens, et al 1979]).

All aircraft with FBW system can be divided on two generations.

The first generation of FBW aircraft are characterized by the following general features:

- Decrease of stability margin;
- Enlargement of maneuverable potentialities (fighters);
- Improvement of controllability;
- Appearance of electrohydraulic actuators;
- Advanced flight control laws provided necessary flying qualities of statically unstable aircraft, new types of dynamic responses (RCAH, ACAH (Attitude Command Attitude Hold), etc.);
- Analog computers;
- Increase of number of redundant (alternative) systems;
- New control surfaces; and

BACKGROUND: EVOLUTION OF SYSTEM INTEGRATION

- Existence of mechanical channel (as a parallel, alternative, basic for one of the channel) in many cases.

The first aircraft developed in Russia as FBW aircraft was designed at Suhoi company. The first flight of this supersonic bomber T-4 (Figure 2.16) was in 1972 [Bushgens et al 2001]. Two years later first American mass production fighter F-16 (Figure 2.17) carried out the first flight. Aircraft T-4 had a weight 100 tons, $M_{cruise}=3$, FBW system in all control channels. Its additional control surface was used for improvement of flight performances in all flight envelope. It had quadruple redundancy and reduced stability margin – 0% average ($\pm 5\%$). The first Russian FBW fighter SU-27 (Figure 2.18) was also developed at Suhoi company. It had quadruple redundancy, FBW system was in longitudinal channel [Shenfinkel]. The aircraft has small static instability (up to 5%). Its leading edge deflects as a function of angle of attack. SU-27 demonstrates the super agile potentiality (the “cobra” maneuver).



Figure 2.16: The First Fly-By-Wire Aircraft, T-4.



Figure 2.17: The First Mass Production Fly-By-Wire Aircraft, F-16.



Figure 2.18: Fly-By-Wire Aircraft, Su 27.

In development of first generation of FBW aircraft new dynamics problems were exposed. It was shown that existence of the rate limit $\dot{\delta}_{\max}$ in combination with aircraft static instability can cause the instability of aircraft – FCS dynamics when the unstable oscillations take place (so-called ‘instability in gross’). It was shown also that the existence of actuator’s nonlinearity in case of unstable aircraft can cause the stable oscillations with constant amplitude (so-called ‘instability in small’), deterioration of actuator frequency response (for small input signals).

Because of these problems there were developed a number of means for their suppression. For the suppression of unstable cycles it was design nonlinear prefilter (Figure 2.19), used now widely for FBW aircraft [Belosvet et al, Shenfinkel]. It was developed also the technique for the selection of feedback filters and actuator parameters guaranteed the stability of ‘aircraft + FCS’ system provided the specific amplitude of oscillations [Berko et al, Bushgens et al 2001, Kluev et al, Konstantinov et al 1999].

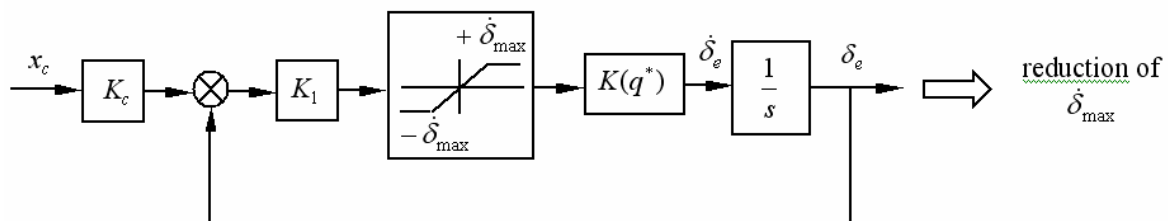


Figure 2.19: Prefilter for Fly-By-Wire System.

For the conservation of stable cycles with limited amplitude of oscillations ($\Delta n_z \leq 0.2g$ and $\Delta \theta \leq 0.1 \text{deg}$) it was developed a number of scheme, design and technological means for improvement of actuator characteristics [Bushgens et al 2001, Kluev et al].

In the frame of first generation of FBW aircraft there were created fly-by-wire systems for passenger and transport aircraft too: A-320 and later configurations (France); IL-96-300, AN-124, AN-225 (Russia).

All Russian FBW systems for civilian planes had the similar principle shown on Figure 2.20.

BACKGROUND: EVOLUTION OF SYSTEM INTEGRATION

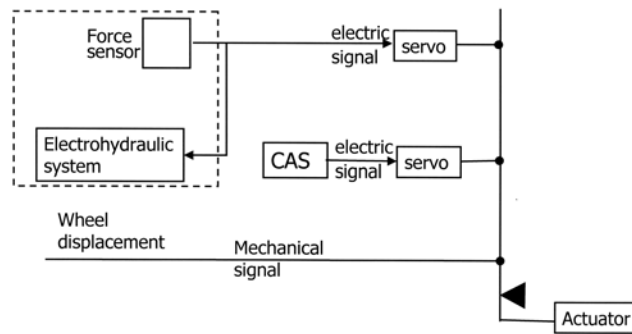


Figure 2.20: Confirmation of Fly-By-Wire and Mechanical Systems.

The electrical signal from wheel and CAS are mixed with mechanical signals constantly in all flight phases.

The second generation of FBW system is characterized by new features associated with the use of:

- Digital computers (instead of analog);
- Adaptive control laws, reconfiguration of FCS and control surfaces;
- Integration of FCS with of critical regimes warning and barrier system and FCS;
- Additional control surfaces (canard, leading edge, thrust vectoring); and
- Enlargement of control modes.

The first completely digital FBW system developed in Russia for aerospace Buran's experimental prototype –'BTC'. This vehicle was used for testing of FCS during the unpowered landing in manual and automatic control.

The FCS of modern FBW aircraft is characterized by enlargement of control modes and functioning of a number of control surfaces (see Figure 2.21).

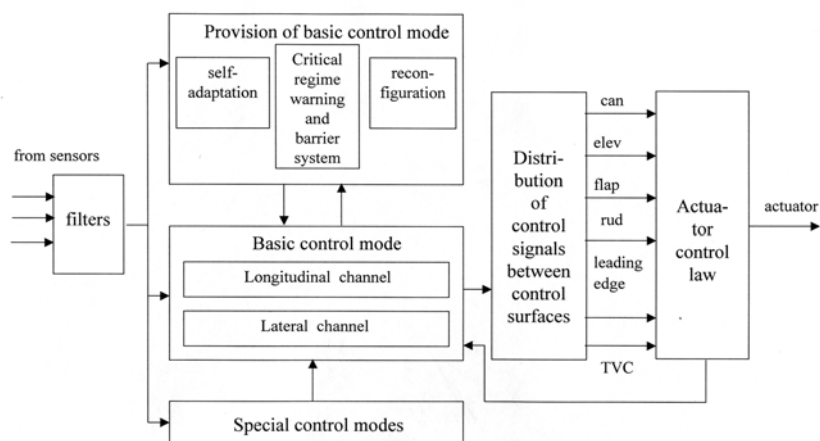


Figure 2.21: Enlargement of Control Surfaces and Modes.

For the last FBW generation of fighters there were generated the new algorithms of FCS based on principles of adaptation [Dinnikov et al 2000, Dinnikov et al 2001, Konstantinov, 2002]. It allowed to

improve the flying qualities and flight safety, to reduce rate limit ($\delta_{C_{max}}$) in 1.5 – 2 times. On the basis of RCAH system potentialities there were developed and installed the new critical regimes and barrier system Figure 2.22.

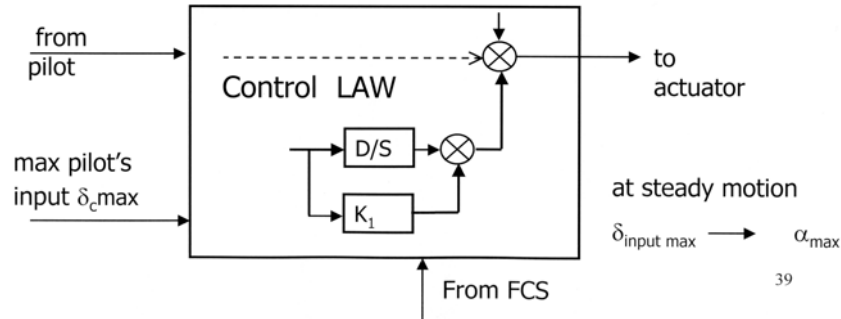


Figure 2.22: Integration of Flight Control System and Stall Warning System.

Its algorithm was integrated with FCS law. The typical FBW military aircraft have the increased numbers of control surfaces (F-22, SU-30). It leads to the necessity to distribute the control signals between the surfaces. It is realized by digital computer in optimal way as a function of piloting task and flight regime. The high and new potentialities are realized by use of thrust vectoring control (TVC). It allows to realize completely new maneuvers including “loop”, “cobra” and stable flight at all angle of attacks. Such super agility is realized by the different ideology in Suhoi and Mikoyan aircraft with TVC. As for Suhoi aircraft (SU-30 MK) each TVC is rotating along one [Lokshin et al] and as for new Mikoyan aircraft MIG-29 TVC [Obolensky] TVC rotates along two axes.

The new means for suppression of instability of statically unstable aircraft were developed. One of them is the additional control surface (canard SU-30 MK) Figure 2.23. Its integration with elevator allowed to decrease rate limit δ_{max} [Konstantinov, 2002].



Figure 2.23: Su 30 MK Aircraft.

There were developed also the new prefilters for suppression of PIO. One of its version developed for aerospace vehicle ‘Buran’ is shown on Figure 2.24 [Efremov et al, 1995].

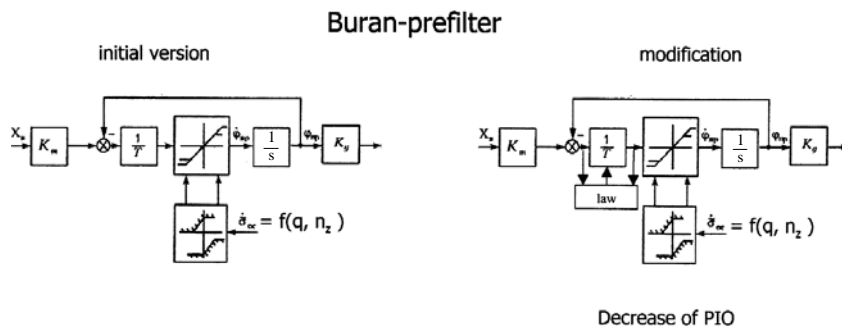


Figure 2.24: The Buran Prefilter and its Modification.

The conservation of stable oscillations with amplitudes ($\Delta n_z \leq 0.2g$ and $\Delta \theta \leq 0.1\text{deg}$) for the modern FBW fighters are reached by self-adaptation algorithm of actuators (see Figure 2.25).

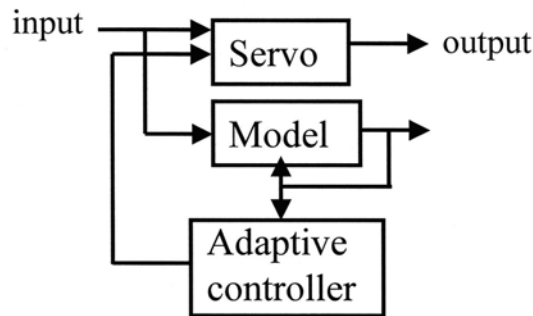


Figure 2.25: Self-Adaptive Actuator

It allows suppression of the influence of input signal on frequency response, even for very small amplitudes. Therefore it reduces the tendency to rate limit and thus improves flying qualities too.

One of the functions of all FBW system is the reconfiguration. It has to be realized when the alternative control laws (or their parameters) have to be used. The specific reconfiguration was offered to new generation of trainers (MIG-AT and YAK-130). These aircraft have the potentiality to change their flying qualities by digital FBW FCS to train pilots to control different aircraft.

The change from mechanical linkage to electrical systems caused the modification or appearance of new manipulators for FBW aircraft. There is a miniwheel for TU-204, and a side stick for the F-16 and A-320. The digital FBW technology was used in passenger aircraft designs also, for example Airbus A-340 and A-380, Boeing-777, Russian TU-204 and TU –334. All these planes are characterized by reduced stability margin. The TU-204 have analog and mechanical channels in addition to digital as alternative [Bushgens et al. 1995]. They can be switched on in case of failure of the digital system. Many modern configurations, e.g. the TU-334, have completely digital FBW system without any alternative in case of failure.

Chapter 3 – BEST PRACTICES FOR SYSTEM INTEGRATION

3.1 PROBLEMS OF SYSTEMS ENGINEERING

The Introductory chapter expresses a ‘Systems Engineering Process’ as a sequence of identifiable aspects, which are described below.

- 1) **Breaking the system down into component parts** has been done very well historically. It has produced well-defined disciplines in scientific areas such as propulsion, aerodynamics, hydrodynamics, structural design, etc. If we recall conventional aircraft design, for instance, the process was to optimize the various parts, then build multiple prototypes against a set of requirements and fly them to choose the winner.
- 2) **Understanding each individual part** might have been done with a good degree of success. Individual technical disciplines are quite well understood, however they require continued research progress as a function of their level of maturity.
- 3) **Determining how the parts interact** is not a straight forward process, and interactions are often found in flight tests and field work. One example of progressing past understanding the ‘individual part’ for aircraft design is the area of Multidisciplinary Optimization (MDO). It started as efforts to integrate the calculation of air loads into the structural optimization. This field is also expanding, e.g. by inclusion of control aspects. We thus have a question of how to define the interactions, how to assess the significant interactions vs. the negligible. The appropriate answer can be expected to be unique for different applications.
- 4) **Defining the contribution of each component to system performance** is a continually improving process. But we can ask the question of what system metric is appropriate if components are designed (optimized) to different aspects. Also, what system model is needed for any particular aspect? What is the necessary level of fidelity?
- 5) **Putting the system back together** in the ideal sense would require a complete physics-based model. In the practical world, however, there are many questions to be addressed and many potential problems. How have the components been validated as being ready for system integration? One part of the preceding question is what data is experimental and what is analytical. It should not be assumed that any of the data is completely accurate, so that an explicit accounting of uncertainties is required. How have the component disciplines been coordinated during the design process? Linear superposition?
- 6) **The final aspect**, i.e. build it when the analysis shows that the design meets requirements, also raises many questions. What is the appropriate analysis or should it be analyses? The answer to this question may be not the same for all applications. There may be the exception for minor evolutionary upgrades to an existing system, but even that should not be taken for granted.

As a final comment, we strongly suggest that ‘System Engineering’ should not be applied as a sequential process, but as a continual iterative process following best practices as defined in the following section.

3.2 SYSTEMS ENGINEERING BEST PRACTICES

At the time this Task Group began, The Columbia Accident Investigation Board’s (CAIB) report [Anon 2003b] had just been released. In reviewing that report, the members of this Task Group found many lessons learned in the management of high-technology endeavours which are applicable:

BEST PRACTICES FOR SYSTEM INTEGRATION

- Don't use prior success as an indicator of future success or as a rationale for accepting increased risk. Don't normalize deviations (where undesired, unexpected or unexplained behaviour of a component of the system is progressively accepted as normal). Anomalies should not be used as a source of engineering data to justify further operation.
- Within technical and programmatic organizations minority opinions must be sought, encouraged, even created at times for a healthy debate. Play "devil's advocate" to force issues and data into the open.
- Never stop "what-if" games. They are the heart of an effective Failure Modes and Effects Analysis (FMEA).
- Individuals must assume "ownership" and believe they are personally accountable for the success of both their part of the whole and for the whole itself. (This was one of the characteristics of the members of the F-22A Flying Qualities Working Group and is viewed as a major contributor to the success of F-22A flight control development [Harris and Black]).
- Specify component, subsystem, and system characteristics and operating environment carefully and thoroughly, test to specification(s), and fly what you've tested (and test what you fly). If deviations occur and they are outside specification and test boundaries, retest to the revised boundaries and fix anything that does not pass the new test(s).
- Successful high-technology endeavours share a characteristic of very thorough, disciplined design, test build up, and operations processes.
- Systems Engineering takes a finite amount of time and resources to do it right. Compressing schedules increases risk, sometimes dramatically.
- Use trend analysis and statistical analysis to look for warning signs of pending problems.
- In the last 15 years there has been a strong push to apply organizational structures and philosophies which have been very successful in the mass production of consumer goods to Research and Development programs and organizations. These production organizations tend to be hierarchical and can be characterized by manage-by-efficiency ("faster, better, cheaper") organizational structures and methods. Chapter 8 of the CAIB report strongly criticizes the application of such management models to high-technology experimental efforts.

The final point is worth further elaboration. One of the members of the CAIB was Brigadier General Duane Deal. Subsequent to the release of the CAIB report he authored an article delineating his own views and lessons learned as a result of his participation in the Columbia accident investigation [Deal]. (The authors of this report **strongly** recommend **any** participant – engineer, manager, or other – in research and development of advanced technology read this article.). Among his observations, General Deal highlights the temptation to become focused on *process* over *product* (emphasis in the original), and says the following about managing high-technology endeavours:

"A healthy pessimism is required in high-risk operations. (A) preference for a clever analogy can serve as a recipe for repeating catastrophic mistakes, whereas insistence on analysis over analogy can prevent potentially disastrous situations."

"Although bombarded by "management by objectives"; Deming-driven, off-site quality thrusts; and "one-minute-management" techniques, leaders must ensure that the latest "organizational fad" does not negatively influence their operations. ...These principles ... work well in a manufacturing process producing 10,000 bolts a day, or at a scheduled airline where a technician may perform the same steps dozens of times per week. However, the same principles do not necessarily apply in an environment where only three to six flights are flown each year, and workers may accomplish certain processes just as infrequently. Process verification must be augmented when critical operations take place with an "eyes-on, hands-on" approach."

“To avoid developing a focus on metrics for metrics’ sake, the quantity being measured must be understandable, applicable, measurable, and the goal must be attainable. Ideally, there should exist a process that consolidates and assimilates data from multiple databases, providing a comprehensive picture of system performance, costs, malfunctions, and other trends of utility to management.”

“In the 1990s, the NASA top-down mantra became “Faster, Better, Cheaper.” The coffee-bar chat around the organization quickly became, “Faster, Better, Cheaper? We can deliver two of the three – which two do you want?” While the intent of the mantra was to improve efficiency and effectiveness, the result was a decrease in resources...”

“Leaders must contemplate the impact of their “vision” and its unforeseen consequences. Many must also decide whether operations should be primarily designed for efficiency or reliability. The organization and workforce must then be effectively structured to support that decision, each having a clear understanding of its role.”

“Leaders must remember that what they emphasize can change an organization’s stated goals and objectives. If reliability and safety are preached as “organizational bumper stickers,” but leaders constantly emphasize keeping on schedule and saving money, workers will soon realize what is (really) deemed important and change accordingly.”

Regarding Quality Control/Quality Assurance, “checks and balances using “healthy tensions” are vital to establish and maintain system integrity in programs from the federal government to aviation. High-risk operations dictate the need for independent checks and balances. Successful organizations must have a review process that addresses the findings and recommendations from third-party reviews and then tracks how that organization addresses those findings. To further this approach, leaders must establish and maintain a culture where a commitment to pursue problems is expected – at all levels of the program and by all of its participants.”

Regarding Configuration Control, “Leaders must insist on processes that retain a historical knowledge base for complex, legacy, and long-lived systems. Configuration waivers must be limited and based on a disciplined process that adheres to configuration control, updated requirements, and hardware fixes. If workers at the lower level observe senior leaders ignoring this path, routinely waiving requirements and making exceptions to well-thought-out standing rules, they too will join the culture of their seniors and begin accepting deviations at their level – adding significant risk to the overall system. Senior leaders must also ensure the steps required to alter or waive standing rules are clearly understood.”

The authors of this report assert that the Systems Engineering process as applied to the development of the flying qualities of the F/A-22 fits the criteria of healthy development outlined by Deal and the CAIB, and can be extrapolated to complete systems and to systems of systems. The key (from a technical perspective) is that a simulation model of the complete system be designated as the “truth model” of the system and be subjected to the disciplined development and configuration control process described above in this report. It should be noted that the “truth model” simulation need not include physics or thermodynamics based models at the component or subcomponent level (such as imbedding a cycle deck of a jet engine in the simulation), only that the component models within the truth model be sufficiently detailed for the purposes for which the simulation will be used, and that their operation and effects are directly traceable to the more highly-detailed component models (which is their “pedigree). Other keys displayed by the F-22 FQWG were the assumption of ownership of the aircraft and its safety and success by the members of the FQWG (already discussed) and a healthy scepticism and commitment to find and address potential problems and issues (expressed by one of the members of the FQWG as “if we are our own worst critic, we need not fear any independent review.”).

3.2.1 Land Systems

Although the unstructured, outdoor terrain encountered by unmanned ground vehicles (UGV) is extremely demanding, it is also very forgiving. Under most circumstances, especially R&D situations, a UGV can simply choose to stop. Thus, the response to a system problem or error is relatively straightforward, the UGV stops and waits for external assistance. A UGV's complex yet forgiving environment stands in stark contrast to a UAV's simple yet unforgiving environment. A UAV doesn't have the "stop" option, it must be under active control until it successfully lands on the ground. As a result, quality control and assurance are paramount for UAVs, but at this point in time are significantly less important for UGVs.

Historical and current UGV systems focus on sensing, representing and understanding the complex environment in which they must operate. This "obsession" with the environment means that UGV systems have not placed a significant emphasis on quality control and assurance. DARPA's second Grand Challenge clearly illustrates the current state of UGV system development. Twenty-three UGVs qualified for the race and only 5 UGVs completed the course. An analysis of the 18 unsuccessful UGVs revealed that only 3 of the failures were directly attributable to a misrepresentation of the environment. The remaining 15 failure modes ranged from mechanical failures, to electrical problems such as sensing failures, to software bugs. The teams that were successful recognized the importance of reliability and quality. The race winner, Stanley, formed an independent Testing Group whose sole responsibility was testing [Thrun et al, 2006]. Second and third places were taken by vehicles from Carnegie Mellon University, and this team's development process consisted of short development cycles interleaved with periods of intensive field testing [Urmson et al]. It should also be noted that all successful teams used commercially available vehicles, which undoubtedly contributed to more reliable implementations.

Chapter 4 – COMPLEXITY, AUTOMATION AND AUTONOMY

4.1 INTRODUCTION

This chapter addresses complexity, automation and autonomy. The advent of portable and powerful micro-processors has allowed control systems to become more sophisticated and complex. Integrating multiple control systems together can yield highly automated systems that exhibit autonomous capabilities. An autonomous system can make decisions without human guidance, thus the long term goal of autonomous control is to impart human-like capabilities. To achieve human-like capabilities, an autonomous system may require the ability to learn from experiences.

For both single vehicles and multiple vehicle systems, the design of the system needs a good grasp of the capability that is required to meet the desired performance requirements. For autonomous systems this requires a precise description of the environment in which the system must operate, together with the desired behaviours of the autonomous system. This in turn demands precise definition of the behaviour that the system is required to exhibit and under what conditions. Hence some time must be spent in defining autonomous behaviour and the complexity of both the environment and the complex system structure that is usually associated with autonomous systems.

This chapter discusses the following important topics such as the complexity of systems, man-machine interfaces yielding higher levels of automation, autonomous vehicles and associated problems, and artificial intelligence.

4.2 COMPLEXITY

As the control systems and computers are becoming more and more sophisticated and complex, they require a high degree of reliability and maintainability and they must have fault accommodation in order to operate successfully over long periods of time. Reconfigurable controller has to achieve the following goals:

- 1) Keep the system performance within acceptable boundaries during operation;
- 2) Increase the performance of the process; and
- 3) Achieve the goal for fault accommodation.

Reconfigurable control(ler) is a critical technology to detect the fault and recover the functionality of the faulty system as same as that of the nominal system. Various methods are used for reconfigurable control to cover the requirements of different applications. The behavior of the reconfigurable control depends upon whether the approach is passive or active. Such control ideas have been implemented on a variety of military and commercial applications in last two decades to accommodate faults, for example on flight control systems on space technology in and on unmanned underwater vehicles.

The steady increase in complexity of modern systems and infrastructures has placed strong demands on the requirements for control systems technologies. New advances in computing technology, microelectronics and intelligent devices (MEMS, self-validating sensor and architectures) have facilitated the development of powerful scientific and engineering methods in control. Almost all embedded systems now involve Control particularly at the high levels of embedding. Control technologies go significantly beyond the scope of traditional or classical solutions and now have the capabilities to address new challenges arising from large-scale networking, distributed service provision and sophisticated safety-critical applications. Complex Systems emerge in many disciplines and domains and have many interpretations, implications and problems associated with them. In addition to real-time requirements

these systems have to be dependable which most often requires fault-tolerance. It is well known that the development of fault-tolerant real-time systems is a very challenging topic.

An interesting characteristic of complexity is that higher levels of complexity tend to result in increased involvement of the human operator. Humans are difficult to model and their interaction with the plant, process or vehicle induces additional complexity and uncertainty. An interesting characteristic of complexity is that higher levels of complexity tend to result in increased involvement of the human operator. As systems increase in complexity this is becoming more important as more “pervasive” system tools become available. In recent years Automatic Control has been branching out in the directions of Software Engineering and Cognitive Science, giving rise to a discipline that can now be called “Embedded Cognitive Control”, as shown in the figure below.

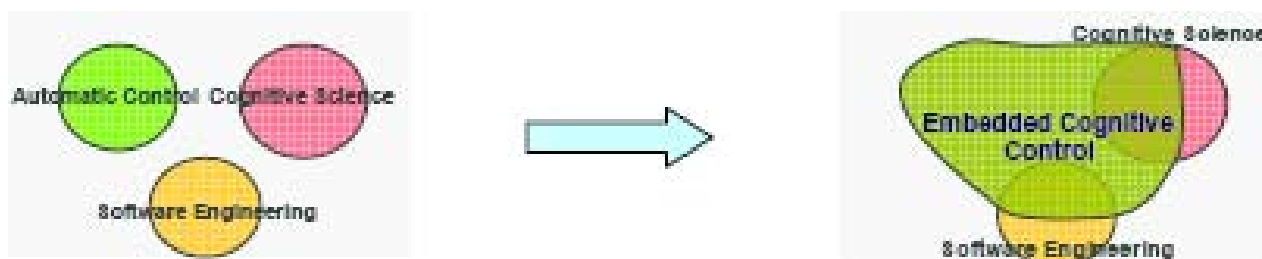


Figure 4.1: Embedded Cognitive Control Engineering Discipline.

The “human self” can be represented much more in terms of reasoning, as a cognitive agent and as a “digital system” (as a part of the embedded computing structure of the system, etc.). The human self can even be regarded as an “executable” system. It is apparent that as systems increase in complexity these emerging issues and concepts involving the role of the human operator become more and more important. In some aspects humans will be dominant in the sense that there are some abilities that cannot be replaced by an automated system (e.g., in the analysis of many decision situations, where a comprehensive analysis of diversified elements is not possible to be formalized), want to keep decision-making sovereignty while using a decision support system for analysing the problem. On the other hand in some situations a dynamic and/or complexity of system may require automatic control. New activities in advanced control face the challenge of the complexity (characterized by size, structure, irreducible uncertainty, risk, diversified performance measures, etc.) of modern engineering and business systems and enterprises, spanning bio-technology, information technology, space and aeronautics, vehicle systems, process and manufacturing systems, life sciences, the economy, etc.

Lui Sha, from the University of Illinois at Urbana-Champaign, in his article entitled:” Using Simplicity to Control Complexity. He has indicated his “call this approach using simplicity to control complexity”. Computational complexity is modeled as the number of steps to complete the computation. Likewise, we can view logical complexity as the number of steps to verify correctness. Logical complexity is a function of the number of cases (states) that the verification or testing process must handle. A program can have different logical and computational complexities. For example, compared to quick sort, bubble sort has lower logical complexity but higher computational complexity. The wisdom of “Keep it simple” is self evident. We know that simplicity leads to reliability, so why is keeping systems simple so difficult? One reason involves the pursuit of features and performance. Gaining higher performance and functionality requires that we push the technology envelope and stretch the limits of our understanding. Given the competition on features, functionality, and performance, the production and usage of complex software components (either custom or COTS) are unavoidable in most applications. Useful but unessential features cause most of the complexity. Avoiding complex software components is not practical in most applications. We need an approach that lets us safely exploit the features the applications provide.

The notion of using simplicity to control complexity ensures the critical properties. It provides us with a “safety net” that lets us safely exploit the features that complex software components offer.

“Control of Complex Systems” by Karl J. Aström et. al has extensive analysis of complex systems. This book is an example of the types of approach that European researchers are using to tackle problems derived from systems’ complexity. It has grown out of activities in the Control of Complex Systems (COSY) research program the goals of which are to promote multi-disciplinary activity leading to a deeper understanding and further development of control technologies for complex systems and if possible, to develop the theory underlying such systems. The material in this book represents a selection of the results of the COSY program and is organised as a collection of essays of varying nature: surveys of essential areas, discussion of specific problems, case studies, and benchmark problems. A selection of the results of the Control of Complex Systems research program, COSY, and is organized as a collection of essays. Topics include modeling and complex physical systems, control design, learning control, satellite attitude control, and passivity-based fault identification and fault tolerance.

4.3 MAN-MACHINE SYSTEMS

The knowledge about human behavior in man-machine system design for the vehicle with high level of autonomy can be used for:

- Design of operator’s station or other means in cabin of manned vehicles for monitoring of such vehicles.
- Design of intelligent systems by use of knowledge on human operator behavior.

These aspects of knowledge are discussed below.

4.3.1 Supervisory Control in Man-Machine System

The level of vehicle autonomy influences on role of a man in control of the vehicles. For the lowest level a human-operator acts as an active controller in man-machine system.

In such manual control tasks human-operator each moment reacts actively on perceived stimulus (visual cues, vestibular cues, etc.) by deflecting a manipulator for transition of control signals to the vehicle. The brief analysis of pilot behavior in manual control and usage of this knowledge for integration of human-operator and vehicle is given in Chapter 2. The scheme characterized the human-operator activity in man-machine system for manual control is given on Figure 2.1. The increase of level of autonomy changes the role of human-operator. He acts in that case as a supervisor. There is supposed that for case of supervisory control the task is accomplished by automatically and human-operator ability is to check its fulfillment and to act as a monitor. It means that operator ability is the monitoring of the control process. His active participation will take place in case when the task (mission) will be changed or system performance (for example, accuracy) will approach or increase the requirements. In principle pilot can fulfill the manual control and monitoring tasks simultaneously. The pilot workload influences considerably on distribution of pilot activity between control and monitoring. The decrease of pilot workload index will lead to increase of human-operator activity as a monitor. In other case, he has to be more active control element in control loop. The Figure 4.2 reflects man-machine system in supervisory control. The main design problem in supervisory control is optimization of interfaces provided minimum human errors in recognition of appearance and/or development of accident or failures with minimum operator workload.

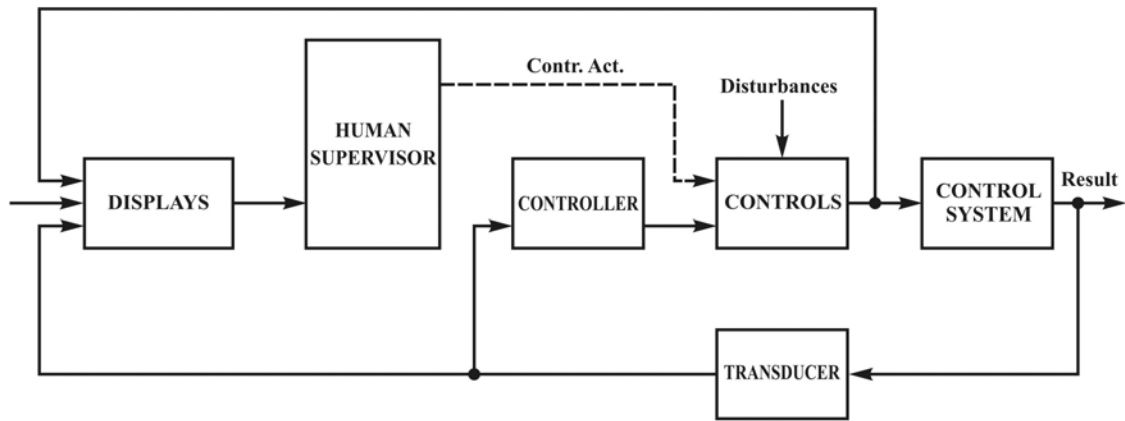


Figure 4.2: Man-Machine System in Supervisory Control.

The solution of this problem requires to develop the models of operator as a supervisor, to define the workload indexes (WI), and to select the secondary task (for evaluation of WI).

The model of human operator as a supervisor can be presented with help of the scheme shown on Figure 4.3.

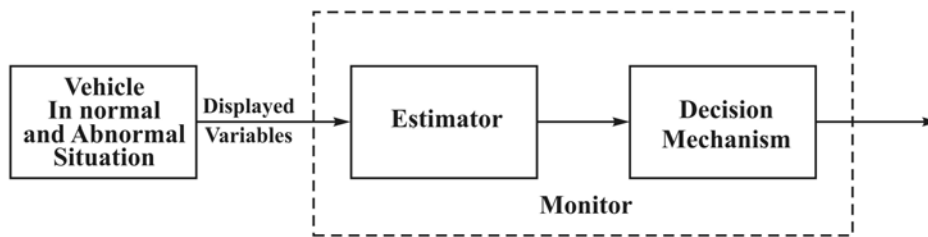


Figure 4.3: Human Operator Schematic.

It consists of two elements: a linear estimator and a decision mechanism. As a linear estimator they use Kalman filter estimates the state variables $\hat{x}(t)$ and the measurements $\hat{y}(t)$ as well as the measurement error (residual) $\varepsilon(t)$, where $\varepsilon(t) = y(t) - \hat{y}(t)$. The model takes into account the observation noise. For a number of instruments or metrics observed by operator the model takes into account the effect of sharing attention. If the operator observes more then one instrument his observation noise for each observation increases by a constant factor [Levison W.H., Elkind, J. and Ward J.,1971] and [Baron S., Kleinman D., Levison W., 1969]. The level of noise is inversely proportional to the fraction of attention that he spends monitoring that specific instrument.

Decision making mechanism can be based by different way. One of them proposed in [Gai E., Curry R., 1975] is based on sequential analysis. The last one uses the hood-like ratio $l(m)$ as a decision function after m observations. The two criteria levels, A and B are chosen, and the decision rule is given by the sequence:

- if $l(m) \geq A$ choose failure
- if $l(m) \leq B$ choose normal
- if $B < l(m) < A$ take another observation.

Where A and B are determined by desired probability of false alarm P(FA) and probability of miss P(MS) follows:

$$A=1-P(MS)/P(FA)$$

$$B=P(MS)/1-P(FA)$$

One of the problems in definition of the best way for presentation of symbols and their location, indicator design, is the low sensitivity of the criteria used for solution of the problem (for example, detection time) to the workload index used in researches. As an example, the dependence of detected time as a function of normalized workload is shown on Figure 4.4.

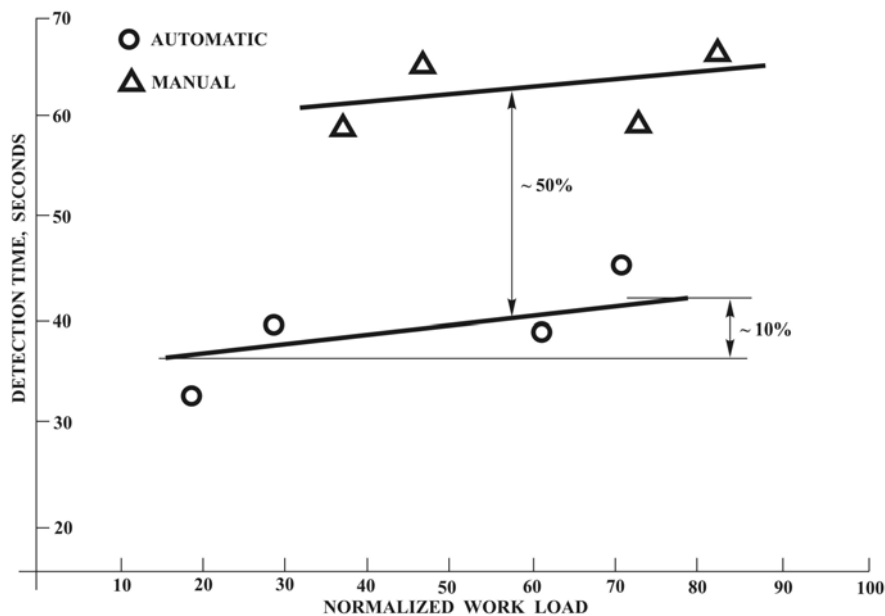


Figure 4.4: Detected Time vs. Normalized Workload.

The research [Ephrath A.R., 1975] was dedicated to the estimation of detection time required to define a failure of one of the instruments. The research was fulfilled for manual control when pilot uses the observed information actively acting as controller and for automatic regime when pilot's role was a monitor. The research was fulfilled for different pilot workload, associated with fulfillment of the secondary (no control) task. There is seen that in average the increase of normalized workload in its wide range causes the insignificant increase of detection time (up to 10% only). As for manual control the detection time was 50% higher in comparison with automatic control.

This result leads to idea to use the manual control task as a secondary task for definition of detection time. It might be more sensitive test for solution of applied design problem. As an example the first order

unstable plant $\left(W_c = \frac{\lambda}{p - \lambda} \right)$ with different level of unstable root can be used for the secondary task and

the value $\bar{\lambda} = \frac{\lambda_s}{\lambda_c}$ as a WI (here λ_c maximum value of root achieved in single loop manual control

task), λ_s – value λ_s corresponded to the specific level of error.

4.3.2 Adaptive and Intelligent Systems Grounding on Human Operator Experience as a Basis for System-Level Integration of Control

4.3.2.1 Intelligent Control Systems as a Reflection of Experienced Human Pilot Skills

The intelligent control systems are the systems having an ability to emulate human capabilities, such as planning, learning and adaptation. Intelligent control systems may be considered as a reflection of experienced human pilot (operator) skills in some artificial media. We would like to generate a control system, which will be comparable with a test pilot from the point of view of control skills.

It is necessary to define what is meant above by intelligent, a term used here to refer to a specific class of problem solving. The technical committee on intelligent control of the IEEE Control Systems Society has defined the general characteristics of intelligent control systems as having an ability to emulate human capabilities, such as planning, learning and adaptation [Linkens, D.A. and others, 1996]. Learning and adaptation especially are essential characteristics of intelligent control systems and, while adaptation does not necessarily require a learning ability for systems to be able to cope with a wide variety of unexpected changes and environments, learning is invariably required.

As it is shown below, intelligent control systems must allow solving of control tasks, which are too difficult or unsolvable by means of traditional control techniques.

4.3.2.2 Control Problems for Advanced Manned and Unmanned Aircraft – General Description

There are many tasks associated with flight control for modern and advanced aircraft including unmanned aerial vehicles (UAVs), which are not solved (or solved very unsatisfactorily) with traditional tools. It has been recognized in recent years, that realization of more flexible and effective control systems requires to combine other elements, such as logic, reasoning, heuristics etc., with the algorithmic techniques provided by conventional control theory, and such systems are known as intelligent control systems.

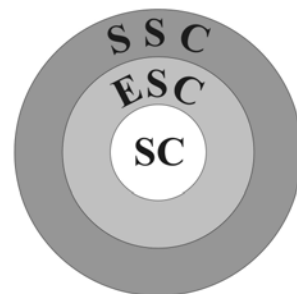
Evidently uncompleted list of such tasks includes: flight control for agile and post-stall aircraft; flight control in complicated cases (influence of atmospheric turbulence, wind shear, flight and landing in complicated weather conditions, landing on aircraft carrier, refueling etc.); flight control with possibility of sharp or smooth changing of mathematical models for aircraft motion caused corresponding changes in vehicle shape and parameters (dropping internal and/or external stores, damages of aircraft, engines, avionics – sharp foreseen and/or unforeseen changes; maintenance of aircraft with taking into account smooth changes in its shape and parameters – icing of planes, wear of aircraft systems, expenditure of fuel from tanks etc.); and flight management and control in case of group of vehicles. These tasks can be characterized with such features as:

- Wide range of conditions (flight modes, motion parameters, external disturbances etc.) needed to take into account;
- Presence of many uncertainty factors belonging to various classes;
- Essential nonlinearity of aircraft characteristics;
- Essentially non-steady nature of processes realized with aircraft as controlled system;
- Broad using of supersonic flight regimes to implement aircraft supermaneuverability;
- Possibility of many abnormal modes caused by various kind of failures and damages in aircraft structure and equipment as well as with some external effects;
- Unmanned advanced aircraft autonomy challenge; and
- Necessity of collective operations for vehicles (cooperative actions, collision avoidance etc.) including a case of dissimilar vehicles.

4.3.2.3 Semi-Soft Computing as a Basis for Implementation of Intelligent Control Systems

Research in the intelligent control field is based mainly on soft computing methods and tools as well as on extensions of these ones. It can be divided on three levels of the methods and tools needed to solve problems mentioned above [Brusov, and others, 1996, Tiumentsev Yu. V., 2002, 2004a, 2004b] (Figure 4.5):

- Soft computing (SC) methods and tools: artificial neural networks (ANN), fuzzy logic (FL) systems, evolutionary techniques (genetic algorithms (GA), genetic programming etc.), uncertainty management techniques;
- Extended soft computing (ESC) methods and tools: soft computing methods and tools together with knowledge-based systems (KBS) and multiple-agent technologies; and
- Semi-soft computing methods and tools: extended soft computing methods and tools together with mathematical modeling (MM) techniques.



$$SC \subset ESC \subset SSC$$

Figure 4.5: Relationships between Soft Computing (SC), Extended Soft Computing (ESC) and Semi-Soft Computing (SSC).

It should be emphasized that only presence of system elements based on soft and semi-soft computing techniques **do not cause** this system to be intelligent. After all, these elements can merely “replace” some other elements based on conventional techniques. For example, we can take a controller (a control channel in particular) and approximate its functional dependencies for gains in control law by means of some artificial neural network. After that the ANN-based representation can be used to compute gains related to some particular situation. A representation form is changed here but the essence of control law remains permanent. The controller becomes “ANN-based” but it still does not become “intelligent”. Therefore “neural control”, “fuzzy logic control”, “neuro-fuzzy logic control” and other similar terms are not synonyms at all for the “intelligent control” term. Some system will be “intelligent” only if it is capable to adapt and to learn (Figure 4.6).

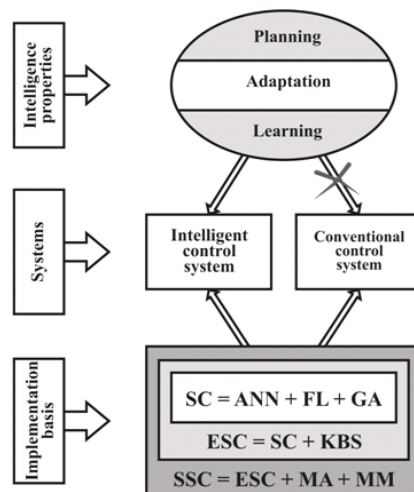


Figure 4.6: Intelligent and Conventional Control Systems versus Soft and Semi-Soft Computing.

It means intelligent systems can reconstruct effectively their behavior depending on current situation as well as accumulate solution experience for various tasks and use this experience to solve earlier unknown tasks. It's quite another matter that a system named as "intelligent" can be hardly implemented without soft and semi-soft computing tools. So then realization of a control system basing on semi-soft information technology is most likely a necessary condition and is not a sufficient one.

Based on soft and semi-soft computing, intelligent flight control are directed to reply the demands solving such main problems as: enhance the mission capability of aircraft; improve aircraft performance by learning from experience; make aircraft less dependent on proper human actions for mission completion; increase flight reliability and safety.

4.3.2.4 Adaptive and Intelligent Systems as a Basis for System-Level Integration of Control

4.3.2.4.1 Adaptation in Controlled Systems

A need for adaptation and adaptive systems arises as a rule in case of multiple interconnected task solved under uncertainty conditions and severe resource constraints.

Adaptive systems play very important role as a part of all kinds of complex systems especially air vehicles both piloted and unmanned as well as vehicles for space, land, and sea domains. Also intelligent systems demonstrate growing significance in recent years.

Adaptive systems are thought in some broad sense namely as systems which are capable to modify their behavior according to varying conditions of their existence (environment, goals etc.). In other words adaptive system is a system, which has some mechanisms providing a possibility to live and to act in conditions of various and numerous uncertainties. One of most important mechanisms of this kind is intelligence, although it is not a unique one.

Four kinds (hierarchical levels) of adaptation can be distinguished:

- Parametrical adaptation (adjustment, self-adjustment);
- Structural adaptation (reconfiguration and/or restructuring);
- Object adaptation (correction of system composition); and
- Goal adaptation (adjustment of demands).

4.3.2.4.2 *Parametrical Adaptation*

Is supported with variation of control system adjustment parameters, for example, controller gains.

4.3.2.4.3 *Structural Adaptation*

A required versatility for a controlled system not always can be reached only with varying values of control system parameters. Next hierarchical level for adaptive systems are systems, which are capable for structural adaptation, i.e. for a modification of the system structure (it includes a set of control system components as well as links between these components) in regard to changing situation and goal. Simplest case is a control system equipped with a set of alternative versions of control laws. Only one of the versions operates at some particular instant.

4.3.2.4.4 *Object Adaptation*

A case is quite possible when no variation of control system structure and parameters exists to satisfy goals of the controlled system. It is quite natural because of potentialities are restricted for any system; the potentialities are constrained with the system "structure". If such a case arises then we can involve next adaptation level, i.e. object adaptation.

The principal idea of this adaptation level consists in solving of a required task by means of a set (group, constellation, swarm) of interacting systems instead of some separate controlled system as it was for two previous cases.

The essence of this approach can be illustrated for a task of intercept of aerial targets, including multiple ones. If airspace area and number of targets are relatively small then the task can be solved often with a single interceptor equipped with missile weapon and multi-channel system for finding and tracking of targets. If these conditions are not satisfied then capability of sole aircraft is not adequate to the task. In that case we can form some heterogeneous group of systems intended to solve cooperatively the mutual task. Such group can include vehicles of several types, ground-based equipment and so on.

4.3.2.4.5 *Goal Adaptation*

If object adaptation level is not adequate to the task similarly two previous levels, i.e. the level does not provide achievement of the goal, it is quite possible the stated goal is not achievable at all. We can change control goal to make it achievable.

The essence of this adaptation level can be illustrated in the following way. Let a task be surveying of particular object for some self-propelled vehicle delivered onto a celestial body. It can be revealed that solving of the task demands too many resources. This circumstance impends to defeat the plans of the expedition. If that's the case, the system can replace one goal with other following some general purposes (for example providing highest possible extraction of knowledge about the celestial body). The system can search for that an object "similar" to excluded one or to refuse this element of the exploration plan at all and to switch over to other plan elements.

4.3.2.4.6 *Implementation Basis for Adaptive and Intelligent Systems*

There are some advanced research and engineering areas associated with various sorts of adaptive and intelligent systems. As it was mentioned above list of these areas includes first of all soft computing technologies which combine into one such approaches as artificial neural networks, fuzzy systems and evolutionary techniques (genetic algorithms, genetic programming etc.). If we add knowledge-based systems to the list then we get extended soft computing. The addition of numerical simulation techniques to the extended soft computing leads to semi-soft computing.

During past 12–15 years methods and tools involved in soft computing, extended soft computing and semi-soft computing have been developed very intensively. Many interesting results have been obtained and accepted [Samarin, A.I., 2005], [Brusov, V.S., and others, 2004], [Stengel, R.F., 1993], [Suykens, J.A.K., and others, 1996], [Sontag, E., 1993], [RayChaudhuri, T., and others, 1995], [Ronco, E., and others, 1997], [Haykin, S., 1994], [Berthold, M., and others, 2003], [Michalewicz, Z., 2001], [Domany, J.L., and others, 1992], [Piegat, A., 2001], [Pal, S.K., and others, 2003], Polkowsky, L., and others, 2000]. Among these results we have those related to all aspects of management and control problems for complex systems: observation, sensor fusion, estimation, control, guidance, navigation, mission planning, situation assessment, decision making and so on.

The results obtained allow to assert that extended soft computing and semi-soft computing techniques provide possibilities to solve management and control problems with conditions unacceptable for traditional control theory methods, for example, for a case with large abrupt change of controlled system configuration.

These possibilities seem to be very important for piloted vehicles as well as for unmanned vehicles, especially to solve problem of system-level integration of control. Such possibilities could be helpful for many subjects and efforts including integration of control for both stability and maneuvering, integrated flight and fire control, integrated flight and propulsion control, automatic ground collision avoidance, swarms of unmanned vehicles, cooperative actions of dissimilar vehicles, safe mixing of manned and unmanned vehicles, automated mission performance of unmanned vehicles, automatic air collision avoidance and many others.

As it was mentioned already it hardly makes sense to search some “miracle cure”, which is some kind of information technology (IT) alternative with respect to the traditional imperative-kind IT, to overcome revealed obstacles. Each IT picked up from the long list of modern traditional and advanced information technologies has both virtues and shortcomings. However as experience demonstrates capability of any single IT is not sufficient to solve all the problems arising during development and maintenance of complex systems especially adaptive and intelligent ones. It could be more productive to combine various information technologies for the purpose of using their strong features and to compensate mutually their inherent shortcomings. Based on these reasons it is not rational also to oppose traditional information technologies and advanced ones as well as some advanced technology to each other.

We have to work for *synthesis* of concepts, techniques and tools consisted in these information technologies. It is the foundation that allows to solve the problems mentioned in previous sections. The most important goal of such synthesis is to define some kind of sufficiently general scheme that will be called as semi-soft computation (SSC) model. This model must allow to obtain partial (special-kind) models for various branches of the SSC namely for artificial neural networks (ANN), fuzzy logic (FL), genetic algorithm (GA), knowledge-based system (KBS), multiagent system (MA) as well as conventional mathematical modeling (MM) by putting into consideration appropriate requirements and conditions.

Let us outline an approach to realization of the semi-soft computation model.

One of the key elements of the SSC approach is an idea of neural network as a special kind composition of mappings and, on the other hand, as a dynamical system what is especially important for feedback neural networks. Such concept makes possible to reveal deep relationships between artificial neural nets and other fields of the SSC as well as their interrelations with conventional mathematical modelling. Similar approach is proved to be effective in respect of other SSC parts namely FL, GA, MA and KBS. Most generally the semi-soft computation model can be interpret as a composition of some special-kind mappings having either static (for instance perceptron-like feedforward networks) or dynamic (for example recurrent neural networks) nature. This model can be reduced to certain “pure” model based solely on the FL, GA, KBS, MA, MM concepts or to some “truncated” hybrid model (e.g. ANN+FL,

GA+ANN, ANN+KBS, GA+MA etc.) applying corresponding conditions. Such kind of representation for the SSC model opens up possibilities to rigorous mathematical analysis of the model and its various specialized versions as well as to define appropriate theoretical foundations for the SSC.

One exists however another aspect of the discussed problem. A computing experiment is needed to evaluate efficiency and capability for approaches and techniques offered. Therefore it is quite important to represent the SSC model not only keeping its mathematical analysis in mind but also taking into account peculiarities of its computing implementation. It seems very promising to accomplish this purpose by using an extended semiotic model which is the second key element of the SSC approach. This model is based on a generalization of the semiotic model concept suggested by Dmitry Pospelov in [Pospelov, D.A., 1986]. The generalization is accomplished as a result of “composition-style” semiotic model reformulation corresponding to the SSC approach.

A value of suggested theoretical (formal) models will be not so much without appropriate procedures which allow to reassert various kinds of application problems in terms of the SSC. In this connection the third key element of the SSC approach has to be a development of relevant tools ensuring the reassertion.

The fourth key element of the SSC approach consists in development of suitable problem solving techniques. These techniques are based on concepts and tools of the SSC branches (ANN, FL, GA, KBS, MA and MM) transformed to keep the SSC-style of model representation, i.e. using special-kind composition of mappings plus extended semiotic model.

Available results demonstrate very deep interconnections between such semi-soft computing areas as ANN and MM, ANN and FL, ANN and MA, FL and KBS, FL and GA, ANN and GA, KBS and conventional imperative technologies etc. Therefore one might conclude that it is very important to reveal, study and develop such kind of relationships because this approach promises to integrate really achievements from various SSC fields into comprehensive whole system. Besides a possibility emerges to formulate corresponding methodological principles and to derive suitable theoretical foundations for the SSC. These results are quite needed to build an information technology based on the SSC concepts and models. This technology, in turn, makes possible to increase considerably a level of complexity for application problems, which are available to solve them effectively.

4.3.2.4.7 *Multiagent Architecture for Adaptive and Intelligent Systems*

Adaptive and intelligent systems are very complicated ones. Complexity of these systems prevents their effective implementation for useful applications by means of conventional software architectures based on pure imperative information technologies. There is an alternative way to implement adaptive and intelligent systems involving *multiagent models* and corresponding *multiagent systems*.

The multiagent system is a system consisting of numerous autonomous agent modules interacting with an environment (Figure 4.7) and having such properties as:

- Each agent possesses an autonomy;
- There is no centralized control for agents;
- Data sources and data access are decentralized; and
- Agents operate in asynchronous mode.

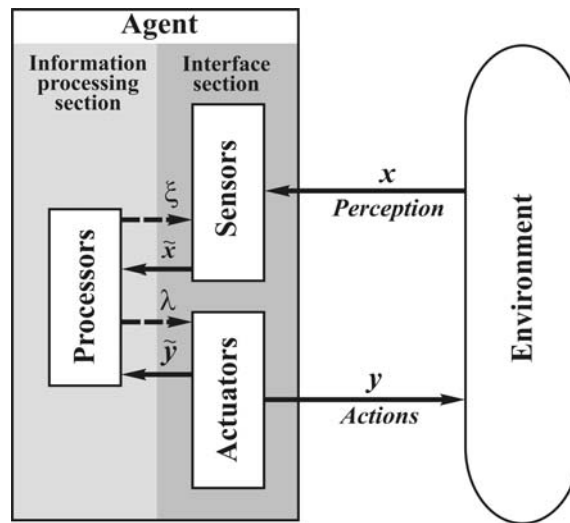


Figure 4.7: Generalized Structure of Autonomous Agent and its Interaction with Environment.

The multiagent model is spreading now more and more widely including complex system development processes. Conventional approach to solve the complex system development problem is based on representation of the system as some subsystem hierarchy with subsystems rigorously subordinate to each other as well as with explicit structurization of all interconnections and interactions between subsystems.

The new approach introduces a concept of autonomous system called agent instead of the subsystem concept. The agent has a high autonomy degree and it is independent of other agents. The rigorous set of relations for conventional-style model is replaced with a set of rules (protocols) defining interactions between agents. Besides some set of procedures is usually introduced to allow agent interactions if we are needed cooperative behavior for the agents solving certain common problem. This approach ensures to reduce considerably complexity and expenditures for the systems developed including adaptive and intelligent ones. Complexity of interactions between parts of the systems can be also reduced significantly. Therefore a true possibility appears to create really useful application systems.

Two examples of multiagent architectures are shown in Figure 4.8: multiagent model based on direct interaction with the environment and without any coordination between agents (Figure 4.8a); and multiagent model involving agent cooperation via a blackboard (Figure 4.8b).

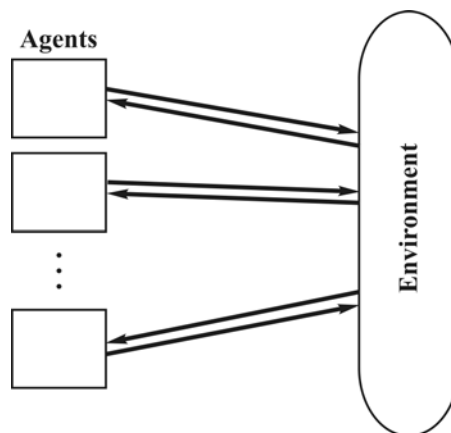


Figure 4.8a: Multiagent Architectures Based on Direct Interaction With the Environment and Without any Coordination Between Agents.

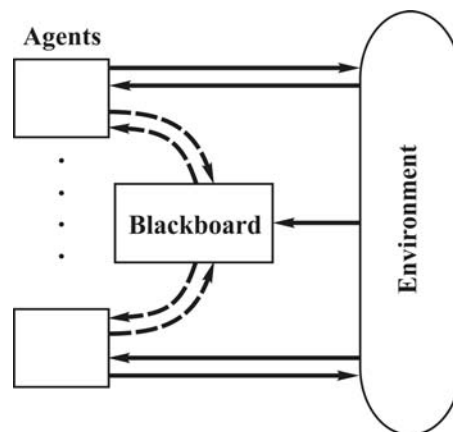


Figure 4.8b: Multiagent Architecture Involving Agent Cooperation via Blackboard.

4.3.2.4.8 *Intelligent Autonomous Vehicle as a Typical Semi-Soft Computing Application Example*

There are developed and widely used manual control and automatic control techniques as well as automated control techniques combining manual and automatic ones. Development of advanced piloted vehicles as well as highly autonomous unmanned aerial vehicles demands some new ways to solve automatic and automated control problems with the regard conditions listed above.

Grounding on soft and semi-soft computing techniques we can solve problems associated with highly autonomous vehicles including a case of intelligent autonomous vehicles (IAV). The IAV are systems which are capable:

- To reach specified goals within highly dynamical environment taking into account various and numerous uncertainties;
- To modify specified goals and to generate new goals and goal sets basing on some motivations;
- To extract (to mine) a new knowledge, to accumulate and generalize experience in solving of diverse problems, to learn basing on the experience, to modify system behavior basing on the new knowledge and experience;
- To adapt to problems which are needed to solve including some problems not presented at the original system design; and
- To form “collectives” (“communities”) made up from IAVs directed to cooperative solving of some common complex problem.

An activity of IAV within some environment can be divided into three main parts:

- Perception of a current situation (situation = external-situation + internal-situation), i.e. revelation of a challenge – sensor functions;
- Producing of a system reaction (“system’s reply”) for the current or predicted situation (kinds of possible reactions are system state evolution, reconfiguration, restructuring, adaptation of goals, unsupervised learning, self-organization etc.) – decision making functions; and
- Implementation of reaction produced for the current or predicted situation – effector functions.

A capability to learn and to accumulate (and to generalize) experience is provided for a controlled system a very high adaptability level in regard to variations in activity conditions.

COMPLEXITY, AUTOMATION AND AUTONOMY

In particular, a possibility arises to solve such problems as:

- 1) System-level integration of control (one-vehicle-level) for piloted and unmanned air vehicles:
 - Integration of control for both stability and maneuverability;
 - Integrated flight and propulsion control;
 - Integrated flight and fire control;
 - Automatic ground collision avoidance;
 - Reconfiguration and restructuring of control as a reaction on large abrupt changes of controlled system configuration;
 - Structure adaptive and intelligent control (smart materials and structures, adaptive structures);
 - High-autonomous intelligent vehicles; and
 - ... many others.
- 2) System-level integration of control (multiple-vehicles-level) for piloted and unmanned air vehicles:
 - Automatic air collision avoidance;
 - Cooperative actions of dissimilar vehicles;
 - Safe mixing of manned and unmanned vehicles;
 - Tasks for communities of high-autonomous intelligent vehicles;
 - Swarms of unmanned vehicles; and
 - ... many others.

4.3.2.5 Intelligent Control Techniques Based on Human Operator Experience

Let us consider in more details the second element of the IAV activity structure because of key role of the decision-making (control) functions.

There are many ways to solve the problem. One of these ways is based on using of rich control experience accumulated in piloted aviation.

There are three ways to generate intelligent control laws needed to solve tasks indicated above:

- “Mimic” approach based on using of some neural or fuzzy-neural network or some ensemble of networks to imitate control actions produced by human pilot (operator); this approach essentially uses experience accumulated by pilots to generate control laws implemented with automatic systems; the pilot experience needed for “mimic” approach realization can be revealed by using of flight simulators;
- “Formal” approach based on learning of some neural or fuzzy-neural network according to certain set of indices describing required behavior of controlled system; this approach do not use any pilot’s experience at all; and
- “Combined” approach which merges “mimic” and “formal” approaches; in accordance with this approach some controller is constructed and learned at the beginning by means of “formal” approach tools, then such controller is refined using “mimic” approach tools using flight simulator and knowledge base containing pilot experience.

A human operator experience and skills can be used in the context of mimic and combined approaches by means one of such two ways as:

- Imitation of operator's behavior for specified conditions in solving some control problems; and
- Synthesis of a control law implementing human operator's experience and skills by means of approximation of dynamical model, which describes operator's activity in solving some control problems.

Implementation of mimic or combined approach causes some problems. The problems arise as a result of necessity to generate an approximate representation of human operator as a dynamical system.

The human operator as a modeled object is characterized with such peculiarities as:

- 1) Operator's model is generally very complicated, nonsteady and nonlinear especially for cases of multi-channel control tasks, influence of complex external impacts and disturbances, control tasks in the event that dynamic of controlled object changes in sharp and unpredictable manner (for example because of structural damages and equipment failures of controlled object).
- 2) Operator's model depends essentially on a task which needed to be solved. There are many of such tasks in the case of complex multiple-mode controlled object. Accordingly, a lot of different models are needed to describe appropriately human operator activity as a whole.

Thus it is necessary to solve approximation problem for dynamical systems realizing complex and multiple-variant behavior.

Suppose a considered dynamical system realizes a transformation (mapping) F of input signals x into output signals y . It is necessary to find an approximate representation for F such that a behavior of the dynamical system with mapping F would be "similar" (in some predefined sense) to a behavior of a source system, i.e. some system with a human operator as a controller.

Mathematically and computationally this problem is rather difficult.

Let us assume for the sake of definiteness that inputs x and outputs y are vectors with elements, which belong to one of two spaces, R and $C[a,b]$: R is the space of real numbers and $C[a,b]$ is a space of real-valued continuous functions on the $[a,b]$ interval, where $a, b \in R$.

Problems of approximate representation for mathematical objects like F can be divided into four types depending on a kind of inputs x and outputs y :

- 1) Problems with $x \in R$ and $y \in R$; they are traditional approximation problems for some function F .
- 2) Problems with $x \in C[a,b]$ and $y \in R$; they are approximation problems for some functional F defined on functions $x \in C[a,b]$ and possessing real number values $y \in R$.
- 3) Problems with $x \in R$ and $y \in C[a,b]$; they are approximation problems for some differential operator F , which depends on real-valued parameters $x \in R$ and possessing function values $y \in C[a,b]$.
- 4) Problems with $x \in C[a,b]$ and $y \in C[a,b]$; they are approximation problems for some integral operator F defined on functions $x \in C[a,b]$ and possessing its function values $y \in C[a,b]$.

First type of problems is traditional approximation problems for functions. A vast majority of works associated with approximation problems for mathematical objects including neural network based investigations is given up to such kind of problems.

COMPLEXITY, AUTOMATION AND AUTONOMY

Second and third types of problems are some problems related to approximation for systems of differential equations. There are traditional ways to solve these problems (difference schemes, functional expansions etc.) as well as non-traditional ways based on using of artificial neural networks, for example. However quantity of papers realizing non-traditional approaches to the problem is relatively small now.

Fourth type of problems is certain problems associated with approximation of integral equations. The state of things in this area is about the same as for the second and third type problems.

Approximation problems related to dynamical systems like human operator concern to the second and third type problems mentioned above. Conventional (traditional) techniques to solve these problems are hardly applicable because of their inconvenience, inflexibility as well as large requirements in computational resources. Techniques based on artificial neural networks are more preferable here. They allow handling considered problems.

Neural network based implementation of intelligent control laws by means of approximation of appropriate dynamical models causes some problems:

- 1) There are many control tasks realized by human operator for sufficiently complex controlled object. It would be ineffective to generate a separate neural network for each individual task, especially for a case of expandable list of tasks. It will be better to use a single network or a system of interconnected and interacted neural networks instead of a set of individual networks.
- 2) An additional on-line learning can be required in many cases, for example if dynamic of controlled object changes in some sharp and unpredictable manner. Under the circumstances arise very hard limitations on operating speed of adaptation mechanisms.

There are two ways leading to solution of these two problems:

- Neural networks with a dynamical preliminary adjustment; and
- Ensembles of neural networks.

Neural networks with a dynamical preliminary adjustment are directed to realization of complex and very complex implicit functional dependencies. These networks include dynamically adjustable work elements (neurons) as well as so-called interneurons, which influence on tune parameters of the work neurons depending on values of input network signals and it possibly subject also to values of some additional parameters. The network is capable to absorb multiple models simultaneously by means of combinatorial interneuron inhibition process. The interneurons may receive their input signals from some other network solving a classification problem. A solution of this problem represents a name (or a number) of the model, which is adequate to current work conditions. These networks can be learned additionally to react on an expansion of the associated task list. Previous experience of the network is preserved and enlarged, i.e. additional training do not destruct its preceding capabilities.

Some different approach is ensembles of neural networks. An ensemble of neural network is used for implementation of required task set according to this approach. Each of these networks associates with some tasks from the task set. Two architecture version of the ensemble is possible depending on type of interaction between networks involved:

- Architecture version based on a model “master–slave” (it can be named more exactly as “conductor–performer” model); and
- Architecture version based on multiple-agent model.

For the “conductor–performer” model there are N neural networks execute required tasks and some $(N+1)$ th network, which “conducts” the other nets: “conductor” orders and “performers” obey.

In case of the multiple-agent version of the network ensemble there are N relatively autonomous and relatively independent neural networks associated with the task set. These networks interact with each other according to some rules. Such interaction is organized to solve all tasks entered in the task list.

It may be stated concluding the discussion that human operator experience and skills together with techniques of semi-soft computing allow to solve various important tasks associated with the problem of system-level integration of control.

4.4 AUTONOMOUS VEHICLES

4.4.1 Automation and Autonomy

The Free Dictionary (www.tfd.com) defines automation as:

- The automatic operation or control of equipment, a process, or a system.
- The techniques and equipment used to achieve automatic operation or control.

Humans have been developing automated systems for centuries. With the advent of modern computers, the complexity of automated systems has risen exponentially and has resulted in systems that exhibit autonomous capabilities.

Autonomous is defined as:

- Not controlled by others or by outside forces; independent, and
- Independent in mind or judgment; self-directed.

Due to the generic nature of autonomy, individual disciplines have defined their own unique meaning to autonomy, with is specific for the context of their research. The following sections reflect the diverse meanings that researchers have applied to the words “Automation” and “Autonomy”.

4.4.2 Autonomy for Robotics Systems

Under the Wikipedia definition, autonomous robots “are robots which can perform desired tasks in unstructured environments without continuous human guidance.” Given this definitions, one route to achieving autonomous operation is the SMPA paradigm [Brooks], where robots sense, model, plan and act. Thus, the components of autonomy may be itemized as follows:

- 1) **Sensing:** Via proprioception, the robot senses its own internal status, while exteroceptive sensors allow the robot to sense the external environment.
- 2) **Modelling:** The robot constructs a world representation using sensor data. The representation’s fidelity must allow obstacles and hazards to be identified. Commonly used world representations include: occupancy grids [Moravec], 2½ D representations [Brotten 2007a] and, potentially, 3 D representations. Into this representation, the robot may insert other types of information such as known landmarks and features.
- 3) **Planning:** In order to accomplish a task, the robot must plan a course of action. This plan must account for the robots internal status, the current environment, and the task or mission goal.
- 4) **Acting:** With a plan in hand, commands must be sent to the actuators and motors that physically propel the robot. Historically, a monolithic SMPA implementation resulted in poor performance as modelling and planning caused the system to be slow and unresponsive. In order to make the act stage responsive, a hybrid architecture, that is composed of both reactive and deliberative control strategies, is commonly implemented [Rosenblatt, Simmons].

In addition to the traditional SMPA paradigm, some leading edge robotics systems have adding learning to enhance autonomous capabilities. Learning can be invaluable as it allows a robot to adapt to environment changes such as variations in lighting conditions and seasonal environmental changes [Thrun].

4.4.3 Complexity for Autonomous Systems

For autonomous systems, complexity manifests itself in different ways:

- 1) Complexity of the operational environment.
- 2) Complexity of the task or mission to be performed.
- 3) Co-operation between multiple autonomous systems also leads to complexity.

The following sections provide more details with respect to the complexity encountered by each of the above items.

4.4.3.1 Environmental Complexity

The environment encountered by an autonomous system varies greatly. Both UAVs and UUVs operate in relatively simple and forgiving environments. Once above the ground plane, a UAV's environment is almost completely obstacle free. While operating in this benign environment, a UAV has only minimal requirements to sense its world, and no need to create a world representation. Although a world representation is not required, there are environmental conditions that add other forms of complexity. This complexity can be lumped into two large groups:

- 1) The atmosphere directly affects the vehicle's motion. These effects include turbulence, shear, and vortices. These atmospheric effects may have a considerable influence on the vehicle's linear and angular motion, and are potentially catastrophic in terms of accidents (recall wake effects caused by large vehicles on smaller ones, for instance).
- 2) Other contributors to complexity include: boom motion during aerial refuelling, deck and optical system motions during carrier landing, target maneuvering and landing on extremely short runways.

UUVs operate in environments that are similar to those encountered by UAVs. A UUV, operating in deep water, can also assume a benign and almost obstacle free environment. As UAVs and UUVs approach ground terrain the environmental complexity increases significantly and eventually approach the complex, unstructured environments in which UGVs must operate.

UGVs must operate over a wide range of environments, from those with low complexity, to those with high complexity. The following items are example UGV environments.

- 1) A low complexity environment as shown in Figure 4.9 (Parking lot or highway, open fields with no or very little vegetation).



Figure 4.9: A Low Complexity Environment.

- 2) A medium complexity environment, which is shown in Figure 4.10 (Open fields with vegetation, gravel or dirty roads, urban streets).



Figure 4.10: A Medium Complexity Environment.

- 3) A high complexity environment is shown in Figure 4.11 (Heavily vegetated terrain such as forests or jungles, terrain featuring rubble whether it be in a natural or urban setting, building interiors, actively hostile areas).



Figure 4.11: A High Complexity Environment.

Currently, autonomous systems usually operate in low complexity environments. Research robots have a limited ability to operate medium complexity environments. Operations in high complexity environments are not possible given the current state of autonomous systems development.

4.4.4 World Representation and Understanding

Due to the complexity associated with ground environments, autonomous ground vehicles must sense and, at some level, understand their environment. The process of representing and understanding the world may be considered from many points of view. A stationary sensor can create a world representation, but its inability to move means this representation has limited internal uses. A sensor mounted on a human piloted vehicle can create a world representation, which humans may use to enhance situational awareness, but it is the human who does the interpreting and understanding.

Tele-operated vehicles require a human in-the-loop, where there is a heavy reliance upon the human for input and guidance. Thus, the tele-operated vehicle has limited requirements for world representations.

For a system to exhibit autonomy, it must make arrive at its own conclusions and internally direct its actions. A fundamental prerequisite to planning and decision making is representing and understanding one's environment.

4.4.4.1 A Ground Vehicle Perspective

Robotics researchers, especially those focused on ground vehicles, have expended significant energies in developing applicable world representations. Due to the inherent complexity of a UGV's environment, researchers in this field were forced to confront the world representation issue at an early stage in their research. Over the years, this research has yielded solutions that allow UGVs to successfully navigate unstructured terrain, but only terrain of a low to medium level of complexity [Brotan 2007b, Herbert, Kweon, Bellutta, Goldberg, Lacroix, Kelly]. These solutions "represent the world", but they make little effort to "understand" it. In a representation such a 2 ½ D terrain map, all obstacles are equal. The classical terrain map implementation does not differentiate between true physical obstacles such as stones and concrete, and soft obstacles such as grass or shrubs.

Achieving "understanding" via learning is a new and emerging field of research. Understanding via learning allows a robot to learn from experience, thus it adapts to changes. Specific research has focused on adapting to environmental changes for unmanned ground vehicles. The Stanley robot, from Stanford University, uses probabilistic learning to enable high-speed operation in unstructured terrain [Thrun]. Other researchers have investigated how to differentiate between vegetation and true obstacles [Macedo]. Finally, NASA researchers have developed algorithms that predict wheel slippage from visual information [Angelova].

4.4.4.2 Understanding from the Machine or Computer Vision Perspective

Scene understanding remains one of the most difficult problems for machines to overcome. An example would be automatic target recognition (ATR) where machines are good at looking for a precisely defined object, but can not recognize a general class of objects (i.e. they can not pick out the cup on the table unless the cup is precisely defined).

4.4.5 Single and Multiple Platform Systems

Complexity, automation, and autonomy appear within single as well as multiple platforms. In this respect, either system may be preferable depending on the mission requirements. A problem that would be well served by a single-asset solution could be identified for instance by the following characteristics:

- 1) Hard to separate into pieces:
 - Highly interdependent system dynamics.
- 2) Physical dispersion adds little benefit:
 - Simultaneous actions add little; and
 - Sequential tasking is adequate/optimal.
- 3) Information transfer is costly/inadequate:
 - Threats make communication undesirable;
 - Geographic separation makes communication difficult;
 - Terrain/environment make communication difficult; and
 - Time lags and latent data compromise stability or optimality.

A problem that would be well served by a multi-asset solution, on the other hand could be characterized by:

- 1) Easy to separate into pieces:
 - Dynamics are loosely coupled; and
 - Time-scale separation is apparent.

- 2) Physical dispersion can be used to great effect:
 - Simultaneous tasking has great utility; and
 - Sequential tasking is inadequate.
- 3) Information transfer is not costly:
 - A global information state can be maintained;
 - Local information is adequate; and
 - Lags and latency are acceptable.

All this with the caveat that the complexity of some problems is so overwhelming that separation is the only realistic option available. The benefits of having multiple assets add degrees of freedom to the solution of a problem, e.g. allowing a choice of vehicles to service a target. However, the flexibility comes with additional complexity that is imposed in the form of constraints, e.g. a target must be confirmed before attack, and attacked before battle damage is assessed. So, the meaning of “complexity and automation” for multiplatform systems perhaps implies different concepts from those associated with single platform systems.

Other key factors that make a multi-asset solution different from a single-asset solution are:

- 1) Problem division; and
- 2) Information availability.

The former includes actions/items such as Order of precedence (Kill chain), Coupling of tasks, Performance, Computations. The latter deals primarily with Communication, Centralization of processing, Correlation of targets, and Moving targets.

4.4.6 Co-operative Control Operations

In this section the challenges of cooperative control operations are outlined. To fix ideas, the discussion is focused on a simplified surveillance scenario. A team consisting of a flight of Unmanned Air Vehicles (UAVs) is tasked with searching for and recognizing multiple targets in a battle space with many false targets. Allowing for an unstructured environment in cooperative control operations is essential. The presence of false targets/clutter forces one to consider a probabilistic setting. While the dynamics of unmanned air vehicles and, in particular small air vehicles are important, considerations relevant to resource allocation and optimization, the prevailing information pattern and unstructured environment/uncertainty dominate the analysis. The potential benefits of cooperative team actions, and, in general, the benefits of cooperation among distributed controlled objects are addressed.

4.4.6.1 A Taxonomy of Teams

A team is here defined as a loose collection of spatially distributed controlled objects, a.k.a., Unmanned Air Vehicles (UAVs), that have some objectives in common [Ho et al., 1972; Marschak, 1972]. Air vehicles may be too restrictive a term – generically, a team consists of members, or agents, and the team can (and generally will) include humans as operators, task performers (think of target recognition), and/or supervisors. The presence of a common objective forges a team and induces cooperative behavior. If the air vehicles are working together to achieve a common objective, then they are considered cooperative. Different degrees are possible: coordinated; cooperative; and collaborative. At the same time, additional individual objectives of the team members can encourage team members to opt for a weak degree of non-cooperative, competitive, or adversarial action.

When team decision and control problems are discussed, it is important to address the unstructured environment/uncertainty, the organizational structure, the information pattern, and task coupling. Individual operational scenarios can be dominated by one of the above, but will contain elements of all of them. The interaction of these different facets of cooperative control of a team cannot be ignored.

4.4.6.2 Team Coordination

This is the strongest degree of cohesive team action. Consider a set of UAVs that have been designated to be part of the team, and they all share a single team objective and thus strive to optimize a single payoff function. The team could have more than one payoff function that it wishes to optimize, which would then entail multi-objective optimization [Luce, 1989]. Oftentimes, the different payoff functions can be assigned weights and rigorously combined into a single objective function. There is no conflict of interest among the team members, otherwise an incentive scheme [Groves, 1973] would need to be devised. The important distinction here is that particular team members do not have individual objective functions: a team member is a resource that can be over-utilized or under-utilized, if that will best achieve the team objective. The team members are obligated to participate and any assignments, tasking, or agreements are binding; they cannot opt out. At the same time, the forming of coalitions is not possible. The team may be geographically distributed, but it operates as a single unit.

4.4.6.3 Team Cooperation

Each of the team members has a private objective function which he strives to optimize, in addition to the team objective function. The private and team objective functions are weighted such that $0 \leq w \leq 1$. A weight of $w = 1$ on the private objective function means the member acts in its own self interest, in which case there is no team action. A range of $0 \leq w \leq 1$ on the private objective functions corresponds to an increasing level of cooperation of the team members, to where $w = 0$ entails strict coordination. There is a possibility for conflict of interest, but, due to the structure of the objective functions used, it is not generally dominant. In most cases, local objectives such as fuel conservation and survivability are not in conflict with the team objectives and can be jointly achieved. Thus, the multi-criteria optimization aspect of the problem is not dominant and a weighted sum of the objective functions yields a conventional optimization. If they are in conflict, the team objective takes precedence according to the weight used.

4.4.6.4 Team Collaboration

This is a looser form of cooperation. In some cases this can be the result of the task assignment or resource allocation method used [Guo, 2001; Bertsekas, 1992, Bertsekas, 1991; Bertsekas, 1993; Bertsekas, 1993 b]. At the global (team) level, the focus is on task completion, a.k.a., feasibility. Each team member tries to maximize his local objective function consistent with team task completion while avoiding tasking conflicts. This requires that a protocol be designed for negotiation to arbitrate conflicts [Olfati-Saber, 2003; Lamport, 1998]; this connects with the currently developed theory of communication networks. Ideally, those tasks that are best for each member to perform according to their private objective function are tasks that need to be done for the team objective function. In addition, there is the additional implicit constraint that the selected tasks are not redundant. Here there are possibilities of significant conflicts of interest: if the team members have a set of equally valuable (to them) tasks, then likely the conflicts can be resolved (mutually agreeably), and collaboration can be as efficient as coordination. Obviously, the more tightly coupled the various team tasks are, the more difficult it is to achieve a collaborative solution. Strong coupling will occur if a homogeneous team of multi-role UAVs is employed, or if the battle-space is small. A coordinated or cooperative operations approach will be needed. Also, negotiation is not compulsory; the members are not forced to participate. If a solution for a particular team member cannot be found, then this team member can opt out and join an alternative team that has a better overall match of member objective with team objective.

4.4.6.5 Goal Seeking Team Action

This is a further abstraction of a team decision and control problem. Here there are no a priori designated team members. The team is formed from a set of available resources that are loosely networked. Each UAV can simultaneously be a member of several teams. Once a team's objective is achieved, the team will dissolve. The goal in general is abstract and has to be converted to a sequence of intermediate objectives/milestones and the objectives in turn have to be converted into a set or sequence of tasks that are assigned to, and coordinated between, the team members. There might be an intrinsic conflict of interest between the goals of the teams that the UAV can simultaneously be a member of. The teams are therefore self organizing [Jadbabaie, 2003]. This results in a high level of autonomy, where the vehicles are independent agents that however work together in an ad hoc fashion, as needed, to achieve an overarching goal. Each vehicle also strives to optimize its utility function, which may be handled through coordination, as previously mentioned.

4.4.6.6 Non-Cooperative Behavior

To this point we have been describing the different modes of how UAVs interact with other UAVs in a team. In a team, by construction, the objectives are basically compatible.

We now address the control of UAVs that are operating in teams that are competing and, possibly, adversarial. If there are two teams, this is the domain of a significant part of game theory research. This includes strictly competitive zero sum games and also non zero sum games, e.g., bimatrix games [Vorobev, 1977; Luce, 1989]. This is the field of much military research as in the Blue team against the Red team war game. However, the field is much richer than this, because in reality there can also be, e.g., a White team and a Green team. What's more, membership in a team can change fluidly, and the teams can form coalitions which can also dissolve. This is a rich area in which complementary and conflicting interests, collusion, hidden objectives, signalling, diversion, gambits, propaganda, and disinformation play a significant role.

4.4.6.7 Conflict of Interest

Conflict of interest is brought about when the UAVs have different objective functions which, in addition, are structured such that joint action which simultaneously optimizes the different objective functions is not possible or feasible. Thus, consider the performance functionals $J_1(u, v)$ and $J_2(u, v)$ for UAV 1 and UAV 2, respectively; the decision variables of UAV 1 and UAV 2 are u and v , respectively. The payoff for UAV 1 is a function of the actions (or decisions) u taken by UAV 1, as well as the actions (or decisions) v taken by UAV 2. Each affects the value of the other's objective function. If the objective functions were not coupled, that is, $J_1(u)$ and $J_2(v)$, then obviously there would be no conflict of interest. Alternatively, if $J_1(u, v) = J_2(u, v)$, there is no conflict of interest either.

If this is not the case and an ahead of time communicated plan (an agreement) is not in place, then this definitely is a non-cooperative scenario. The question is, what strategy does UAV 1 use to determine the "best" action u to minimize his cost J_1 ? Similarly, UAV 2 is faced with the problem of minimizing his cost J_2 . The actions u^* , v^* are "best" if $J_1(u^*, v^*) \geq J_1(u^*, v)$ and $J_2(u^*, v^*) \geq J_2(u, v^*)$. This means that if UAV 2 deviates from v^* , then UAV 1 will do better than $J_1(u^*, v^*)$ and if UAV 1 deviates from u^* , then UAV 2 will do better than $J_2(u^*, v^*)$. Thus, $J_1(u^*, v^*)$ and $J_2(u^*, v^*)$ constitute guarantees for the respective players, no matter what the other player/vehicle does. This is a Nash (non-cooperative) equilibrium point [Vorobev, 1977; Luce, 1989].

Now consider the definition of "best" where u^* and v^* is the point where no further improvement (reduction) in J_1 can be made without an increase in J_2 , and conversely, no further improvement

(reduction) in J_2 can be made without an increase in J_1 . This is a Pareto (cooperative) equilibrium point [Luce, 1989].

There are two problems associated with the Pareto equilibrium concept. 1. Should UAV 1 choose another action u_- and deviate from u^* whereas UAV 2 sticks to his Pareto choice of v^* , then $J_2(u_-, v^*) > J_2(u^*, v^*)$ and $J_1(u_-, v^*) < J_1(u^*, v^*)$. Now UAV 1 did better, at the expense of UAV 2. 2. In general, Pareto equilibria are not unique and therefore an agreement is needed or side conditions imposed on the actual Pareto equilibrium used.

If both vehicles cooperate and agree to play the Pareto equilibrium solution, then they both can do better than the outcome provided by the Nash equilibrium. This is the benefit of cooperation. However, the “agreement” to play Pareto must be rigidly enforced; else one side can choose an action that results in a one sided benefit to one of the teams, at the expense of the other. If the “agreement” cannot be rigidly enforced, then the players are better off playing Nash, because at least they’ll get the guarantee. The latter can be computed ahead of time, before the game is ever played. One might then decide whether it’s at all worth playing the game.

In the context of human behavior, Pareto games provide an incentive to cheat [Deissenberg, 2001]. Hence, the “contract” must specify a penalty for one of the parties breaking it. If the parties are strictly self interested, an expected cheating value calculation can be made which is a function of [Reward – Penalty * P(caught/cheat)]. Of course, the other party in the agreement can make the same calculation and both could violate the agreement, which means that both parties could end up with less than the Nash (non-cooperative) value. This much studied predicament is referred to as the “prisoners’ dilemma” [Luce, 1989].

It is not at all clear if these considerations have much bearing on UAV team performance, but they abound in human teams. For example, in team sports each of the members share the objective of his team winning the game by accumulating more points than the opposing team. There are a series of plays and roles that are agreed on that could be considered a Pareto solution. However, one of the players might have a different objective that is not revealed upfront. That is, his objective is to maximize the points attributed to him, not necessarily to win the game. His team mates stick to the playbook, he scores more points, he wins (becoming more valuable), and his team loses.

Concerning adversarial behavior in an Unmanned Aerial Vehicles (UAVs) team, consider a network (team) of geographically distributed assets that have a range of overlapping capabilities. These assets service targets as well as provide services to other UAVs. These services have a variety of values and associated costs. Each UAV attempts to provide the highest valued services at the lowest cost [Rasmussen, 2003]. Still, if each of the vehicles, driven by its above stated self interest, engages in the pursuit of maximizing value and minimizing cost, then, under mild assumption and in realistic non-contrived scenarios, this should cause minimal churning and lead to maximizing the value for the team. In the context of UAV teams, situations where the UAVs have very different objectives and consequently self interest at heart, exhibit predatory behavior toward each other, as well as actively practice deception, are not envisaged.

4.4.6.8 Distributed Decision and Control Systems

Figure 4.12 introduced a classification of distributed decision and control systems [Bertsekas, 1989; Smith, 1981]. The “centralized” quadrant represents classical centralized control [Dantzig, 1963; Bertsekas, 1995]. The complete state information from the distributed UAVs is sent to a center where the Decision Maker (DM) resides. No independent action is taken by the UAVs. This control concept can render optimal control action in so far as complex constraints and coupling/interactions between the

vehicles can be properly accounted for and not assumed away. Algorithms used here include dynamic programming [Murphy, 1999; Puterman, 2005, Ross, 1983], large linear programs [Dantzig, 1963] and nonlinear programming [Bertsekas, 1995]. This, in turn, causes centralized control not to scale well due to the curse of dimensionality. In addition, a centralized optimal control structure might suffer from fragility and a lack of robustness to e.g., missing data.

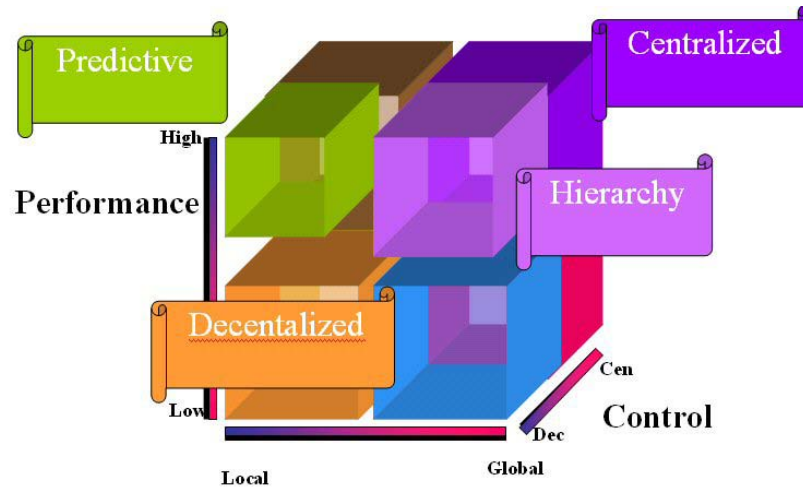


Figure 4.12: Team Decision and Control Metrics.

The “hierarchy” quadrant is where decomposition is prevalent. Motivated by the structure of organizations, hierarchical control [McLain, 2005] schemes are used – see Figure 4.13. The UAVs send the local vehicle state information to a higher level DM. The team members have limited global information, but they send their individual cost functions to the higher level DM. Consequently, an optimal assignment can be made by the DM that is beneficial to the team as a whole. While a degree of robustness is achieved, it is difficult to decompose the control problem and maintain high performance if there is appreciable task coupling. This approach is scalable, but optimality, as with the centralized solution, is not achieved.

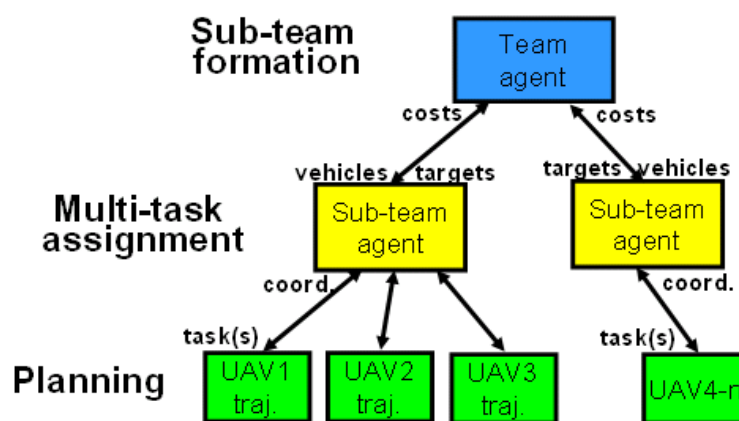


Figure 4.13: Hierarchical Decomposition.

Optimization techniques used here include network flow programming [Ford, 1962; Nygard, 2001; Bertsekas, 1992, Bertsekas, 1993, Goldberg, 1990], mixed integer linear programming [Schumacher, 2007; Richards, 2002; Alighanbardi, 2003; Schumacher, 2004; Richards, 2002], constraint satisfaction

[Modi, 2002; Yokoo, 2000], graph theoretic search algorithms [Nemhauser, 1999], set partition [Balas, 1976], the relative benefit method [Rasmussen, 2002], iterative auction [Bertsekas, 1992; Bertsekas, 1991; Wein, 1995; Bertsekas, 1989; Bertsekas, 1993; Kempka, 1991], negotiation, and consensus schemes [Olfati-Saber, 2003], e.g., the Paxos algorithm [Lamport, 1998].

The “decentralized” quadrant represents strongly decentralized control schemes [Bernstein, 2003] which rely on minimal global information. In the limit, there is no communication at all between the vehicles and information is only inferred about the other vehicles objectives by measuring their actions as seen through own-ship sensors. This line of work is often referred to as emergent behavior [Jadbabaie, 2003]. There is an assumption that the simple low level interactions of the vehicles will result in complex, but more importantly, optimal team behavior that meets a team objective. In general, however, this approach leads to a mediocre macro level response [Jadbabaie, 2003]. For example, the team maintains some loose cohesion, and a nominal center of mass trajectory. The team behavior appears to be complex and one marvels at the complexity brought about by a few simple rules. However, the achieved performance here is consistently low, due to the dearth of communication and consequently little or no global information. Some techniques used here include biological analogies/biomimetrics [Passino, 2005], mechanical analogies ala potential fields [Passino, 2005], and parallel auction [Bertsekas, 1991] which might cause churning.

The “predictive” quadrant is the domain of high decentralization and, at the same time, good performance. The individual team members are highly autonomous, capable of independent action, but share a common objective. Some mechanization concepts come from economics: negotiation [Wellman, 1998]; incentive games [Groves, 1973]; consensus voting [Lamport, 1998]; and distributed constraint satisfaction [Modi, 2002]. Since there is little global coordinating information, there is a high dependence on state estimation and predictive models. The better these models are at estimating future states, the higher the overall performance of the team. In general, the more coupled the various tasks the vehicles are to perform, the more complex and time consuming the arbitration to resolve the task conflicts.

4.4.6.9 Complexity in Cooperative Teams

In cooperative teams an interactive decision making process between vehicles takes place, while individual vehicle autonomy is preserved. There is a continuum between centralized and decentralized control. If a fully decentralized team means no communication, then in a cooperative team the minimum level of globally communicated information that allows the desired level of team performance to be achieved, is required.

In general, the performance of cooperative control can be characterized by task coupling, uncertainty, communications delays, and partial information. The interaction of these dimensions renders cooperative optimal control a complex problem. Currently there is no working theory of cooperative systems that takes into account all these dimensions. A hierarchical decomposition is normally tried to reduce the problem to more digestible bits, but optimality is forfeited in the process. Some degree of coupling is ignored to achieve decomposition. This results in a suboptimal solution which is traded for solvability and some degree of robustness. Indeed, many times robustness comes at the expense of optimality, and vice versa. Indeed, the optimal operating point might be sensitive to changes in the problem parameters.

Team control and optimization problems are decomposed in space, time, or along function lines. Forming of sub-teams of UAVs and tasks can also be done by graph theoretic methods [Papadimitriou, 1982], set partition approaches [Balas, 1976], and relative benefit optimization techniques [Rasmussen, 2002], as well as by brute force search [Rasmussen, 2003]. The sub-team optimization problem (see Figure 4.13) then reduces to the multiple assignment problem; determine the task sequence and timing, for each team member, that satisfies all the constraints while minimizing an overall team objective function. The individual vehicles then perform their own task planning and send coordinating information,

preferably a sufficient statistic, around the network or to a team leader. Algorithms for constrained multiple task assignment include: heuristic search, e.g., branch and bound, Tabu search, or genetic algorithms [Balas, 1996]; generalized assignment [Burkard, 1998]; linear programming [Dantzig, 1963]; iterative network flow [Bertsekas, 1992; Goldberg, 1990]; and iterative auction [Bertsekas, 1988; Kempka, 1991]. One of the primary contributors to the complexity of multiple assignment is task coupling in the face of floating timing constraints – the latter brings in aspects of job shop flow optimization, or scheduling.

4.4.6.10 Task Coupling

UAV team missions such as suppression of enemy air defences and wide area search and destroy are dominated by coupled tasks with floating timing constraints. There are a number of issues involved in solving the multiple assignment problem in a cooperative framework. Chief among these is the ability to decouple assignment from path planning for specific tasks. This means that tasks and path plans are generated to determine costs that are then used in the assignment process. The assumption is that these calculations are still valid after the assignment is made. This is even more so for tour sequences. Unless all possible tours are generated, sub-optimality is being ignored when chaining together tasks.

Also, decoupling assignment from timing is often done. For example, task tours are assigned first. Then the task order and precedence are enforced. This can be done myopically to set the task time using the earliest task that needs to be done, then the next, etc.; or the task times can be negotiated between the vehicles until a set of task times is arrived at that satisfies all the timing constraints. The assumption here is that these task times will not have a different, closer to optimal assignment. The decomposition assumptions to address coupling may lead to infeasibility, where all the tasks can not be assigned, as well as a significant degree of sub-optimality a.k.a., poor performance. If the task coupling is strong, decentralization is a bad idea [Rasmussen, 2003] – optimality is sacrificed, the algorithm might induce churning, and worse, feasibility is not enforced.

4.4.6.11 Uncertainty

Some cooperative team problems can be dominated by uncertainty rather than by task coupling. This is true for those missions where the target identification, target localization, threat identification, and threat location are not known in advance. Some of this information may be known, while the rest is estimated using a priori given probability distributions. The challenge is to calculate the expected future value of a decision or action taken now. For example, if the UAVs use their resources on targets now, there may be no reserves left for targets that are found later and that have even higher value [Girard, 2006; Freeman; Ross, 1983]. At the same time, actions taken now might decrease the level of uncertainty. The latter can be gauged using information theoretic concepts. Possible choices are to myopically follow the decision path of least risk, or follow the decision path that maximizes the possible options in the future. Of course, the safest and middle of the road decisions are not generally the best. Furthermore, one source of uncertainty is associated with the actions of an adversary in response to an action taken by the UAVs. Possible approaches to account for uncertainty are stochastic dynamic programming [Puterman, 2005; Ross, 1983], Markov decision processes [Lovejoy, 1991; Puterman, 1994], Bayesian belief networks [Jensen, 1996], information theory [Gallager, 1968], and, in the case of no information – game theory [Vorobev, 1977; Luce, 1989].

4.4.6.12 Communication

The basic premise of cooperative control is that the UAVs can communicate whenever and as much as they need to. All networks incur link delays, and if these delays are sufficiently long compared to the time between events (see Figure 4.14), they can completely nullify the benefits of team cooperation (cooperative control). Recall that control system delays in the feedback path are conducive to instability.

A critical choice here is whether the team decisions are synchronous or asynchronous. Synchronous implies that there is a common (and up to date) data base accessible to all the vehicles. If an event occurs locally, the event and all associated information is shared across the network and a decision based on the new event cannot occur until this happens. Under this protocol, a slow actor can slow down the whole team and compromise time critical tasks. Strategies are needed to maintain team coherence and performance. Asynchronous decision protocols, while more robust to delays, are much more difficult to verify in order to prove correct operation, are susceptible to inconsistent information across the network, and can lead to decision cycling/churning and, what’s worse – infeasibility. The higher the rate of occurrence of events, the more difficult these problems become because the input’s frequency exceeds the system’s “bandwidth”. Some useful protocols include consensus voting [Olfati-Saber, 2003], parallel computing [Bertsekas, 1989; Bertsekas, 1995], load balancing [Bertsekas, 1989], job shop scheduling [Sycara, 1996], and contract nets [Smith, 1981; Sandholm, 1993]. However, with false information and sufficient delays, consensus may never be reached. Indeed, false information strongly negates the benefits of cooperative control. The situation is somewhat analogous to the feedback control situation: feedback action is superior to open loop control, provided the signal to noise ratio in the measurement is high. One then takes advantage of the benefit of feedback. If however the measurements are very noisy, one might be better off ignoring the measurements and instead opt for open loop (feed-forward or model-based) control.

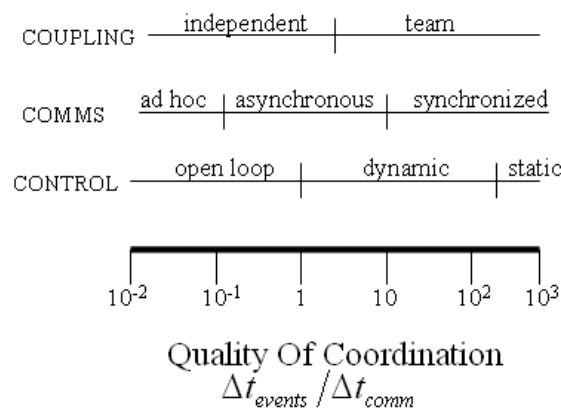


Figure 4.14: Notional Coordination Metric.

4.4.6.13 Partial Information

Decentralized control [Kuhn, 1953] and cooperative teams are characterized by limited or partial information. The full information state is not available anywhere in the network. Worse, the information pattern is not nested. The challenge is to perform the tasks cooperatively and achieve a degree of optimality under limited and distributed information.

A specified level of team performance and team coherence requires a minimum level of shared information, e.g., the team objective function, a subset of the relevant events – e.g., pop up target information, and part of the state vector, e.g., the fuel state of the UAVs. There should be sufficient information to ensure that all the tasks are accounted for and the tasks are consistent.

This can be considered the “sufficient statistic”. Also, the vehicles may have state estimation and prediction models to provide information that is not available locally. Interestingly, the vehicles may have different objective functions as well as inconsistent and delayed information that can result in conflicts that need arbitration or negotiation. False information (see Figure 4.15), in particular, can induce poor performance in a cooperative team – one may be better off using non-cooperative control.

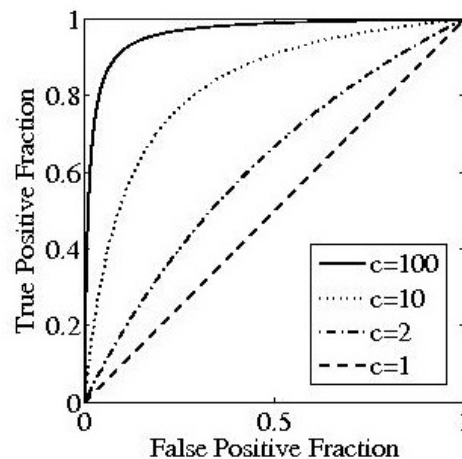


Figure 4.15: Family of Receiver Operating Characteristic (ROC).

For a fixed parameter c in Figure 4.15, the operating point on the ROC determines both the probability of correct target classification, and the probability of a false positive. The parameter is set by the flight condition and the operating point is determined by the threshold setting in the sensor.

4.4.6.14 Operator

Fundamental to the field of cooperative control is the level of realized team autonomy. A continuum of team autonomy levels is possible, from completely autonomous action from release, to human operator management by exception, to human operator management by permission – the latter being the least autonomous. These different autonomy levels can exist as a function of engagement phase or current task, and dynamically change as a function of system state. Furthermore, an autonomous team is a network of distributed functionality, where for example an operator could provide the critical target recognition functionality (difficult to accomplish “by machine”), interface to outside systems, and/or provide supervisor functions. Autonomous target recognition is a long range goal and it therefore makes sense to assign the object classification task to a human operator.

In the context of stochastic cooperative control, the operator can introduce significant uncertainty. Chief among these are operator errors, e.g., object classification errors, and delays. If the team consists of a number of vehicles all sending sensor streams to an operator, the operator’s workload can be high, which would increase the operator’s delay and error rate. For time critical UAV operations a high degree of automation of team decision and control is required. To this end, a model of the human operator’s performance is needed [Pew, 1998], e.g., the probability distribution function of the operator delay or the operator’s error rate – as in a Poisson process model [Parzen, 1960]

Close coordination of a UAV team, including the operator, can be maintained in spite of significant communication delays and processing delays associated with operator cognition phenomenology, operator workload, and operator errors. These operator characteristics must be captured in a stochastic model and incorporated into a controller obtained by solving a stochastic program. This enables robust team coordination to be maintained despite a significant level of uncertainty brought about by operator misclassification of objects of interest, as well as delays.

4.4.6.15 Adversary Action

Much of the research done to date on cooperative teams has been with passive targets, not allowing for intelligent adversary action.

Although threats were incorporated into simulation models, the targets did not react to actions taken by the UAV team in the work described herein. While in some simulation studies threats do react to UAV actions, behavioral/reactive modelling is routinely used, namely, the threats' reactions are pre-programmed and the strategy is known ahead of time to the UAV team. For example, should a UAV penetrate a missile engagement zone, it will be fired upon. For the known missile emplacements, UAV trajectories are planned around the zone ahead of time. If there is an approximate distribution of threats, then UAV trajectories are planned based on the threat expected locations. Unplanned for threats encountered by the UAV trigger a fixed set of UAV responses, such as evasion, or a deviation around the new threat is replanned. Addressing adversary action on the fly is fundamentally different, in that the intelligent adversary observes the state of the engagement, is cognizant of the attacker's capabilities, and solves for his optimal strategy. Here a UAV or team of UAVs must derive a strategy based on the possible actions the adversary may take for every action that the UAV team may take. This two sided optimization problem explodes exponentially for increasing sequences of action-reaction "moves". This can be cast as a dynamic programming problem if all the possible action/reaction pairs are known. Such two sided optimization problems are solvable offline for a modest number of stages, a.k.a., a few moves deep, so that the computed optimal control strategy can be implemented online.

Making the situation even more complex is the possibility of adversary deception such as the employment of decoys, diversions, traps, and false information. For example, the employment of decoys complicates the operator's classification task and thus causes an increase in the rate of false positives. This, in turn, depletes the team's resources, adversely affecting the team's performance.

4.4.6.16 Algorithms for Cooperative Control

Following is a partial list of algorithms that could be used for cooperative control of a team of UAVs. We refer to the relative benefit optimization method [Rasmussen, 2002], network flow [Nygard, 2001], iterative network flow [Rasmussen, 2003; Goldberg, 1990], iterative auction [Bertsekas, 1992; Bertsekas, 1998], decomposition (assign, then time) [Rasmussen, 2003], parallel auction [Wein, 1990], combinatorial or bundle auction [Guo, 2001, Mixed Integer Linear Programming (MILP) [Schumacher, 2007; Richards, 2002; Schumacher, 2004; Richards, 2002], Partially Observable Markov Decision Processes (POMDP) [Lovejoy, 1991], Bayesian belief networks [Jensen, 1996], Genetic Algorithm or generalized search [Rasmussen, 2003; Passino, 2005], centralized or distributed constraint satisfaction or optimization [Yokoo, 2000], stochastic dynamic programming [Puterman, 2005; Ross, 1983], job shop scheduling [Sycara, 1996], vehicle routing [Toth, 2002; Reinelt, 1994], parallel computing [Bertsekas, 1989], voting or consensus [Olfati-Saber, 2003; Lamport, 1998], contract nets [Smith, 1981], games [Vorobev, 1977; Luce, 1989], receding horizon to periodically reevaluate the strategy [Cassandras, 2002; Mayne, 1990], and multiple agent systems [Wellman, 1998].

While not exhaustive, this is a good cross section of the available options for cooperative control algorithms synthesis. For strong task coupling, the centralized algorithms such as MILP and dynamic programming can give optimal solutions for problems that are computationally tractable. For scenarios with weak task coupling, network flow and auction protocols are reasonable approaches, while for scenarios with an intermediate level of task coupling one can call on iterative methods, including relative benefit. Many forms of uncertainty can be accounted for using stochastic dynamic programming, POMDP, or Bayesian belief networks. Some of these can also account for various levels of coupling, but they are not readily decomposable. More strongly decentralized approaches such as distributed constraint satisfaction and parallel auction cannot easily capture coupling among different tasks. Asynchronous parallel computing may address some of the communication issues previously discussed.

In summary, there is no one approach that addresses all the manifold facets of cooperative control. The development of heuristic methods and iterative schemes geared to addressing the dominant traits and the specifics of a particular operational scenario is required.

4.4.6.17 Coupling of Tasks vs. Decentralization vs. Communication

Each of the distributed decision and control approaches or algorithms discussed, as applied to the cooperative team problem, has certain advantages which can be brought to bear on the team problem at hand. A notional trade space is shown in Figure 4.16, where strength of task coupling, communications volume/bandwidth, and the degree of decentralization feature. The corners of the cube, labeled 1 – 8, show those approaches that appear to be more suitable to address the problem characteristics represented by that node. The origin, or node 3, is typified by: weak task coupling or few constraints; full information, that is, a centralized or nested information pattern; and very low communication costs – high bandwidth, near zero delays, and complete connectivity. The single assignment optimization problem with binary decision variables and simple constraints can be posed as a network flow problem. This class of integer programs can be solved using slow relaxation techniques, or fast specialized binary tree search algorithms. The control can be partially distributed through implicit team coordination, which is where the centralized solution is computed, redundantly, by each of the UAVs.

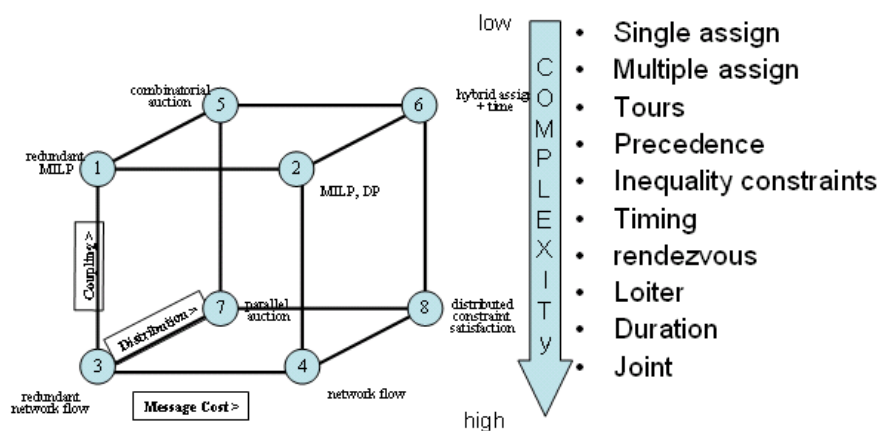


Figure 4.16: Coupling-Decentralization-Communication Trade Space.

As the cost of sending messages increase, the need for redundancy could motivate us to use the centralized network flow approach shown at node 4. Concerning node 3: as the tasks become more strongly coupled and, particularly, when floating timing constraints are included, the network flow solution is not suitable. MILP can rigorously include both integer and continuous constraints. While a full information (centralized) scheme, the same approach can be used to have the solution computed redundantly by each of the vehicles, as at node 1. As the cost of sending messages increases, the need for a degree of redundancy leads one to use the centralized MILP algorithm, which could also be formulated as a dynamic program at node 2. Receding horizon optimization can reduce the computational burden, enhancing scalability, and also reduce the need for information about the environment, but at the expense of optimality.

One objective of cooperative teams is to distribute the control so that the UAVs can operate more autonomously, but cooperate in performing team tasks. Decentralization increases in moving from node 3 to node 7. The Jacobi auction [Kempa, 1991] iterative algorithm, while intuitively appealing, is, in its basic form, mathematically equivalent to the network flow optimization problem. This iterative optimization scheme does not require full information locally, but it does require a centralized auctioneer with an attendant increase in the message load. Continuing to node 7, the centralized auctioneer is eliminated in the asynchronous parallel auction scheme. Again, this is feasible because only the simple constraints (coupling) of network flow are active, and the very low message cost does not incur a penalty on iteration (bidding). The full benefits of parallel computing are not achievable, since targets cannot be processors.

As the task coupling strength increases at node 7, the simple constraints easily accommodated using parallel auction based algorithms, are no longer appropriate. Combinatorial auction at node 5, where bundles of goods are traded, is a mechanism by which more extensive task constraints can be included: coupled tasks are bundled together during the UAVs bidding phase. In the worst case, every feasible permutation could be a separate object to bid on. Preparing such bids may require full information. Timing constraints cannot be readily incorporated, and combinatorial auctions cannot be directly layered on top of a parallel auction. From node 7, as communication costs increase, parallel auction – based protocols are no longer appropriate, since the latter are contingent on near instant messages. Distributed constraint satisfaction type resource allocation algorithms at node 8 may be preferable because less messages are needed, and they are strongly decentralized. Only simple constraints can be accommodated here, however.

Finally, at node 6, we have the most challenging case of strong coupling, minimal communications, and strong decentralization. No one approach or solution could address these requirements, and, in some cases, no suitable approach currently *exists* which addresses all the requirements, since, in some sense, they are contradictory or mutually exclusive. One practical strategy is to apply a network flow, auction, or parallel auction algorithm iteratively for a fixed (receding) horizon. While not optimal, this allows more complex constraints to be incorporated, including timing constraints. This is a two stage strategy, which partially decouples task assignment from task planning. The minimum deviation task times allow the vehicles to refine their trajectories (or bids) to meet the team task constraints.

4.5 ARTIFICIAL INTELLIGENCE

Historically artificial intelligence (AI) research focuses on the simulation of human-intelligence; today this is called strong AI. Modern AI though is concerned with producing useful machines to automate human tasks requiring intelligent behavior. One area of modern AI is AI robotics which concentrates on bringing the software intelligence to the hardware realization. In practice the machines are developed to work autonomously utilizing the least human interaction possible. In order to help AI robotics, several competitions are held challenging different areas of the research, such as legged robots, wheeled robots, flying robots, all in different sizes and also human like robots to name a few.

The research focuses on developing an autonomous flying robot and since this area is a very new field of research it is necessary to look at what the other fields have achieved.

The closest to the UAV application are the ground robots which are discussed in the literature exhaustively.

4.5.1 Architecture

The first subject to address is the robot architecture which is well described in [Coste-Manière]. Basically in the process of designing an architecture for robotic systems several requirements have to be set, from computational power restrictions through hardware arrangements to modularity for future enhancements.

4.5.2 Uncertainty

The greatest challenge in AI robotics is posed by the operation in unknown environment as described in [Saffiotti]. Industrial robots are capable of highly complex actions but they lack the reaction to changes in the environment; in other words they operate in an environment that is assumed to be completely known to them.

4.5.3 Deliberative Architectures

The early robotics scientist tried to solve the challenge of handling uncertainty by developing systems that would gather information about (sense) the environment and react to changes in it by planning their sequence of actions accordingly, hence the name deliberative architectures. These systems showed their disadvantage immediately because of the very limited computational power at that time. The moment an event occurred (a deviation from current states and 'planned current' state) the replanning had to be initiated and thus it took a relatively long time to carry out a mission. A sophisticated deliberative architecture is presented in [Saffiotti].

4.5.4 Reactive Architectures

A new wave of research started with the introduction of reactive robotic architectures which lacked the world model and therefore the exhaustive planning and acted 'instinctively'. These robots were impressively fast in carrying out missions but having no model of the environment prevented them from being able to achieve complex tasks.

4.5.5 Three-Layer Architectures

A natural evolution of architectures was to create hybrid systems capable of both reactive and deliberative behaviors. These are the three-layer architectures described in detail in [Gata]. Today most of the architectures are based on the three-layer concept but differ in accordance to what they are designed for.

An excellent overview of existing architectures in respect to an aerial vehicle implementation is presented in [Burridge].

4.5.6 Cognitive Architectures

A different approach originates from the cognitive science. Researchers developed cognitive architectures that emulate human behavior. These are based on observing human behavior and possess a structure accordingly. These systems evolved into highly complex structures and are widely used on different fields of AI research; the most popular being Soar described in [Lehman]. The use of cognitive architectures for aerial combat simulation which is closely related to the problem of coordination of unmanned vehicles is presented in [Jones]. The serious disadvantage of these systems is their need for massive computational power.



Chapter 5 – MISSION MANAGEMENT AND ROBUSTNESS

5.1 INTRODUCTION

Autonomy is often praised as a technology capable of reducing cost and enhancing performance in mission operation and management. Through careful deployment within the overall mission architecture, automation can augment or replace human decision making in order to increase reaction speeds, reduce errors and stress, mitigate cognitive overload, enhance safety, lower costs, focus analysis, cut bandwidth requirements, and free human reasoning for strategic tasks requiring high levels of robustness. Here, system health management is a specific mission operation task that provides the robustness. Through the use of automation and fuzzy algorithms the mission tasks can be enhanced further.

5.1.1 Risk Management

The role of risk assessment and risk management is to continuously identify, analyze, plan, track, control, and communicate the risks associated with an integrated system. Risk in itself, the possibility of suffering a loss, should not be avoided, and rather is essential to progress because failure is often a vital part of learning. Managing risk is the key to success, so that it is beneficial and not detrimental. In the context of software engineering and development, risk can be defined as the possibility of suffering a diminished level of success (loss) within a software-dependent development program. This prospect of loss is such that the application of the selected theories, principles, or techniques may fail to yield the right software products. In terms of the hardware, the risk in failure, unavailability of the hardware due to schedule or unavailability of expert manpower also contributes to risk. Resources available manpower, and schedule delay all contribute to risk. Table 5.1 in basic terms indicates the differences and consequences among risk assessment, risk abatement and risk management. Risk Assessment is an essential element of risk management. Once a risk has been recognized and evaluated as being important, steps must be taken to minimize the risk. Risk abatement is the main strategy for managing risk.

Table 5.1: Risk Assessment, Abatement, and Management

| Risk Assessment | Risk Abatement | Risk Management |
|---|------------------------------------|--|
| What can go wrong? | How we can prevent from happening? | What can be done? |
| What is the likelihood that something could go wrong? | What needs to be done? | What are the available options and tradeoffs? |
| What are the associated consequences? | Cost and schedule changes? | What are the impacts of current decisions to future options? |
| Adverse effects? | What option is the best option? | What is the best option? At what price? |

Figure 5.1 describes typical top-level actions that are common to all methods in conducting risk assessment. This process includes risk category or areas such as cost, schedule, and performance. The process includes a core set of assessment tasks and is related to the other two categories. This requires supportive analysis among areas to ensure the integration of the assessment process.

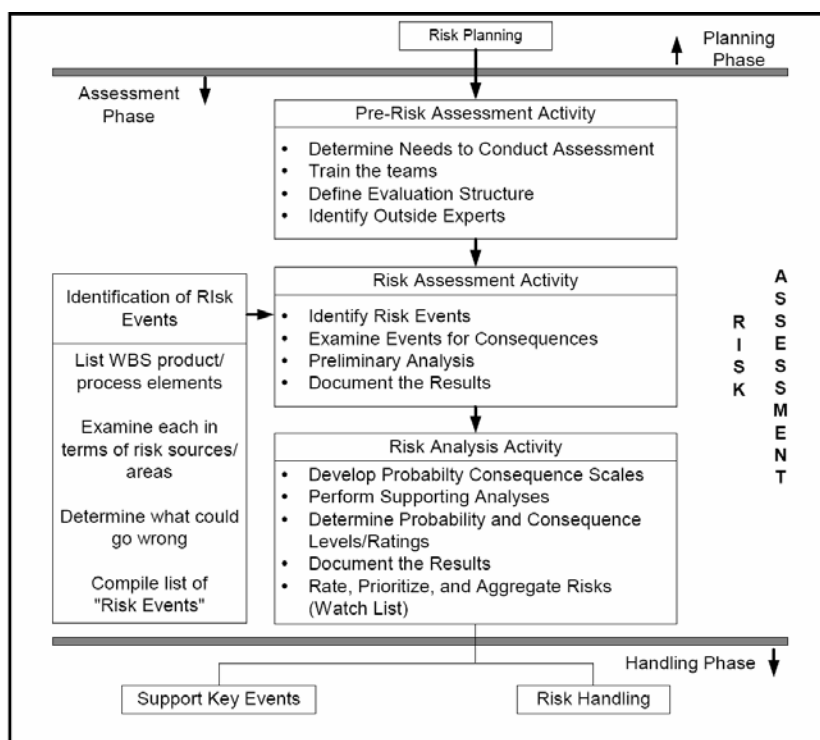


Figure 5.1: General DoD Risk Assessment Process [Anon, Risk Management Guide for DoD Acquisition].

5.1.2 Uncertainty Management

Uncertainty management is a significant long-term foundational issue for planning, design and management of engineering systems. Given the spatial and temporal uncertainty associated with battlespace visualization, uncertainty management must be an integral component of future systems designed to aid commanders and staffs in developing, disseminating, and executing a commander’s concept of the operation. Problems ranging from control of the UAVs, to spacecraft formation right for planetary exploration, to supply chain management are all examples of systems in which networked controls, communications, and computation are playing an increasingly integral role. All of these problems: distributed computation, cooperative planning, and uncertainty management, present challenges for which the existing theory leaves many important questions unaddressed. Current tools for analysis and design of flight control systems are focused on control of individual vehicles or small numbers of coordinated vehicles. While these tools allow sensing and communication channels to be included, there is not yet a deep understanding of the role that information now plays within a networked control system, especially as the number of vehicles increases. There has been some recent progress in this area for networked control systems; but, there has not yet been an exploration of the fundamental limits imposed by sensor and communication topologies (e.g., what is the minimal information needed between sets of vehicles in order to maintain stable formations in the presence of uncertainty). Early work on string stability in longitudinal control of automobile platoons showed that this information now was important and provided insights for the one dimensional case. This situation becomes much more complicated for aerial vehicles operating in dynamic, uncertain and adversarial environments. In the absence of a centralized controller, the Formation Self-Assessment (meaning determining its own properties) is necessary. Examples of self-assessment include health monitoring and role management. When the information flow topologies are themselves uncertain and dynamic, a crucial component of formation self-assessment is the determination of those topologies to enable implementing other distributed algorithms which act along those communication networks. What the above challenges have in common is

that they involve managing the interaction of an intelligent, dynamic, and networked system with a dynamic, uncertain, and adversarial environment in a setting where the means to manage that interaction are constrained by limitations in the flow of information within the network. Accordingly, we have identified the following areas to pursue to in our research. Taken together, these different research goals address aspects of cooperative control of multi-vehicle systems which we consider to be crucial in enabling this technology. Our overarching goal is the implementation of a control architecture and control laws which are capable of achieving mission objectives and reacting intelligently and robustly to the uncertainty in the environment while maintaining the formation’s ability to communicate with itself. We intend to proceed incrementally toward this admittedly ambitious goal, emphasizing the need to clearly articulate a framework through which these questions can be analyzed, and leveraging our research experience in dynamical systems theory and control of individual vehicles wherever possible. In the context of autonomous flight control, the adaptive model-based propulsion control system can deliver the desired thrust response to the vehicle management system where a pilot might otherwise have needed to manually adjust the throttle. This is particularly important for an aircraft with multiple engines, since asymmetric thrust response can result in an unacceptably large yawing moment. Additionally, the sluggish thrust response of a degraded engine would be an uncertainty for an autonomous vehicle management system performing transient maneuvers.

5.1.2.1 Definitions of Uncertainty

Uncertainty is a general concept that reflects our lack of sureness about something or someone, ranging from just short of complete sureness to an almost complete lack of conviction about an outcome. Uncertainties exist on sensor fault, output signal, and models. This uncertainty can be viewed as output multiplicative uncertainty. DeLaurentis and Mavris provide the most fitting definition for our application: “Uncertainty is the incompleteness in knowledge, either in information or context, which causes model-based predictions to differ from reality in a manner described by some probability distribution function.” Uncertainty categorized by various sources and can be described in Table 5.2. Each of these uncertainty types must be considered to maximize the probability of success or robustness. When uncertainties in consequence (performance, cost, and schedule) are calculated in a design uncertainty analysis, the resulting consequence probability distributions are a complete measure of risk.

Table 5.2: Types of Uncertainty [Alan Brown and Timothy Mierzwicki]

| Types of Uncertainty | Description |
|---------------------------|---|
| Inherent Uncertainty | Uncertainty due to the variability inherent in a system design, technology or the environment |
| Statistical Uncertainty | Uncertainty due to the incompleteness of statistical data |
| Modeling Uncertainty | Uncertainty resulting from the simplification of nature and physics |
| Human Uncertainty & Error | Uncertainty due to differences of opinion (or subjective uncertainty) and misdiagnosis (or diagnostic uncertainty) including: errors in calculation; selection of the wrong known data; inadequate design review; failure in calculating critical conditions; poor quality fabrication; use of the wrong materials; and abuse by operators. |

There is a significant need for data integration capabilities in land, air and sea, which has manifested itself as products in the multination defence infrastructure. However, in dealing with flight and engine or perhaps system data it has become apparent that existing data integration products do not handle uncertainties in the data very well. This leads to systems that often produce an explosion of less relevant answers which subsequently leads to a loss of more relevant answers by overloading the operator of that

machine. How to incorporate functionality into data integration systems to properly handle uncertainties and make results more useful has become an important issue. Data uncertainty results from two different sources:

- 1) Inherent data uncertainties are attributes of the data itself and not artifacts of its representation. Data generated from laboratory experimental methods often have inherent uncertainties.
- 2) Data Representation Uncertainties. Data representation uncertainties result from the mapping of real world information onto a computable representation of this information.

In general, uncertainty describes the extent to which a system environment is known and understood. Efficiency points to how well data can be communicated with the systems. Another important issue that must be accounted for in the design of model-based fault diagnosis and fault-tolerant control systems is the presence of model uncertainty (also referred to as the mismatch between the system and the model). Model uncertainty was a key motivation for introducing feedback and that classical control theory had very effective ways of dealing with uncertainty both qualitatively and quantitatively. Process uncertainty could be described very easily as a variation in the process transfer function with the caveat that the disturbances do not change the number of right half plane poles of the system. A good starting point for discussing deviations between model estimates and observations is a conceptual framework.

Let us consider the situation when the parameter set α has a certain set of values [H.R. Oleson]. Let us assume that our model is deterministic and thus purports to predict one number for each α : the ensemble average (\bar{c}) °C for a large number of realisations. If we consider just one realisation, then the observed concentration (C_o) can be decomposed as follows:

$$C_o(\alpha, \beta) = \bar{C}_o(\alpha) + c(\Delta c) + c(\alpha, \beta) \tag{1}$$

Observed
concentration

Ensemble
average

Measurement
error

Inherent
uncertainty

Note that the decomposition depends on the definition of α and thus is model-dependent. This is an annoying fact, which we must accept. The “measurement error” term accounts not only for trivial instrument inaccuracy, but also for the fact that we may not measure the parameter that we assume. The measurement error can in principle be reduced by increasing the number and the accuracy of measurements. A way to avoid unnecessarily severe effects of measurement error is through the use of quality indicators assigned to data, so that misleading observations can be discarded. The key parameter is how we can make a decision based on the data and knowledge gained.



Figure 5.2: From Data to Knowledge.

The disclosed systems and methods are directed to exchange of uncertainty information between interacting modules. Variances in the outputs of one module may be required as uncertainties in the inputs of another module. The disclosed systems and methods allow for simple and efficient communication of the uncertainty information between any modules. In one aspect, a system of integrated modules includes one or more application modules and at least one interface module. The application modules are adapted to receive inputs and/or generate outputs, including uncertainty distribution information. The interface modules are adapted to communicate with one or more application modules and to translate uncertainty information between a predetermined uniform format and an application-specific format.

The key idea is to decompose, respectively, observed concentrations and modeled concentrations. It is a basic assumption that we have a model formulated in terms of a set of input. A good starting point for discussing deviations between model estimates and observations is a conceptual framework, which has been described in several papers.

5.1.2.2 Sources of Uncertainty

Uncertainty sources might be characterized in terms of system level axis. At one end might be uncertainties due to the dynamic model stability and control derivatives. At the other end might be mission type environmental factors like numbers of targets or false targets. Somewhere between these extremes might be damage recovery effectiveness and losses due to enemy defences. Also, it might be notable that uncertainty reduction can be a cause for consideration of coordinated, multi-asset solutions to begin with. Rationales for multi-asset solutions are that:

- 1) There are things that can be done with multiple assets that simply cannot be done with one; and
- 2) There is greater flexibility and inherent fault tolerance with multiple assets.

For example a problem requiring saturation of an enemy defence cannot be done with one asset.

5.2 REVIEW OF CURRENT APPROACHES

The engineering of mission management knowledge based processes for the command of multiple intelligent autonomous vehicles for land, air and sea is the coordination of elements of a distributed system so as to generate coherent behaviour and robustness. As such the techniques apply to the management and control of military command and control. The ever increasing demand for cost effectiveness, project efficiency and increasing productivity resulting from open competition is resulting in greater demand for coherent, systems solutions for complex systems consisting of single or multiple vehicles operating in land, air and sea. These integrated systems are characterised by high capital value and extended life time, which together raise a requirement for evolution in system capability and the acceptance of change as an inherent characteristic of system infrastructure. The acceptance of change implies the need for a rigorous approach to the analysis and design of these systems which emphasises the achievement of modularity. In addition, since these systems often are required to operate in dynamic large, complex, uncertain, unstructured, non-benign environments without human intervention, there is a requirement for an intelligent adaptive ability which can react to environmental dynamics. Adaptive systems are more resistant to system and environmental changes potentially resulting in significant cost saving though increased operational life.

Benjamin Lussier et al in their paper “On Fault Tolerance and Robustness in Autonomous Systems” concerns about the dependability of autonomous systems, notably because of advanced decisional mechanisms and other artificial intelligence techniques on which the systems rely. The paper focuses on two non-exclusive approaches aiming to improve dependability, namely fault tolerance and robustness.

Kemin Zhou in his paper entitled “A New Approach to Robust and Fault Tolerant Control” summarizes a new approach to robust and fault control. This design approach is based on a variation of all controller parameterization in which two separate controllers are deployed: nominal performance controller and a robust controller. The controllers works in such a way that when a component (sensor, actuator, etc...) failure is detected, the controller structure is reconfigured by adding a robustness loop to compensate for the fault. This strategy works under various conditions. The price for achieving such a high performance and robust control is the complexity of the controller.

Ufuk Demirci and Feza Kerestecioglu in their paper entitled: “Fault Tolerant Control with Re-Configuring Sliding-Mode Schemes” presented a controller design method for linear MIMO systems in which a sliding mode controller is reconfigured in case of system faults. Faults are detected with the residual vector generated from a standard linear observer. Once a fault has been detected the fault distribution matrix can be obtained and used to update the corrective or equivalent control parts of the sliding mode controller. As a result, fault tolerant adaptive controllers keep the system performance within acceptable limits or at least avoid the system to wind-up. They have concluded that the second approach for reconfiguring sliding-mode controller which uses the extracted fault or disturbance information for the equivalent control part of the sliding-mode controller gives a better performance.

Balajee Kannan and Lynne E. Parker in their paper entitled, “Metrics for quantifying system performance in intelligent, fault-tolerant multi-robot teams” defined “fault-tolerant system” as any system that has the capability to diagnose and recover from faults. In their paper, they have outlined application independent metrics to measure fault-tolerance within the context of system performance. In addition, outlined potential methods to better interpret the obtained measures towards understanding the capabilities of the implemented system. Furthermore, a main focus of our approach is to capture the effect of intelligence, reasoning, or learning on the effective fault-tolerance of the system, rather than relying purely on traditional redundancy based measures. In this paper, they have presented an evaluation metric to measure the extent of fault-tolerance towards system improvement over a period of time. Furthermore, we evaluate two different multi-robot applications based on the defined metrics. Specifically, the research provides a quantitative measure for identifying system fault-tolerance in terms of efficiency, robustness and the extent of learning. In addition, this paper addressed the problem of developing application independent metrics for calculating the influence of fault tolerance towards system performance and identifies potential methods for analyzing the obtained measures towards evaluating the true capability of a multi-robot system.

Benjamin Lussier, et al have described in their paper entitled, “Fault Tolerance in Autonomous Systems: How and How Much?” In their paper, they address the execution of complex missions in uncertain environments and also address dependability as a whole, but focus specifically on fault tolerance. His paper presents several basic concepts:

- 1) Robustness and fault tolerance, both characterizing the resilience of a system towards particular adverse situations; robustness characterizes resilience towards uncertainties of the environment, and fault tolerance characterizes resilience towards faults affecting the system resources; and
- 2) Decisional mechanisms are central to autonomous systems as they embody the ability to dynamically select appropriate actions to achieve specific objectives; they are composed of knowledge specific to a domain of application and an inference mechanism used to solve problems.

They have summarized the state of the art in dependability and robustness mechanisms used in autonomous systems, and some conclusions that we drew from it. In particular:

- 1) Fault avoidance is largely privileged compared to other dependability means, although it rarely appears to be implemented intensively enough.

- 2) Development faults are hardly addressed by fault tolerant mechanisms in autonomous systems; robustness techniques somewhat compensate this problem, but are surely insufficient for critical applications.

Review of Diagnosis and Fault-Tolerant Control book by Blanke et. al covers several model-based failure detection techniques. Although failure detection is a well-defined challenge, the choice of system modelling technique can lead to quite different mathematical problems. A large portion of the book is devoted to presenting alternative modelling techniques. Chapter 1 introduces the fundamental problems of failure detection and fault tolerant control and presents an overview of the ideas behind the methods presented in later chapters. Chapter 1 also provides comprehensive definitions of terminology for use throughout the book. What is missing in this chapter (and in the rest of the book), however, are examples of applications of the methods covered. It would have been useful to give real-world examples that utilize the methods presented; for example, method A is used in the brake system of car X, a variation of method B is used in the autopilot of airplane Y, or method C will be used in a future space station. On the other hand, Chapter 2 presents two examples used throughout the book and, although these applications cannot be considered industrial, they are sufficiently complex to illustrate some problems with implementation of the methods. The first example concerns fluid-level control in a two tank system while the second example concerns steering control of a ship based on a simple model of the ship dynamics. A review of some dynamical system models is given in Chapter 3. In particular, state-space continuous time, discrete-time, discrete event, and hybrid systems are briefly discussed. Chapter 4 examines the application of some graphical analysis techniques to the study of fault propagation in component-based system architectures. The content of this chapter, however, seems unrelated to the rest of the book. Chapter 5 considers continuous time models and begins by deriving an interesting algorithm for finding redundancy among observed or known system variables that usually correspond to system inputs and outputs (u , y). The application of this algorithm to failure detection, however, is questionable because the redundancy relations involve first- and higher-order derivatives of u and y , information that cannot be obtained in many cases due to observation noise. The main results concerning failure detection techniques are presented in Chapter 6. This chapter reviews techniques based on analytical redundancy, in particular, the classical two-stage detection method for dynamical systems subject to additive noise. The stochastic setting with additive noise modelled as a stochastic white process, a two-stage detector was developed; the first stage consisted of a Kalman-based whitening filter generating a residual, and the second stage consisted of a canonical statistical test for deciding whether the residual had the expected statistical properties such as whiteness. Many variants and extensions of the two-stage approach have been studied in the literature and some of these extensions are presented in this chapter. In some cases, algorithms for implementing the techniques are also given and several examples are used to illustrate their application. Chapter 7 examines the fault-tolerant control problem. It is assumed that a finite number of faults is possible, and, for each fault, the dynamic behaviour of the system is modelled as a continuous-time dynamical system. Switching between systems is modelled by an automaton, and the notions of passive and active fault tolerant control are introduced. Passive control is closely related to classical robust control in that behavioural changes due to failure are viewed as model uncertainty and are taken into account in the design of the controller. Active control, on the other hand, is based on controller reconfiguration subsequent to failure detection and identification. Two techniques are investigated, the first being an optimal control approach, which is considered only in the simple case of linear system, state feedback, and quadratic cost. The proposed method does not seem realistic for applications. The second technique for fault-tolerant control is the virtual sensor-actuator. In the case of a sensor, the idea behind this technique is that, if a sensor fails, its output can be reconstructed from the other sensors, and the original controller can be used with the reconstructed observation. An analysis is presented for the limited case of static output feedback control. The applicability of this idea to the general dynamic feedback case is questionable; even stability properties do not seem to be preserved in the general case. The only general-purpose technique presented seems to be “the common sense approach,” which involves designing a controller for each model and switching the controller after a change of model has been detected. Chapter 8 explores a stochastic modelling technique for systems under surveillance. This technique uses a

stochastic automaton with the Markov property, which is usually called a controlled Markov chain. Surprisingly, the authors never refer to the literature on Markov chains or, in particular, to the literature on the application of Markov chains to failure detection. Some elementary properties of Markov chains are derived in Chapter 8, and applications to the failure detection problem are discussed. The idea of a virtual sensor is also examined in this context. In Chapter 9, the authors explore the supervision of a system where only quantized measurements of the inputs and outputs are available. This system is modelled as a stochastic automaton. Although this automaton does not have the Markov property, methods introduced in Chapter 8 are used for consideration of approximate models. There is no discussion, however, concerning stability. Finally, some of the methods discussed in earlier chapters are applied to examples introduced earlier.

Review of the paper of Optimally Robust Redundancy Relations for Failure Detection in Uncertain Systems by Xi-Cheng Lou, et. al, indicates that the robustness of the failure detection process consequently depends to a great degree on the reliability of the redundancy relations, which in turn is affected by the inevitable presence of model uncertainties. A wide variety of techniques has been proposed in recent years for the detection, isolation, and accommodation of failures in dynamic systems. In one way or another, all of these methods involve the generation of signals that are accentuated by the presence of particular failures if these failures have actually occurred. The procedures for generating these signals in – turn depend on models relating the measured variables. Consequently, if any errors in these models have effects on the observables that are at all like the effects of any of the failure modes, then these model errors may also accentuate the signals. This leads us directly to the issue of robust failure detection, that is, the design of a system that is maximally sensitive to the effects of failures and minimally sensitive to model errors. The authors indicate that all failure detection methods are based, either explicitly or implicitly, on the use of redundancy, i.e. on (possibly dynamic) relations among the measured variables. The paper addresses the problem of determining a redundancy relation that are optimally robust, in a sense that includes several major issues of importance in practical failure detection, and that provides a significant amount of intuition concerning the geometry of robust failure detection. The paper provides a procedure, involving the construction of a single matrix and its singular value decomposition, for the determination of a complete sequence of redundancy relations, ordered in terms of their level of robustness. This procedure also provides the basis for comparing levels of robustness in redundancy provided by different sets of sensors.

5.3 FAULT DETECTION AND ISOLATION

There are many methods used today to defeat fault. Fault diagnosis and fault tolerant control have become critically important in modern complex systems such as aircrafts. The basic concept of Fault Detection and Isolation is the ability of a system to diagnose the effect, cause, severity and nature of abnormal behaviour (i.e. faults and failures) in its components. The concept of Fault Tolerant Control is a closed-loop control system that tolerates component malfunctions while maintaining a desired degree of performance and stability. Figure 5.3 shows the possible location of faults that can enter into the control system.

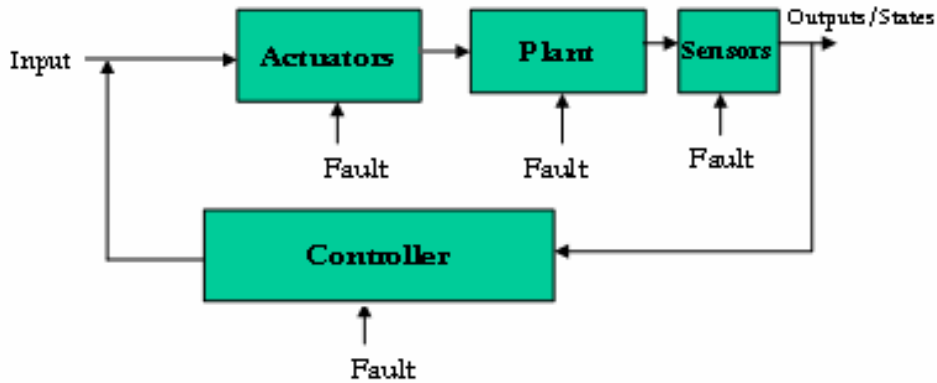


Figure 5.3: Location of Potential Fault in the Control System.

5.3.1 Review on Fault Tolerance

There are many techniques for Fault Tolerance. These techniques are: fault intolerance, fault detection and reconfiguration, and fault masking and reconfiguration. The first question comes to mind is: What is a Fault? Fault is different than error and failure. The definition of these is shown below:

Fault: An incorrect state of hardware or software resulting from failures of components, physical interference from the environment, operator error, or incorrect design.

Error: The manifestation of a fault.

Failure: A result of a delivered service deviating from the specified service caused by an error or fault.

5.3.1.1 Fault Classifications

Based on Jaynarayan Lala and Richard Harper, fault can be classified various classification methods. This is illustrated in Table 5.3.

Table 5.3: Fault Classification

| Classification Method | Types of Faults Involved |
|-----------------------|--|
| By Nature | Accidental Faults, Intentional Faults |
| By Origin | Physical Faults, Human-Made Faults, Internal/External, Design Faults, Operational Faults |
| By Persistence | Permanent Faults, Transient Faults, Intermittent Faults |

Perhaps the best method to classify fault was suggested by D. P. Siewiorek and R. S. Swarz, in which the fault is based on its origin, or whether it is permanent in nature or not. This classification is shown in the following Figure 5.4.

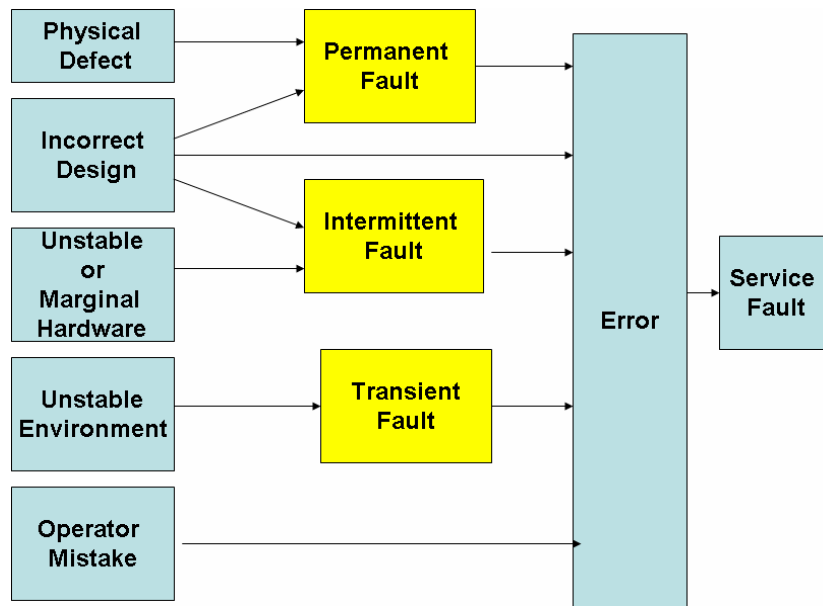


Figure 5.4: Fault Classifications.

5.3.1.2 How to Defeat Faults?

There are two distinct methods on how to defeat faults. These methods are:

- 1) *Fault Intolerance/Prevention Methods*
- 2) *Fault Tolerant Methods*
 - Redundancy
 - Fault Detection and Reconfiguration
 - Fault Masking
 - Software Fault Tolerance

Both methods are used for control system design today. There are various fault tolerance taxonomies. This is how Siewiorek and Swarz categorize fault tolerance techniques, as shown in Figure 5.5. There are other ways to categorize the methods, such as redundancy management systems. Fault tolerance is sometimes also called redundancy management. Mature systems all have ability to dynamically reconfigure after a fault is detected. That’s true for fault detection systems as well as masking systems. (Redundancy Management).

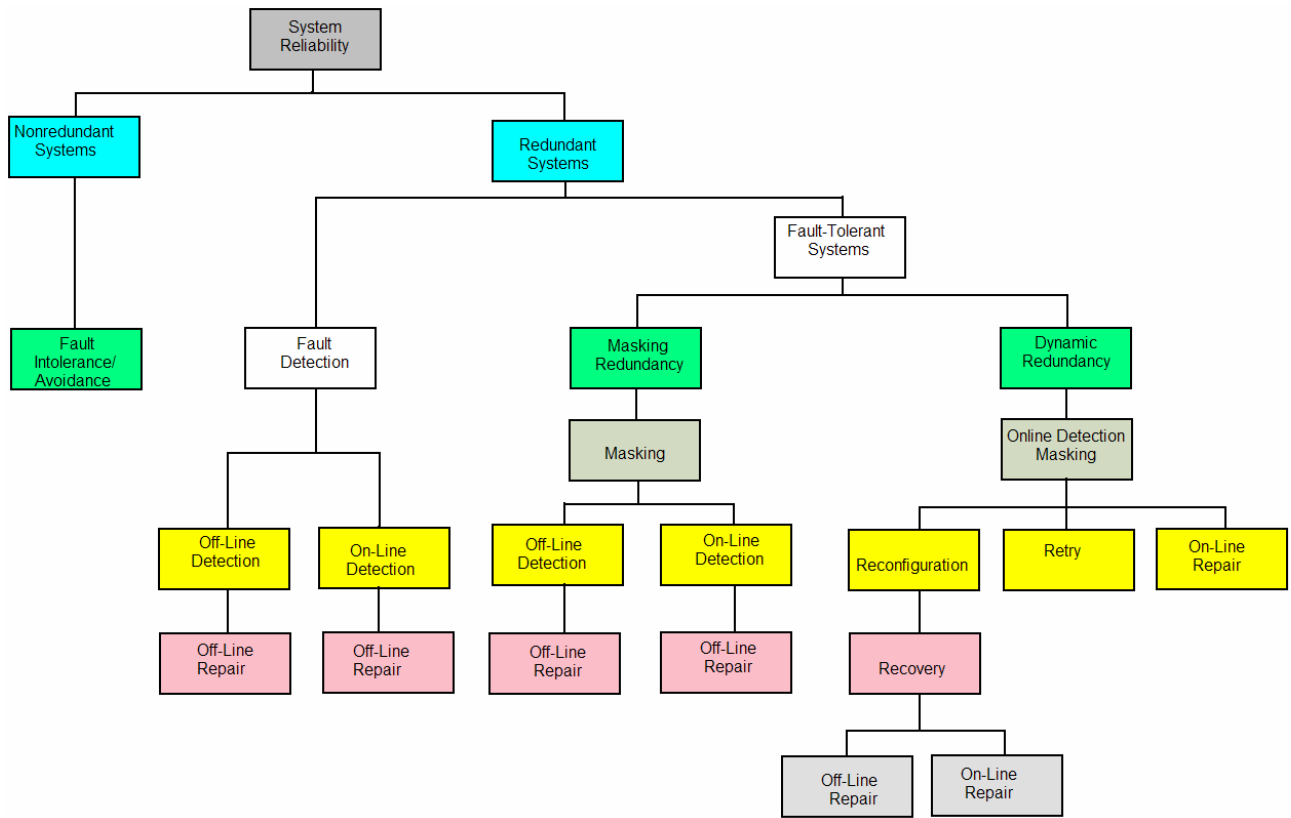
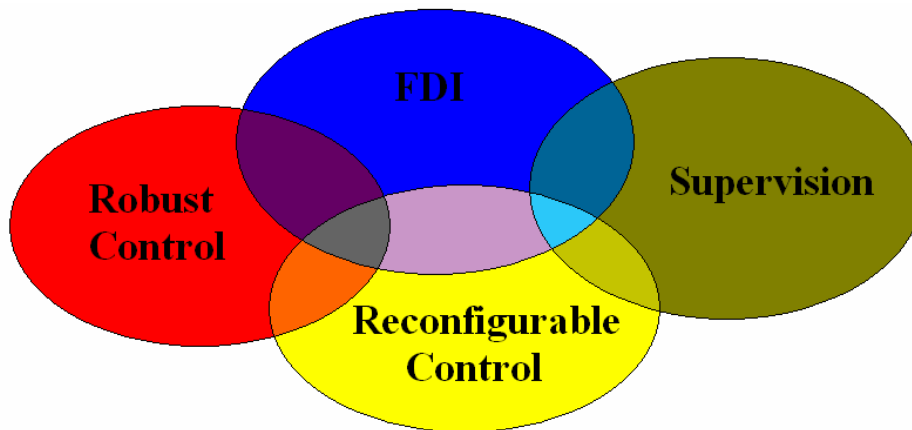


Figure 5.5: Taxonomy of System-Failure Response Strategies.

In a paper by Patton, a graphical presentation of Fault Detection and Identification (now Isolation) (FDI) for a robust control and reconfigurable control is shown in Figure 5.6. Here, the scattered areas of research are shown towards a robust, fault-tolerant system of the future.



Patton, R.J. *Fault Tolerant Control Systems: the 1997 Situation SAFEPROCESS'97*

Figure 5.6: Graphical Presentation of Fault Detection and Identification.

Since no system in the real world can work perfectly all the time under all conditions, it is critical to be able to detect and identify the possible faults in the system as early as possible so that measures can be taken to prevent significant performance degradation or damages to the system. In the last twenty some

years, fault diagnosis of dynamic systems has received much attention and significant progress has been made in searching for model-based diagnosis techniques. Many techniques have been developed for fault detection and fault tolerant control. However, the issue of robustness of fault detection and fault tolerant control has not been sufficiently addressed. The conceptual of model-based diagnostics is shown in Figure 5.7 [Jie Chen and R.J. Patton]. Model-based diagnosis employs analytic redundancy compare actual components with an analytic model (mathematical model). It depends on the validity of the model, and the ability to accurately model a system, and it is relatively straight forward for linear systems, but difficult for nonlinear systems (most software-based systems).

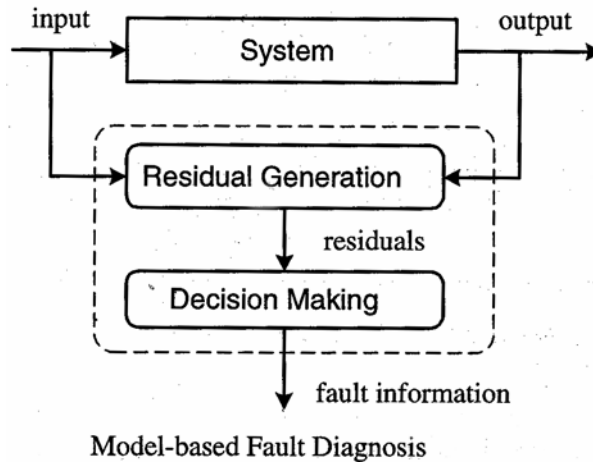


Figure 5.7: Conceptual Structure of Model-Based Fault Diagnostics.

Since disturbances, noise, and model uncertainties are unavoidable for any practical systems, it is essential in the design of any fault diagnosis/fault tolerant control system to take these effects into consideration so that fault diagnosis/tolerant control can be done reliably and robustly. Fault tolerance can be implemented to improve on one hand safety, and on the other hand reliability towards incorrect or incomplete knowledge due to system deficiencies, design compromise for efficiency, and faults in the decision procedure. Robustness can be implemented to improve reliability towards unexpected changes of situation due to the environment or system dynamics, and incorrect or incomplete knowledge due to lack of observability. Figure 5.8 [Jie Chen and R.J. Patton] illustrates the general schematics of Model-based fault detection technique.

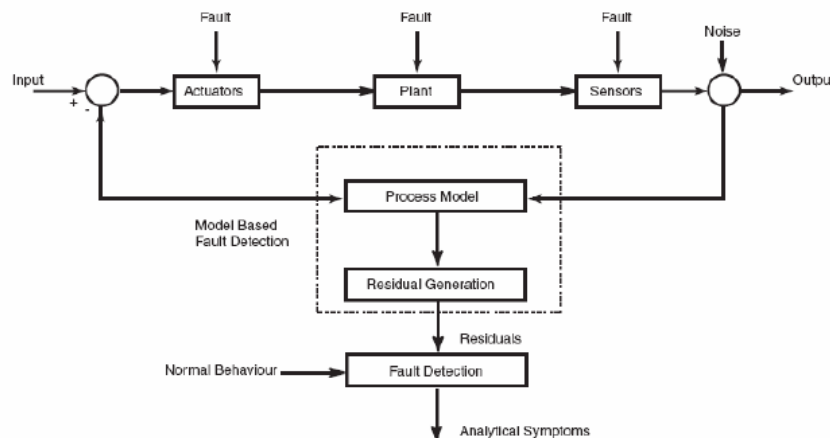


Figure 5.8: General Scheme of Model-Based Fault Detection.

Youmin Zhang’s lecture – FTC part one has suggested an active fault tolerant control which is depicted in Figure 5.9. As shown, the fault, disturbances, and noise can enter at any point. The key point is how to defeat faults. This was shown in Section 5.2.1.

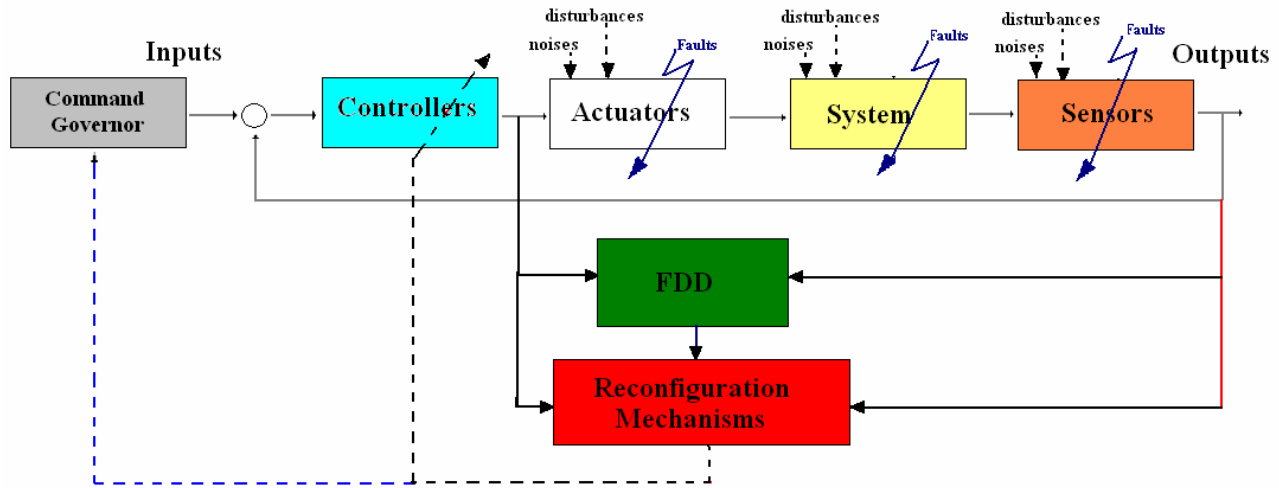


Figure 5.9: One Scheme of Active Fault Tolerant Control.

Rune M. Jensen and Manuela M. Veloso and Randal E. Bryant on their paper entitled, “Fault Tolerant Planning: Toward Probabilistic Uncertainty Models in Symbolic Non-Deterministic Planning” have a goal of a more probabilistic uncertainty model by distinguishing between likely primary effects and unlikely secondary effects of actions. We consider the practically important case where secondary effects are failures, and introduce n-fault tolerant plans that are robust for up to n faults occurring during plan execution. Fault tolerant plans are more restrictive than weak plans, but more relaxed than strong cyclic and strong plans. In their paper, they take a first step in this direction by introducing a new class of fault tolerant non-deterministic plans. Their work was motivated by two observations:

- 1) Non-determinism in real-world domains is often caused by infrequent errors that make otherwise deterministic actions fail; and
- 2) Normally, no actions are guaranteed to succeed. They have concluded that Fault tolerant planning is a first step toward more refined models of uncertainty in Symbolic Non-Deterministic Planning (SNDP). A fruitful direction for future work is to move further in this direction and consider fault tolerant plans that are adjusted to the likelihood of faults or to consider probabilistic solution classes with other transition semantics than faults.

5.4 FAULT AND FAILURE ACCOMMODATION

Future military land, air and sea based vehicles will require advanced system concepts that enable new vehicle capabilities under both nominal and adverse conditions. These systems will be required to maintain control under faults and failures, prevent and recover from vehicle loss of control conditions, and provide automatic collision avoidance capability. A fault is defined as a “malfunction” of any physical component or a sub-system that results in its failure to perform as designed. A system fault can arise from natural wear and tear of mechanical or electrical components, external unknown catastrophic disturbances, and improper maintenance of electro-mechanical components. It is highly desirable that when a fault occurs, it is detected as soon as possible necessary action can be taken. This timely response to faults reduces any disastrous consequences. For this reason, fault detection and isolation (FDI) methods are becoming very desirable for advanced vehicle systems. Faults being dynamic in nature,

the reconfiguration method should be capable of accommodating them quickly, especially for complex systems like advanced military aircrafts. Reconfiguration based on adaptive techniques demands a fast detection and isolation of a fault and is computationally involved. The systems of the future will have a reconfigurable flight control system to a transport aircraft model for the purpose of achieving an integrated failure accommodation and upset recovery system. The major task of such systems is the process of detection, identification and accommodation of faults and failures (FDIA). A number of approaches exist, of which model-based techniques show particular promise. Model-based approaches use analytical redundancy to generate residuals for the aircraft parameters that can be used to indicate the occurrence of a fault or failure. Actions such as switching between redundant components or modifying control laws can then be taken to accommodate the fault.

A review of “Robust Fault-Tolerant Control for Aircraft Systems” Thesis by Phalguna Kumar Rachinayani has addressed the need to design controllers that guarantee both stability and performance upon the occurrence of faults. He has presented different methodologies to design robust controllers that guarantee both stability and robustness for actuator faults and uncertainties. In the first part of this thesis, he has introduced the classical uncertainty formulation using Linear Fractional Transformation (LFT) and describes LFT’s special cases-norm bounded and convex polytopic uncertainty descriptions. Practical methods to formulate these uncertainty structures are described. In the same spirit, formulation of faults and their modelling for robust control system design is provided. In the second part of this thesis, he has demonstrated the application of a Luenberger observer for fast Fault Diagnosis and Isolation (FDI). He has described the methodology to design a robust optimal control for actuator faults and present controller reconfiguration mechanism based on switching for the design of Fault Tolerant Control (FTC). System with both norm bounded uncertainties and actuator faults is formulated and an analytic method to find a robust stabilizing and guaranteed cost reliable controllers are also mentioned.

5.4.1 Robustness and Redundancy

Traditional approaches to fault tolerance – reliability and redundancy. Why? Robustness – ability to continue to function within acceptable bounds in the presence of uncertainties.

Various techniques have been proposed in recent years for the detection, isolation, and accommodation of failures in dynamic systems. All of these techniques involve the generation of signals by sensors that are accentuated by the presence of particular failures, if these failures have actually been occurred. These sensor signals in turn depend on models relating the measured variables. Consequently, if any errors in these models have effects on the observables that are at all like the effects of any of the failure modes, then these model errors may also accentuate the signals. This leads to the issue of robust failure detection, that is, the design of a system that is maximally sensitive to the effects of failures and minimally sensitive to model errors. As a result, designing a failure detection system that is insensitive to model errors (rather than designing a system that attempts to compensate the detection algorithm must be concentrated by estimating on-line uncertainties.

Application robustness, defined as the ability to provide a contained degradation in performance when the algorithm solving the application is perturbed in its structural parameters, has an immediate impact on the design of a reliable system. The robustness of the failure detection process consequently depends to a great degree on the reliability of the redundancy relations, which in turn is affected by the inevitable presence of model uncertainties.

5.4.2 Fault-Tolerant Control and Reconfiguration

Fault tolerance is one of the principle mechanisms for achieving high reliability and high availability in propulsion and any other systems. System controls should be equipped with appropriate fault-tolerance schemes to ensure their reliable and safe operation in the presence of component failures. As an example,

Figure 5.10 describes the block diagram structure of a fault tolerant control system [Phalgun Kumar Rachinayani].

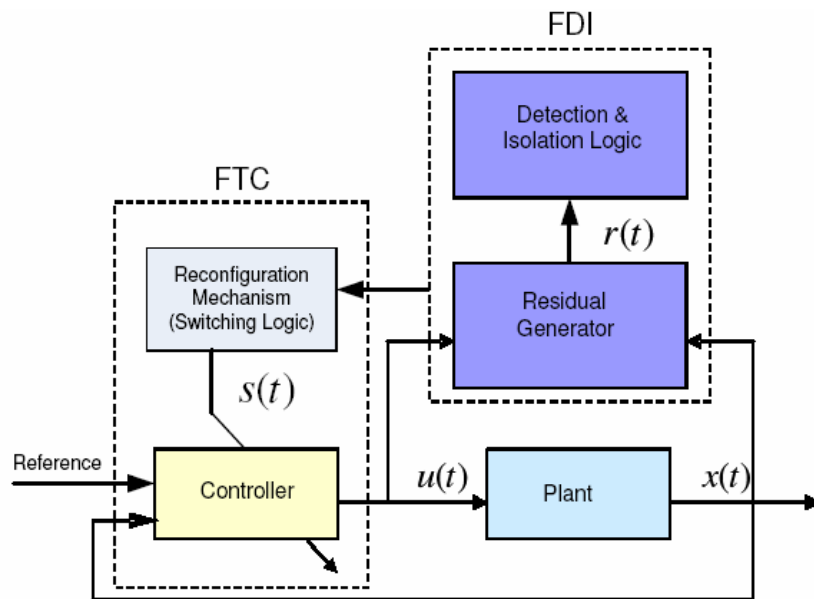


Figure 5.10: Block Structure of Reconfigurable Fault Tolerant Control System.

System reconfiguration, which enhances reliability by dynamically using spatial redundancy, is generally the most time-consuming fault-/error- handling stage. The reconfiguration latency, defined as the time taken for reconfiguring a system upon failure detection or mode change. Various fault reconfiguration strategies to locate and isolate the failures have been proposed. These schemes allow the controller to operate with a degraded performance even in the presence of faults. Fault-tolerant approach for discrete-event systems can be achieved by integrating sensor fusion within a diagnosis and control reconfiguration mechanism. This approach can automatically compute fault-tolerant control/reconfiguration actions which maintain pre-specified control objectives. Reconfiguration is based on model-based diagnostic representations and algorithms, and integrates diagnostics and control reconfiguration for discrete event systems using a single modelling mechanism and suite of control algorithms. Given failures in the system, a modelling approach can facilitate diagnostic isolation while performing sensor fusion to minimize false alarms, thereby increasing tolerance to sensor faults as well as (non-measurable) component faults. The algorithm works dynamically and individually: system reconfiguration starts instantly upon the emergence of a fault and the replacement of a new faulty component is considered independently from previous replacements. Hybrid Estimation and Control is becoming very popular. A broad ‘Hybrid’ philosophy is needed; such as ‘hybrid modelling’ which takes the benefits of both analytical and empirical models; ‘hybrid system dynamics’ which incorporate both the continuous system dynamics and discrete event dynamics which will be more realistic in real engine operations and finally ‘hybrid algorithms’ which take advantage of both analytical and intelligent (rule based) algorithms to tackle these complex issues. Such effective integration of analytical and intelligent tools towards the development of new and innovative identification/control strategies for complex dynamic systems is of paramount importance in achieving a reconfiguration scheme, including a reconfiguration algorithm.

5.5 PROGNOSTICS

Traditionally, maintenance on aircraft has been conducted on-time or on-failure. An example of the former is the overhaul or replacement of aircraft engines after a stated number of operating hours. In avionics and

flight-control systems, the latter is more common, and redundant components are provided in order to preclude loss of function following a single component failure for flight- or safety-critical functions. The desire to lower maintenance costs and associated system down time has led toward driving component failure rates in all aircraft subsystems sufficiently low to allow more maintenance to be conducted on failure or specified degradation of components rather than on time. However, unpredicted failures or the concern for such failures limits the degree to which maintenance and logistics footprint (and associated costs) can be minimized. Only if it were possible to predict exactly when a component (by serial number) would fail or when its performance would fall below that allowable would it be possible to further drive these costs down. By the mid 1990s, it appeared the technology (in the form of electronics, sensors, and computational technology in highly miniature form) existed or was close enough to realization to postulate tracking of performance on an item by item basis to predict exactly when it would fail or cease to perform acceptably, and do so with enough warning to have a replacement part standing by at the right place and time to minimize maintenance impact and footprint (and safely get the maximum life out of each component). Thus, instead of *diagnosing* and correcting problems only after they occurred, the aircraft could proactively predict impending problems in sufficient time to preclude their occurrence. This concept is called *Prognostics*.

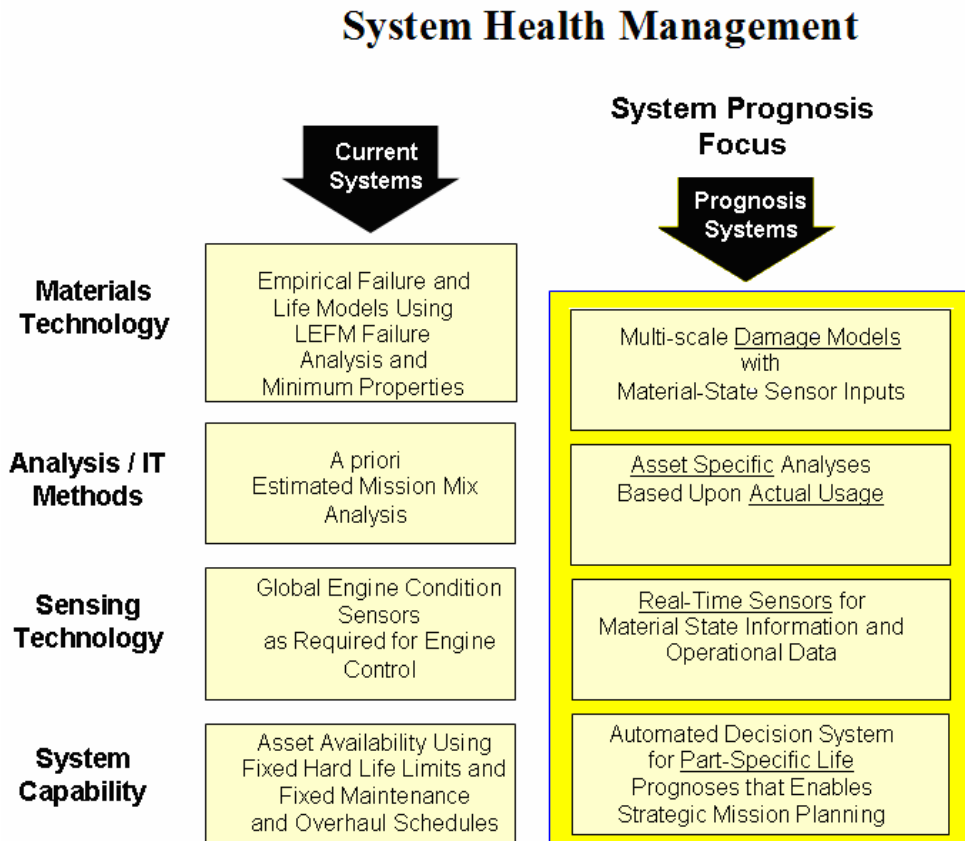


Figure 5.11: System Health Management.

5.5.1 Prognostic Assessment

There are a number of initiatives being pursued to enhance the capabilities of the present maintenance system. There are many programs that are in the process developing prognostic technologies. Many of these technologies are being matured so that they can be deployed on the systems as soon as possible to try to eliminate some of the shortcomings of the present maintenance process. Specifically, propulsion system

developers have taken the lead in the development and deployment of prognostic systems that will greatly enhance the present maintenance system and provide more availability of those systems. In order to accomplish this, the exploitation of what is called the “prognostic region” must be accomplished. This is a departure from past experience that used Built-in Test (BIT) for diagnostics. BIT utilized a percentage above a threshold value to determine when a failure had occurred. At this value the system was determined to be failed and needed to be repaired by replacement parts or other means to get the system functioning again. The paradigm shift has been to develop prognostic algorithms that will predict when a failure will occur and determine the life remaining to prevent a hard failure. This paradigm change in maintenance philosophy will eliminate the reactive maintenance event that might take place during a period of critical operations when the systems are needed to perform. The establishment of the BIT threshold was already established by the diagnostic engineer in the past when determining the BIT threshold; it was just a determination as to where that threshold should be and when to set a Fault Isolation Code (FIC). In determining the prognostic region, it is the same process except we want to move up the curve to determine how much operating time is left until we get a functional or hard failure. It is the slope of the curve that we are attempting to calculate so that an accurate Time to Failure (TTF) can be estimated. If an accurate assessment of the slope of the curve can be derived, then scheduling of the repair can be accomplished prior to a functional failure. While this will not improve reliability of the system, the reliability is inherent to the developed system; it will greatly enhance the availability of the system. The algorithms to predict the remaining life of the system prior to failure are under development.

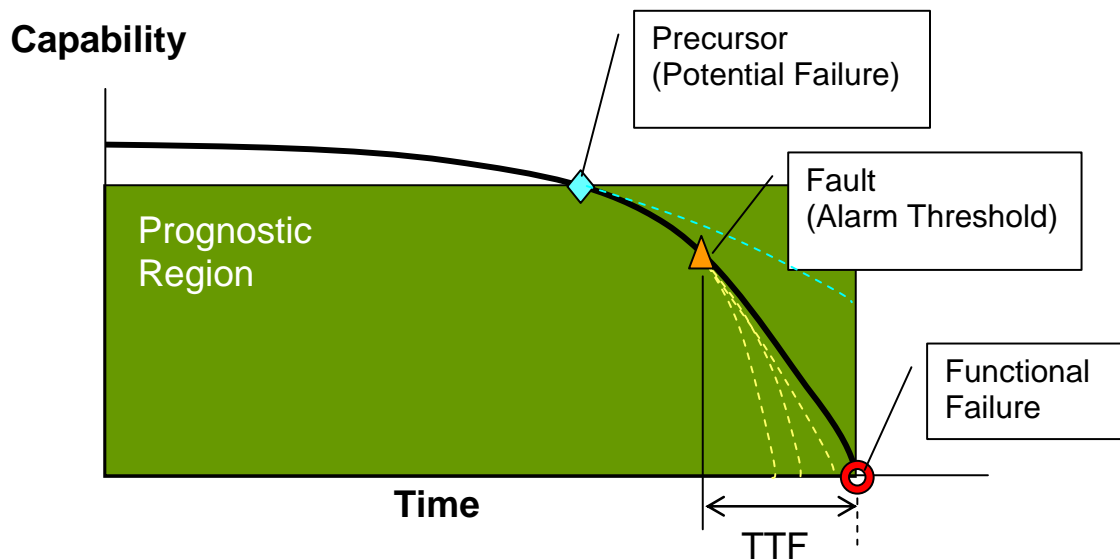


Figure 5.12: Prognostic Regions.

5.5.2 Development of Diagnostic Systems as a Precursor for Prognostics

There have been many developments in Prognostics that have evolved from advanced diagnostic techniques. Initially, a prognostic was thought of as an advanced diagnostic technique, but in reality, it is a paradigm shift of a new way of managing the life remaining and determining the health of the system at any point in time. Previous diagnostic developments were aimed at improving the diagnostic capability of systems and eliminating Cannot Duplicate (CND) and ReTest OK (RTOK) type problems. The development of the Flight Control Maintenance Diagnostic System (FCMDS) was to look at the diagnostic accuracy in terms of the RTOK rate. The RTOK occurs if a Line Replaceable Unit (LRU) is removed from an aircraft during aircraft troubleshooting, but no fault can be found during bench testing or subjecting the unit to an end-to-end testing on a test system. This places a great demand on the supply system and will require the stocking of many spares to keep up with the flight line demand for spare parts

and LRUs. One way around this is to put the LRU into another aircraft to see if it will function normally and work in that aircraft. This was tried at one time, but using a multi million dollar aircraft as a test bed becomes very expensive and very time consuming and this was not a successful maintenance concept. It was discovered that taking the LRU out of the system could potentially remove or change some of the diagnostic evidence and trying to diagnose systems with parts removed would not yield successful results. The FCMDS program was then structured around troubleshooting the system with all LRUs in place. A software model of the flight control system was developed using model based techniques so that anything that could happen in the on board flight hardware could be put in the model to diagnose the system. This was one of the first instantiations of model based reasoning. Additionally, it was realized that the wiring system itself cause many problems to occur, and many fault codes to set when a signal that did not get to its destination. While the setting of the fault code might have pointed the technician to a certain LRU, it was really a case of the signal not getting to the LRU for it to provide an appropriate response to the stimulation. Upon realization of this, the wiring diagnostic module developed as an integral part of the diagnostic system.

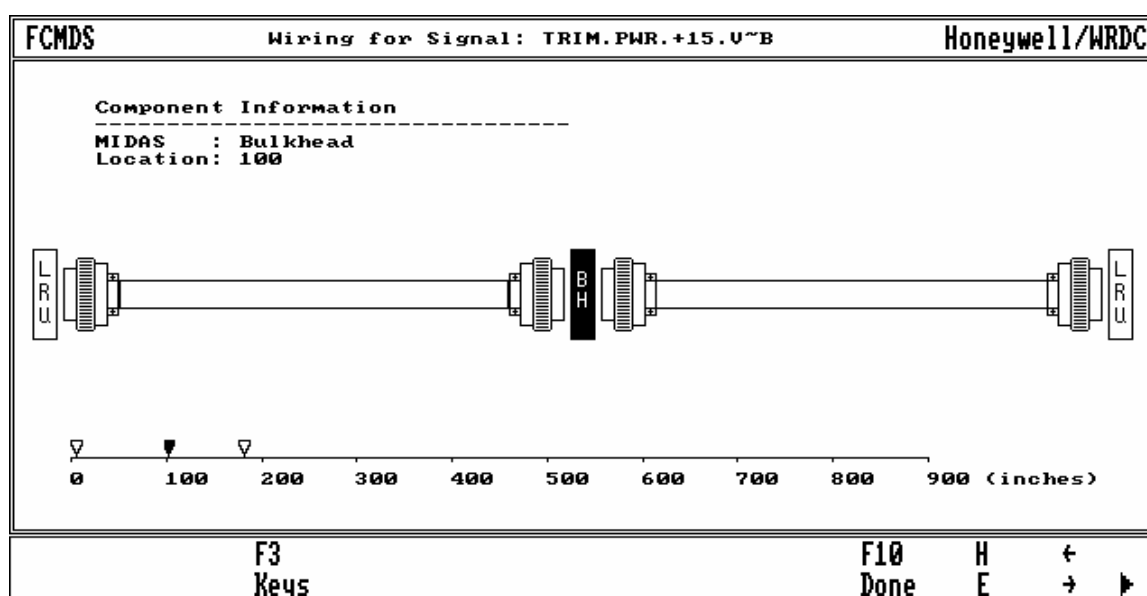


Figure 5.13: Wiring Diagnostic Module.

Wiring is an integral part of the control system because it connects the parts of the system. In a sense, the wiring is the neural network of the system. Wiring is often neglected in development of fault isolation manuals and techniques because wiring failures typically do not occur until the aircraft has been in service for at least five years. Unfortunately, when a wiring failure does occur, the information necessary to rectify the problem may be missing or in a form that is difficult to obtain. In addition, there are few technicians that are expert in diagnosing wiring problems, and when a wiring diagnostician is needed they are very hard to find. Diagnosing wiring is more of an art than a science, and it is something that comes with practice and diligence and a mental attitude of the ability to find the problem. The FCMDS made wiring an integral part of the diagnostic approach. Since information flow was contained in the diagnostic model, the wires were modelled as well. Therefore, each diagnostic observation that was mapped into the model includes all the wiring connections implicitly. The wiring diagnostic model developed for the FCMDS system was manually tested with something like a DVOM for resolution of the problem. Since that development, automated techniques are under development that are nearing maturity. Embedding an automated wiring diagnostic module within the system takes the human out of the loop and finds faults during system operations taking into account many factors that can not be tested when the aircraft is on the

ground. Attributes such as air speed, g loading, altitude, temperature, angle of attack, can all have an impact on the wiring system. When the aircraft returns from a flight, the evidence of a failure that occurred during g loading in a turn is not obtainable on the ground, therefore this problem is almost impossible to diagnose. With the development of embedded wiring diagnostic techniques collecting this data during a flight gives the technician a much better chance of finding the problem the first time it occurs.

When diagnostic information became accessible through a 1553 bus interface, the technician's job got harder. Suddenly to diagnose an aircraft failure, the technician must be able to read hexadecimal code in the digital flight control memory. Many pieces of data must be obtained and interpreted, a source for clerical errors. People make errors when performing tasks like these. FCMDs through an electronic interface drew information from the aircraft in digital form and performed the necessary interpretation of the hexadecimal codes. Computers are well suited to this type of clerical task both in terms of speed of acquisition and accuracy of results. More important, however, is that the data collection became a fundamental aspect of the FCMDs test language design. In a test, if the information is required that the system can obtain automatically, the diagnostic system will establish communications with the aircraft data bus and acquire the information. The technician is informed through the dialog display when and what data is accessed and including the status of the system.

The FCMDs development represented a precursor to the development of prognostics. In FCMDs, the human controlled the diagnostic process and did the entire test sequence manually, guided by the system after inputting test results from individual tests. Taking into account that the system was designed for diagnostics and used a BIT system, the addition of FCMDs to the diagnostic process proved very successful. The field test proved that the FCMDs system could significantly reduce false removals by 80%, and reduce the average diagnostic time by 25% and greatly enhance the level of performance of the maintenance technician using FCMDs over currently used methods at that point in time. This also provided the catalyst to support a two level maintenance concept that was under development at the time. The decrease in the RTOK rate would provide the additional spares to support a two level maintenance concept.

Following the development of FCMDs, a means of automating many of the manual processes that were in place at that time of the FCMDs implementation have been computerized. Rather than having the technician analyze all the data, the data is collected and sent to a central server where data mining techniques are used to analyze all the relevant data from the system. Analysis techniques are embedded into the central server to look at the trends taking place in the system over time. These trends will allow the system to make projections over time as to when the system will need maintenance prior to a predicted functional failure. The system will alert the maintenance operations centre as to the repair schedule for the effected system. This new concept is called the predictive maintenance concept. The prognostic attributes of the system are used to make predictions as to when the system will fail. It is the prognostic attributes that are being developed that will make the predictive maintenance concept possible. The next big step in this process is to integrate the current research in prognostic development is to integrate this with the logistics infrastructure.

The definition of data fusion has developed over ten years. Initially in 1987, the JDL Data Fusion Subgroup described data fusion as a process dealing with the association, correlation, and combination of data and information from single and multiple sources to achieve refined position and identity estimates, complete and timely assessments of situations and threats, and their significance. Additionally, the data fusion process was characterized by continuous refinements of its estimates and assessments, and the evaluation of the need for additional sources, or modification of the process itself, to achieve improved results. More recently, in 1998, Steinberg simplified the definition for data fusion as a process of combining data or information to estimate or predict entity states. Spanning this time frame, many DoD projects under the guise of "sensor integration" have accomplished the "fusing" of data from several

sensors and information sources. Through its evolution, sensor integration has been further developed and termed both “Sensor Fusion”, “Data Fusion”, and “Information Fusion” without a clear definition of the distinctions between them. Capabilities developed were predominantly focused on on-board weapon systems sensor capabilities ranging from target tracking, weapons control, and system navigation. Disciplines, methods, and techniques emerging from sensor fusion investigations run the gamut and include general statistical methods, hypothesis testing, graph theory, data representation, resource management, artificial intelligence (AI), fuzzy logic, and neural networks. Evolving from the earlier projects, military systems have solved more ambitious tasks and have applied more types of sensors and information sources.

The DoD challenge now is to move forward in data management of the sensor data collected from these on-board weapon systems and to fuse this operational data with data generated through test, inspection and repair of the system, as well as manually reported operational data. It is believed that integrating these diverse data sets will enable advanced capabilities for logistics support, to include maintenance practices and processes. A research area currently being developed within the Air Force Research Laboratory (AFRL) calling for this paradigm in data integration is Integrated Systems Health Management (ISHM). The AFRL ISHM program concept is a fully systems-of-systems vision. The AFRL projects which fall under its umbrella range from sensor development, life prediction methodologies, to asset management and advance to improved system engineering design processes. Key capabilities for ISHM is the DoD vision for Conditioned Based Maintenance Plus (CBM+) to improve upon knowledge-based tools and applications.

AFRL’s Health Management research links to the Air Force Transformation Flight Plan, which addressed development of advanced Condition-Based-Maintenance-Plus (CBM+) capabilities (AF/XPXC, 2003). CBM+ is the maintenance management policy being adopted from the Department of Defence (DoD) strategic vision titled “Future Logistics Enterprise” (FLE), now referred to as: Force-centric Logistics Enterprise). The DoD FLE vision documents were paramount in initiating the force transformation efforts being implemented by US military services today. The FLE is comprised of six focus areas, one of which is CBM+, [McRuer, D.D. Graham, E. Krendel, and W. Reisner, Jr.].

The CBM+ maintenance management concept is unique to the DoD. As documented in the DoD FLE, the interim policy states:

“Condition-based-maintenance (CBM) can be defined as a set of maintenance processes and capabilities derived, in large part, from real-time assessment of weapon system condition obtained from embedded sensors and/or external tests and measurements using portable equipment. The goal of CBM is to perform maintenance only upon evidence of need. CBM+ expands on the basic concepts of Condition-Based Maintenance by encompassing other technologies, processes, and procedures that enable improved maintenance and logistics processes”, [McRuer, D.T. and E.S. Krendel].

Each military service is in the process of adopting the DoD FLE guidance on CBM+, making further refinements upon the definition [Tustin, A]. The FLE goal is to implement CBM+ with enabling technologies that advance and implement the capabilities or programs, listed in Figure 5.14 below, on future, acquisition, and sustaining weapon systems where possible.

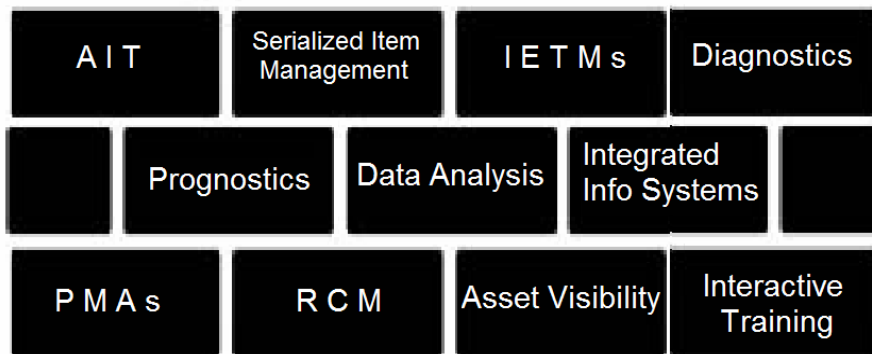


Figure 5.14: DoD CBM+ Initiative: Listed CBM+ Capabilities or Programs Being Targeted under the DoD's FLE.

5.5.3 US Air Force CBM+

The Air Force kicked-off its examination for developing its CBM+ policy by enlisting the Air Force Logistics Management Agency (AFLMA) to conduct a comprehensive study on the subject. The AFLMA study concentrated on “Air Force CBM+ implementation as a basis for establishing an Air Force CBM+ policy”, [McRuer, D.T., and E.S. Krendel]. This study was completed in September 2003.

Systems Health Management is not specifically called out in the AFLMA study. However, their definition for CBM+ combines On-Condition Maintenance attributes with the Reliability Centered Maintenance, Joint Total Asset Visibility, and System Health Management concepts. The AFLMA study proposed the Air Force adopt the DoD FLE CBM+ definition with the following additional phrase: *“These future and existing technologies, processes and procedures will be addressed during the capabilities planning, acquisition, sustaining and disposal of a weapon system”*. The AFLMA also proposed the following technologies should be listed under the AF CBM+ as enablers:

- Prognostics
- Diagnostics
- Data analysis
- Interactive Training
- Portable Maintenance Aids
- Integrated Information Systems
- Automatic Identification Technology
- Interactive Electronic Technical Manuals

New on-board and off-board technological capabilities are needed to support transforming maintenance to implement CBM+. The overarching AFRL vision for Health Management fully integrates CBM+ technologies, with research primarily focusing on prognostics and diagnostics capabilities that influence or drive capabilities of the other technologies listed.

AFRL’s Health Management research is in line with the Air Forces’ desire to develop transformational capabilities as called out in the Air Force Transformation Flight Plan, published November 2003. In the Flight Plan’s chapter titled “Developing Transformational Capabilities”, the first of sixteen goals stated is to develop “Seamless joint machine-to-machine integration of all manned, unmanned, and space systems”. In this statement, “machine-to-machine” can be interpreted to include air-to-air, ground-to-ground, air-to-

ground and vice versa, the word “all” would imply systems in both sustainment and new acquisitions, encompassing the spectrum of logistics and operational functional applications.

Further discussion in the transformational capabilities chapter, under Section F. titled “Agile Combat Support”, is focused on transforming logistics. This section calls out several Information Technology (IT) programs, as well as Condition Based Maintenance-plus, as key programs or future systems enabling transformational capabilities to meet the lighter, leaner, and faster combat support goals being addressed here. AFRL is working to bring these capabilities to the Air Force, which requires collaboration with a diverse systems infrastructure.

5.5.4 Integration with the Logistics Infrastructure

In order to take advantage of the advances in developed prognostic systems, these prognostic algorithms must be proven and tested in an operational environment. Additionally, the prognostic algorithms must be integrated with the present and future logistics environment. At the present time there have been many prognostic developments that have been advertised and some that have been tested, but not many in an operational environment and none with the integrated with the logistics infrastructure. While there is still a great deal of work to do in development of prognostics to determine the life remaining of systems and components, it is time to start integrating this technology with the logistics infrastructure. The Air Force is moving from many stand alone IT systems that provide single purpose assessments for systems in the aircraft to a centralized IT system (i.e. propulsion oil debris data) that will provide the needed services for the Air Force. This system, formerly known as Air Force Knowledge Systems (AFKS) is called Global Command Support System (GCSS). As many diagnosticians have known for many years that having the data to analyze in a central location makes the job much easier. Propulsion systems, for example, have many diverse data systems to support operations and maintenance, and most are stand alone and some are not available to be stored in a permanent location. This data is lost and will never be recovered. A movement is underway to integrate all the data into a central location so that is possible to do “predictive maintenance” on propulsion systems. The idea behind predictive maintenance is to use the prognostic TTF to estimate the time remaining. Using the multiple data sets from the propulsion system and development and use of data mining techniques will allow a greater resolution into the prediction of the life remaining until a functional failure occurs. The resolution of that aspect can be used by the present logistic infrastructure to start to schedule propulsion repairs as “future maintenance events” that occur prior to a functional failure. With GCSS, the supply data, operational schedule, personnel and proper tools can be scheduled for a future maintenance event. There is a great deal of research that will be needed to accomplish this futuristic maintenance system for the Air Force.

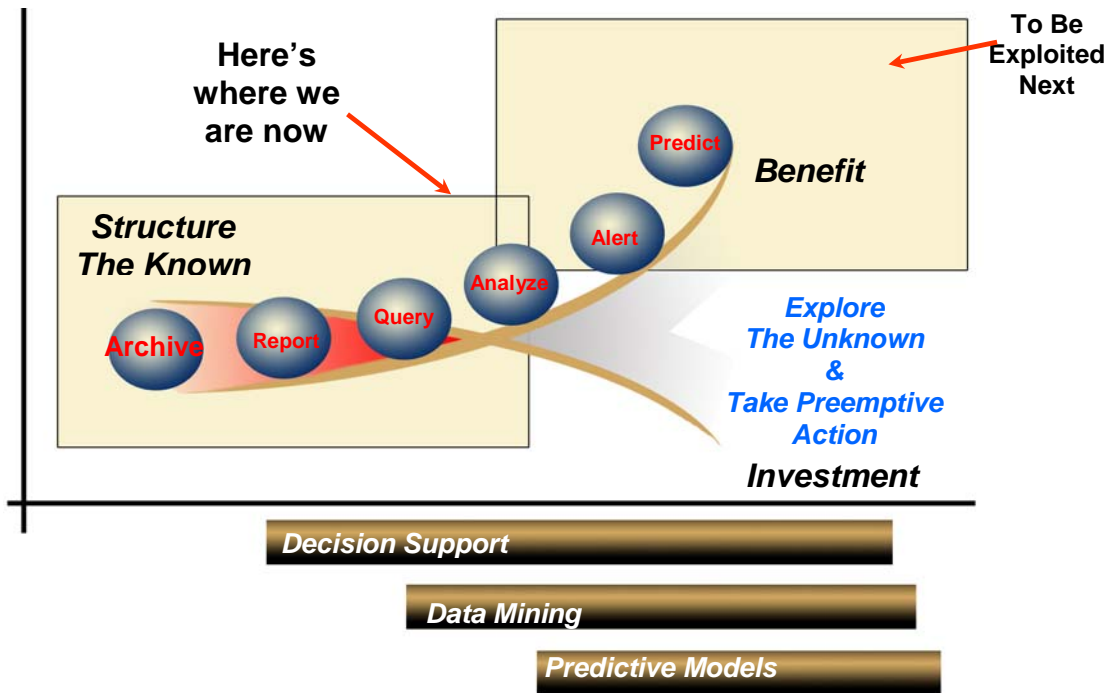


Figure 5.15: GCSS AF Data Services Capability Concept.

5.6 INFORMATION FUSION

Information fusion methods in principle have been applied ever since data has to be handled. There is no task finished with receiving data, they always have to be related to other data or previous knowledge to make sense and enable decisions. A very good example for basic, yet difficult information fusion is stereoscopic vision: one image alone just offers two-dimensional information, but fused with another, very similar image information enhances to 3D. Because information fusion is such a basic technique it can be found in many disciplines, from biology to the design of computational databases. Although there are a lot of solutions for various applications, no theoretic or conceptual work on information fusion has been done a long time. Even the term “information fusion” has not been existed up to 1991, when the JDL (Joint Directors of Laboratories, DoD, USA) introduced a definition meant for military applications. Due to increasing complexity and new applications groups of all disciplines became aware of the need for systematic research on that area. Now there are journals, web-sites and conferences dealing with information fusion as an independent yet interdisciplinary topic.

There is no general information fusion method which works for all addressed problems. In fact basic information fusion techniques of different disciplines have nothing in common except to provide a better use of data, which may be unstructured, inaccurate, overwhelming by volume or difficult to interpret. Often information fusion is referred as “filtering”, “pre-processing”, “decision making”, and so on. A short overview on the different approaches shows the variety and bandwidth of information fusion.

Information fusion subdivides in two major categories: Condensation and interpretation of data. Under “condensation” all methods are summarized, that shrink huge amounts of redundant data to valid information, that combine complementary data to new insights or that find hidden information in disordered data. A lot of well established methods like state estimators (Kalman filter, Luenberger observer, maximum likelihood methods and others), averaging concepts and voting schemes have been designed for that purpose. More recent methods are (active) perception management or entropy methods, which both are used to get the most useful information out of sensors or other data. Interpretation of data

is less algorithmic and more “smart” than the condensation task. Here data is associated, sorted, classified and decisions are derived. This is accomplished by expert systems, artificial intelligence and the often used probabilistic methods like Bayesian Networks or Dempster-Shafer theory. This discipline is object of ambitious development. In Figure 5.16 the main methods are outlined.

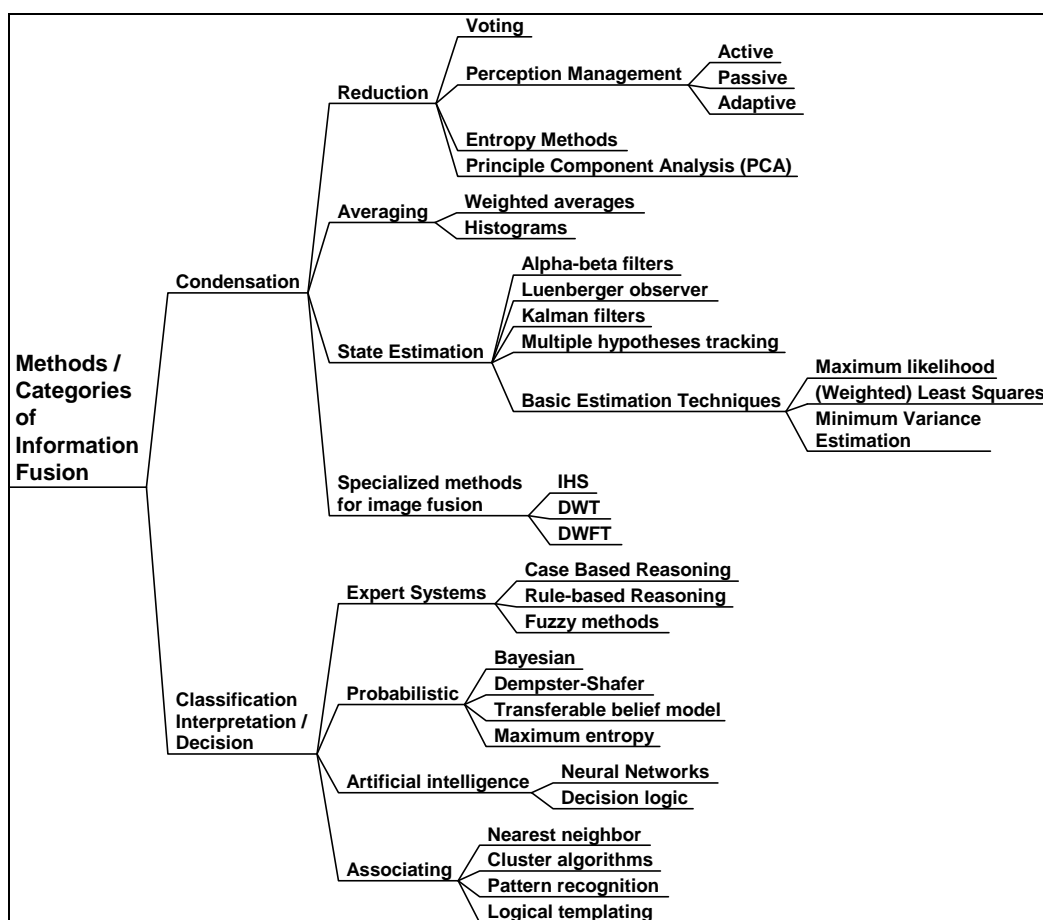


Figure 5.16: Methods of Information Fusion.

Technically it can also be decided between different types data fusion:

- Fusion of temporally data of **one** sensor.
- Fusion of data of multiple sensors of the **same type**.
- Fusion of multiple sensors of **different type**.
- Fusion of sensors and models.
- Fusion of sensors and a-priory knowledge.
- Fusion at raw-data, feature and decision level.

5.6.1 Definition of the Fusion Process

The following is the information fusion process:

- The combining or merging of data from different sources in a manner that provides extra information and/or better quality than any of the single sources involved (see Figure 5.17).

- The process of acquisition, filtering, correlation and integration of relevant information from various sources (see Figure 5.18).

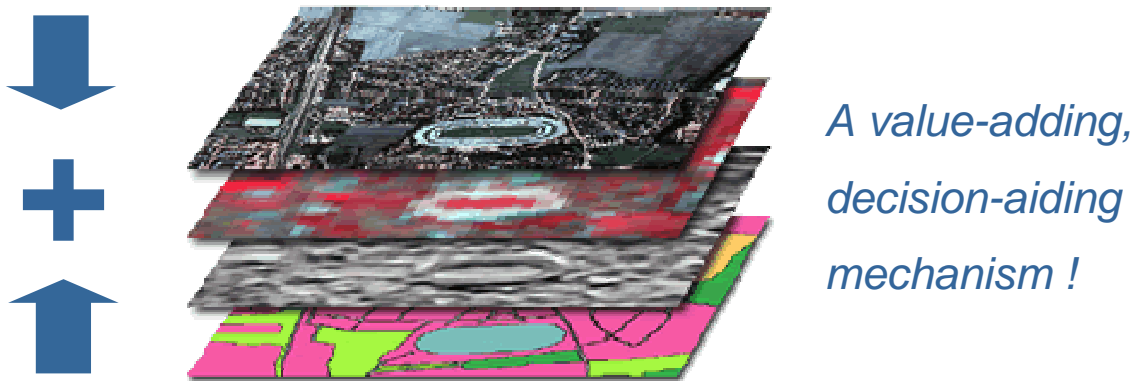


Figure 5.17: Information Fusion Levels.

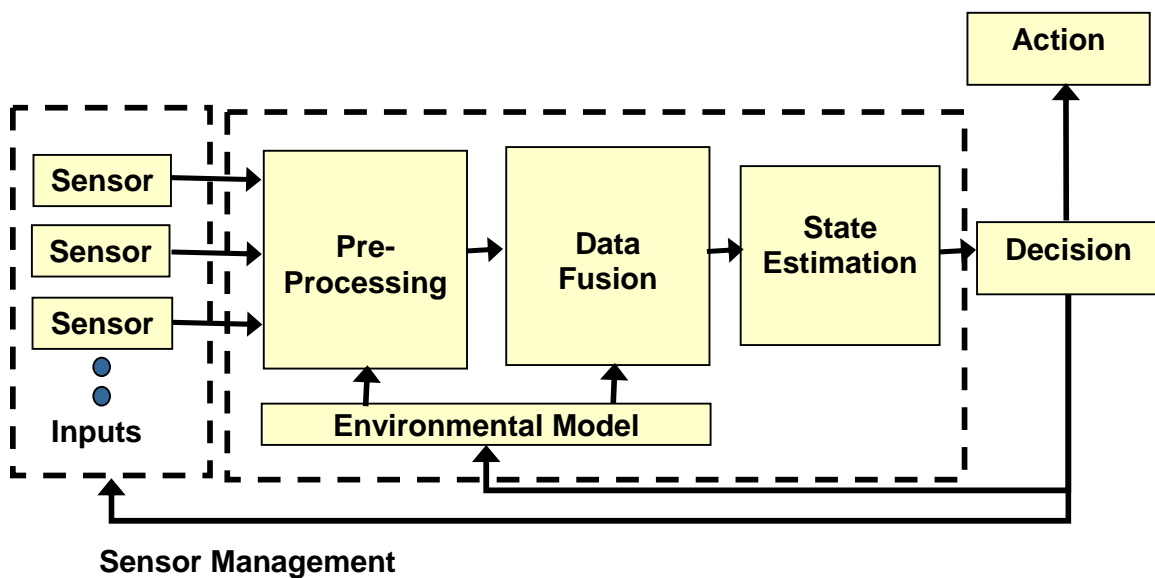


Figure 5.18: Acquisition Data Process.

5.6.2 Data Types

Data sets typically have various, recognisable patterns of distribution of the individual values. The following describes the data types encountered in various types of system operation:

- Not all of the data will be in a nice, user-friendly, easy-to-use form.
- Military information is often defined in terms of:
 - IMINT (Imagery – FLIR, Visible, SAR, etc.);
 - ELINT (Electronic Emissions – RADAR, MAWS);
 - HUMINT (Human Generated Intelligence);

- SIGINT (Signals); and
- COMINT (Communications).
- Combining different types:
 - Imagery (e.g. IR and Visible versions of the same scene) can be difficult if the two sensors are not co-located; and
 - How does one combine an image with text (HUMINT) that may, or may not, describe the same scene?

5.6.3 Categories of Fusion Concepts

Data fusion is defined as a formal framework in which are expressed the means and tools for the alliance of data originating from different sources. Data fusion techniques combine data from multiple sensors, and related information from associated databases, to achieve improved accuracies and more specific inferences than could be achieved by the use of a single sensor alone. It aims at obtaining information of greater quality; the exact definition of 'greater quality' will depend upon the application. In exploring the concept, data fusion has the following elements:

- Condensation:
 - Shrink redundant data to valid information;
 - Combine complementary data to new insights; and
 - Find hidden information in disordered data.
- Classification / Interpretation. Data is:
 - Associated;
 - Sorted;
 - Classified; and
 - Decisions are derived.
- Complementary Fusion:
 - E.g. several visual sensors pointing in different directions.
- Competitive Fusion:
 - E.g. measuring of a distance: laser ranger sensor, acoustic (ultrasonic) sensor pointed on the same point.
- Cooperative Fusion:
 - E.g. fusion of physical measurements
2D images → 3D representation.

5.6.4 Data Acquisition Problems

Within the current environment, the typical approach taken for AFRL health management programs is to acquire sample data from the multiple information systems used in supporting maintenance on the chosen legacy system, to accomplish development of the programs proof-of-concept. One would think that the targeted legacy weapon system would also benefit from this research as well, but this is not often the case

because the IT systems supporting sustaining operations are many and diverse. So, in the early stage of health management research, data acquisition is one of the primary issues AFRL managers contend with.

Each support function of a legacy weapon system typically utilizes multiple Air Force standard, and weapon system unique stove-pipe or stand-alone IT applications for support equipment. Typically these IT systems only share data if required by stake holders within the functional domain of the support activity, as depicted in Figure 5.19. Examples of the functional domain data for propulsion are: oil analysis results, digital diagnostic tool results, or on-board operational performance data.

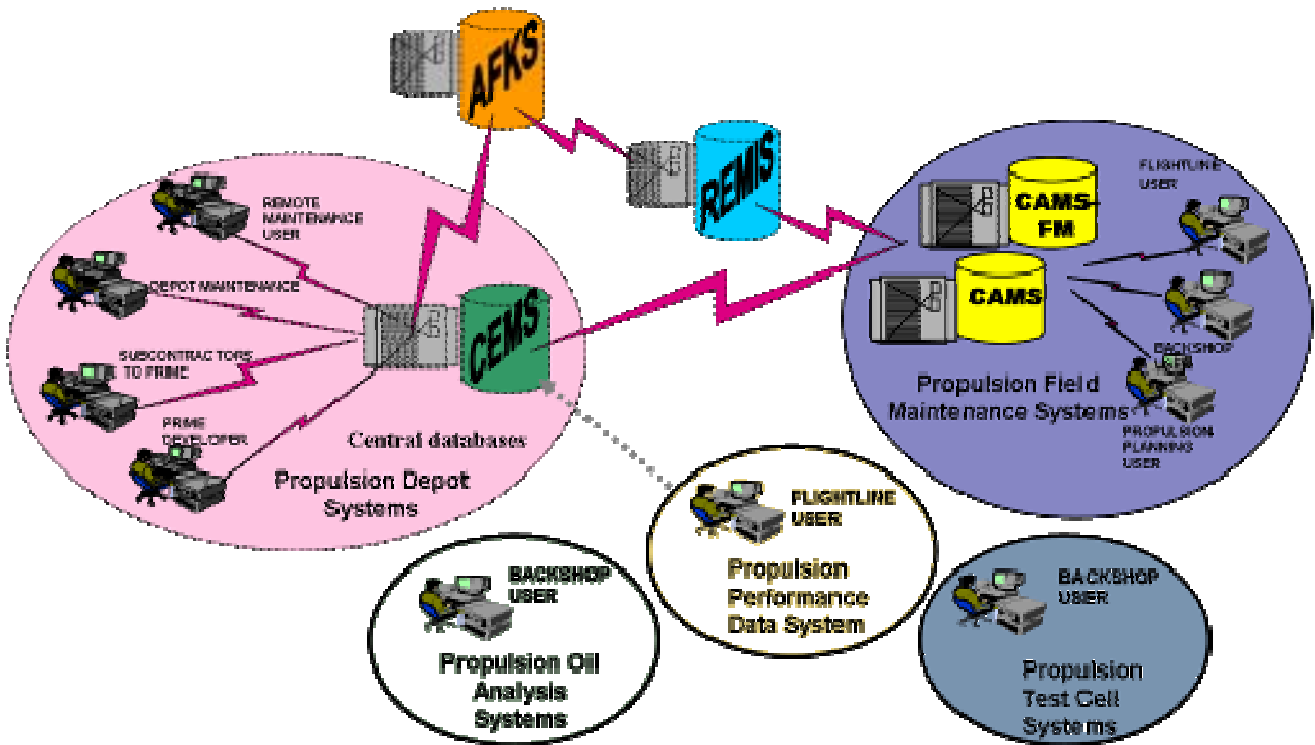


Figure 5.19: Legacy Propulsion IT System Interfaces: Initial GCSS Implementation of Legacy Central Data Systems Data. This figure also depicts legacy central data systems interfaces with base level systems and peripheral systems with no interface requirements that support propulsion assets.

It seems baffling that a research community, belonging to an organization as large as the Air Force, doesn't have readily available the operational data needed to conduct research. However, the AFRL engineer will typically spend 80% of the project time acquiring data and the remaining 20% on analysis. Therefore, the data situation requires the AFRL project engineer to spend countless hours coordinating with IT owners, or system users of the data sources, identified to support the specific program initiative.

The data acquisition problem is further compounded when a research project calls for data from multiple information sources, which is typically the norm for application concepts that hinge on data integration or data fusion. Also, there are times when multiple AFRL programs require the same data set from one of the available information systems, which under current practices, cause duplicate efforts from researchers for acquiring data to occur. The EHM team's goal is to reverse the two percentages previously stated, so that 80% of the time is dedicated to analyzing the data and 20% to acquiring the data. To overcome the data acquisition obstacle, the AFRL EHM team is seeking to forge a new paradigm in IT program management through collaboration, beginning with the GCSS Program.

5.6.5 Data Management Policy

Air Force data management policy has historically concentrated on the manual processes required to report equipment availability and maintenance activity. Any guidance on the disposition of data collected digitally, via test equipment or from equipment operation, has been left to the managing system/equipment program office or a Major Command to provide. Therefore, the collection of digitally generated data is sporadic and it's not normally stored or transferred for purposes beyond the test or operation being conducted. For this reason, the AFRL EHM team has initiated discussion among AFRL ISHM team members and Air Force policy makers on the need to draft an overarching data management policy to be supported by standard IT systems.

New policy is required to expand the central collection of system performance data and other data generated from sophisticated electronic test equipment. Such policy would provide the possibility of integrating on-board weapon systems collected data with other functionally related data. Thus enabling the creation of advanced CBM+ capabilities, or "the integrated application of a collection of advanced engineering, maintenance and information technologies to improve maintenance and logistics practices" as proposed by the AFLMA.

5.6.6 AFRL GCSS Research Environment

The goal is to establish an AFRL research environment where both programs (AFRL and GCSS) managers and their IT developers will collaborate on delivering program initiatives.

Over the life of a project, current thinking is that phase one would consist of concept and methodology development, which should include data acquisition requirements, data relationships and draft interface specification documentation. During phase two, concept maturation and demonstration, the team would concentrate on data acquisition and the database architecture required for developing the research capability.

Finally phase three, technology transition, would complete maturation of the Engine Health Management (EHM) decision support concept by broadening the capability to address all pertinent components within the targeted propulsion system. The theory behind this program progression concept is that the capability would be developed in a relevant environment or in an IT environment capable of supporting the concept. For the team this reality transforms for AFRL, GCSS, and their collective customer by meeting the program goals for a predictive maintenance system.

Changes to AFRL EHM program management have already been implemented using this program development strategy concept. The EHM team has begun targeting additional applicable IT acquisition programs open to this collaboration concept. The team is currently managing a number of Small Business Research projects to start the implementation of this strategy.

This conceptual process and strategic alliance is being developed by propulsion engineers and logisticians to bridge IT capability development gaps, which have existed for many years. Upon implementation of this concept the ability to use GCSS to store and acquire data will be realized. The team believes a strategic alliance with GCSS will ensure the probability of success in developing workable prognostic and diagnostic capabilities.

The EHM team also foresees a need to demonstrate new capability concepts that could be applied to sustainment logistics IT systems. They believe similar arrangements for AFRL program development could be implemented to bridge CBM+ capability gaps as legacy IT systems transform. However, much work is needed in changing paradigms on both fronts of the systems engineering process between the S&T and legacy IT development organizations.

There have been many developments in the past that would have gone forward faster if there had been partnering relationships and/or a GCSS system in which to conduct research. Finally, the EHM team likens the use of a GCSS research environment to the use of their Demonstrator Engine program, which is used to satisfy testing requirements of S&T propulsion component programs and is critical to program success. It is envisioned that an AFRL Research Environment within GCSS will become a reality in the very near future.

Finally, once the strategic alliance or program partnering concept is approved and implemented, this arrangement will benefit a broad range of other AFRL Science and Technology (S&T) initiatives, related to data fusion, modeling concepts, or the development of knowledge-based e-tool capabilities. Although the alliance team is currently focusing on the propulsion labs Engine Health Management S&T need and use of GCSS resources, the GCSS Architect expects the final program partnering document will serve as a “GCSS User Self-Development Guide” model, for even a broader range of Air Force customers to use in developing their desired knowledge-based capabilities.

Vision: GCSS AF DATA SERVICES Research Environment /AFRL Laboratory Resource

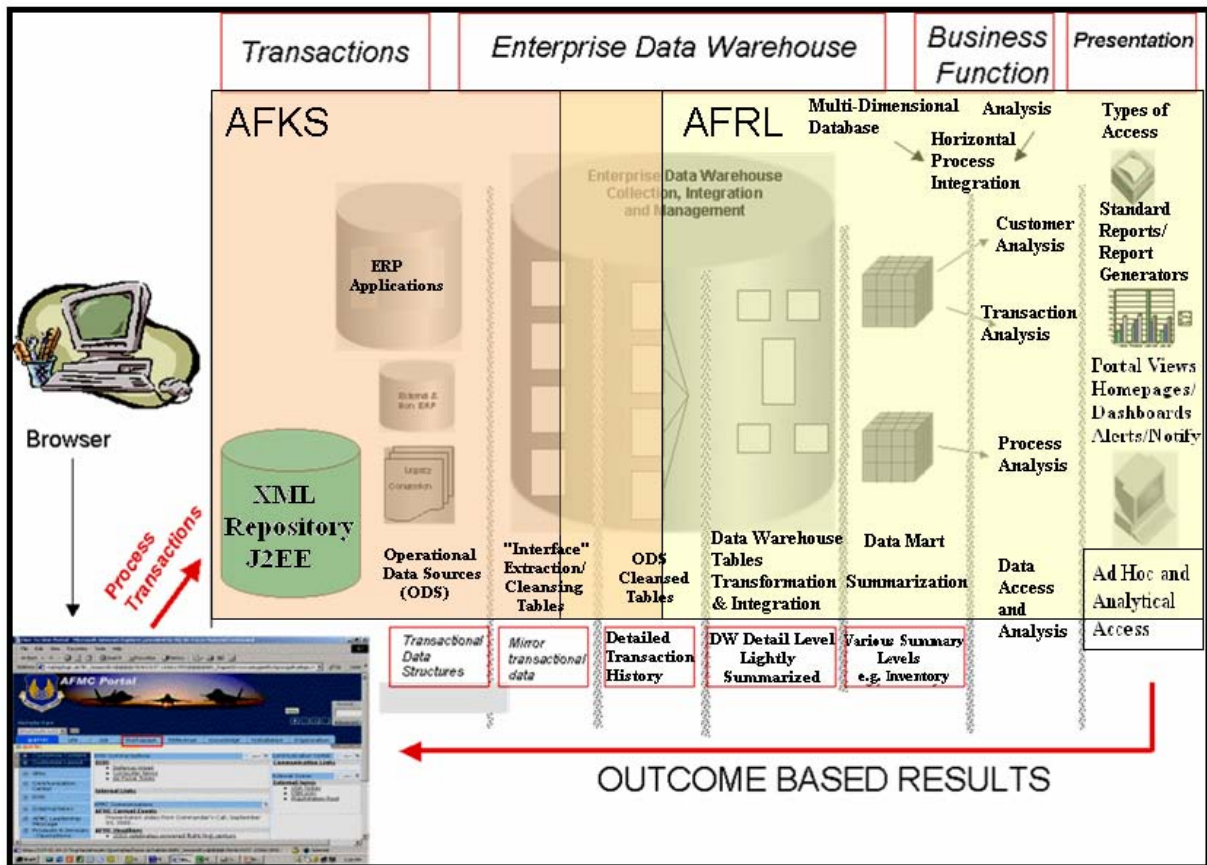


Figure 5.20: Integrated Vehicle Health Management (IVHM) and Condition Based Maintenance (CBM).

5.7 APPLICATIONS IN HUMAN-MACHINE INTERFACE

Application and usability of human-machine interface (HMI) aspects of tools/systems designed to enhance human functioning is vital to the augmentation of human-machine partnerships in the military

environment. In general terms, HMI systems are designed and optimized to visualize, control and report on data that is either being polled from or sent from other systems in near-real-time (see Figure 5.21). HMI are pilot-centered systems that provide prioritized data at the right time and in the right format to optimize situational awareness and increase mission effectiveness. Some aspects of HMI are listed below:

- Multimodal Interfaces:
 - Refers to interfaces that support non-GUI interfaces
 - e.g. complementary speech and pen input
 - Often distributed on a network

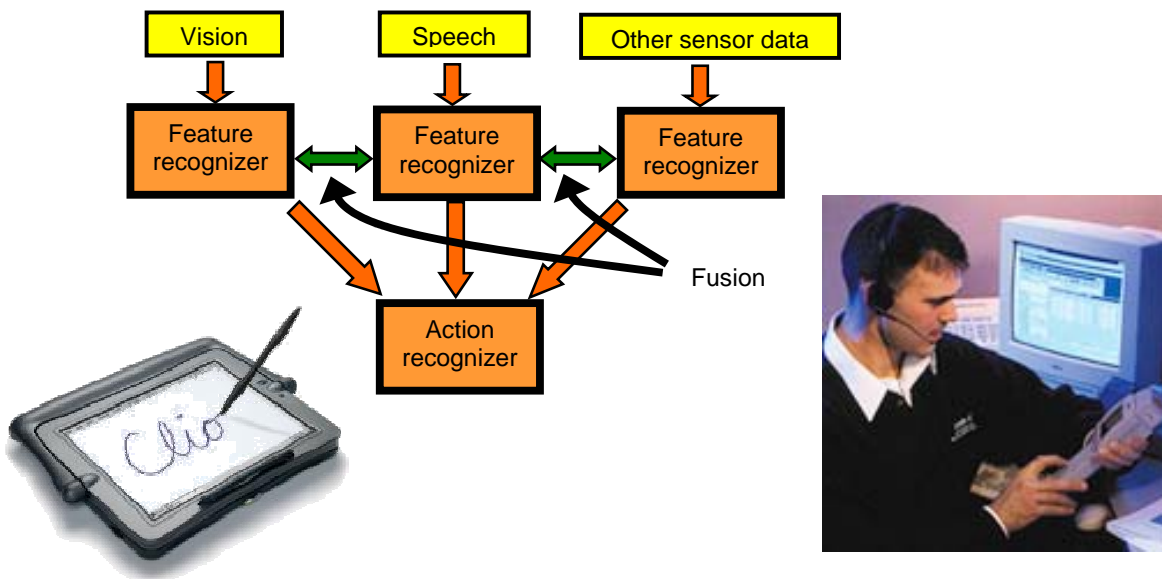


Figure 5.21: Human-Machine Interface.

5.8 SYSTEM HEALTH MANAGEMENT

Increased demands on autonomy in high performance aircrafts, UAVs and spacecrafts has led to the development of the concepts of Integrated Vehicle Health Management (IVHM) [Tustin, A.] and Condition Based Maintenance (CBM) [Tustin, A.]. Health monitoring or health assessment is an integral part of IVHM and CBM systems.

Health monitoring, as the name implies, involves monitoring the state or condition of components in a system. Most of the algorithms for health monitoring of aerospace components have evolved from fault tolerant control design applications. In these applications, in addition to monitoring the health of components, appropriate reconfiguration actions have to be initiated based on the diagnostic/prognostic information.

Efficient HM could be achieved only by combining data/information from multiple sources. Thus, data fusion is an important component of health monitoring systems where correlated data from multiple sources are used to determine the state of a system. Data fusion helps in achieving robustness, extended spatial and temporal coverage, increased confidence, reduced ambiguity, lower uncertainty and improved resolution [McRuer, D. T., D. Graham and E. Krendek]. Data fusion also helps in increasing diagnostic visibility, reliability and in reducing the number of false alarms due to the effective utilization of all the available data/information. Data gathered from different types of sensors need to be combined with

linguistic, knowledge-based information for achieving efficient monitoring. In systems where the health of several components is to be monitored, choice of appropriate data fusion architecture plays an important role. Also for applications like engine HM, a single diagnostic system may not be adequate to cover all types of faults/failures that could occur. In such cases, methods for combining information from several diagnostic systems to make appropriate decisions are also to be given due consideration.

The common goal of IVHM [Tustin, A.] and CBM [Tustin, A.] systems is to achieve automated health management. Hence, several common features would be evident in the description of both the systems given below.

The purpose of IVHM is to achieve the ability to perform vehicle maintenance based on component/subsystem condition and operational requirements, to automate flight certification, monitor and manage/schedule maintenance resources. For unknown fault/anomaly detection and decision support, IVHM utilizes model based diagnostics, intelligent data collection, signal processing, context models and correlation techniques.

In general, IVHM involves:

- i) Diagnostics – to determine the state or capability of a component to perform its function(s);
- ii) Prognostics – predictive diagnostics to determine the remaining life or time span of proper operation of a component;
- iii) Health Monitoring – to monitor the state or condition of a component; and
- iv) Health Management to make appropriate decisions about maintenance actions based on diagnostics/prognostics information, available resources and operational demand.

Figure 5.22 [Tustin, A.] shows the components of an IVHM system for an aircraft. Using the data gathered from the GPS, INS and fuel systems, aircraft health assessment is made using OSACBM (Open system architecture for Condition based maintenance) components [Tustin, A.]. Fusion of system identification and component health information helps in advanced diagnostics and prognostics for adaptive flight control reconfiguration as also shown in the example in Figure 5.22.

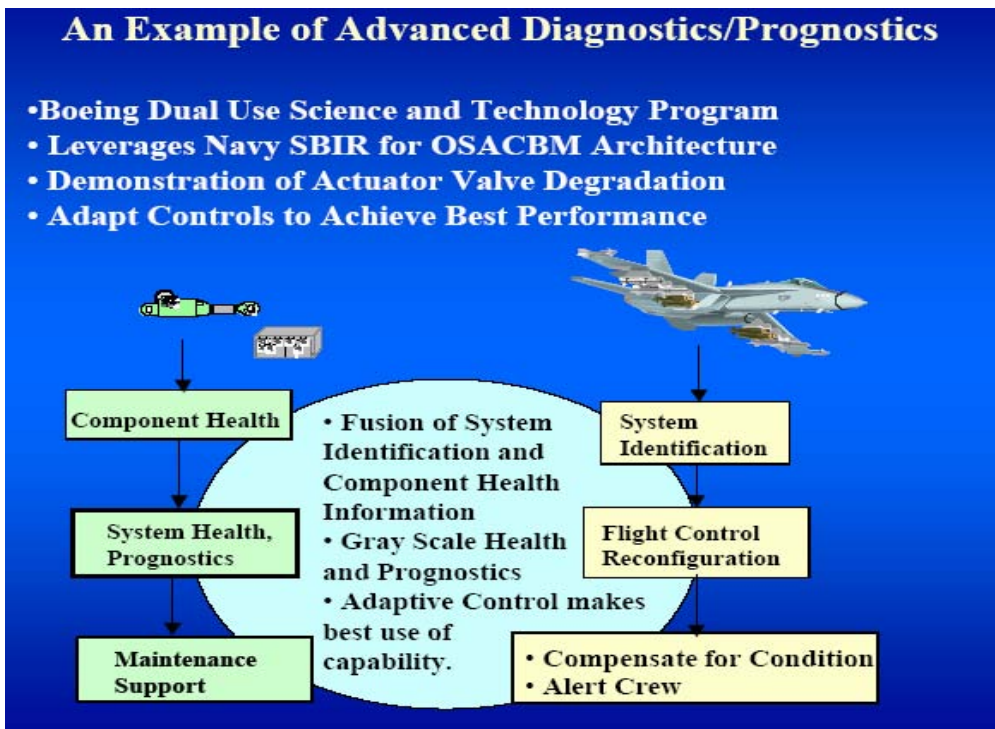
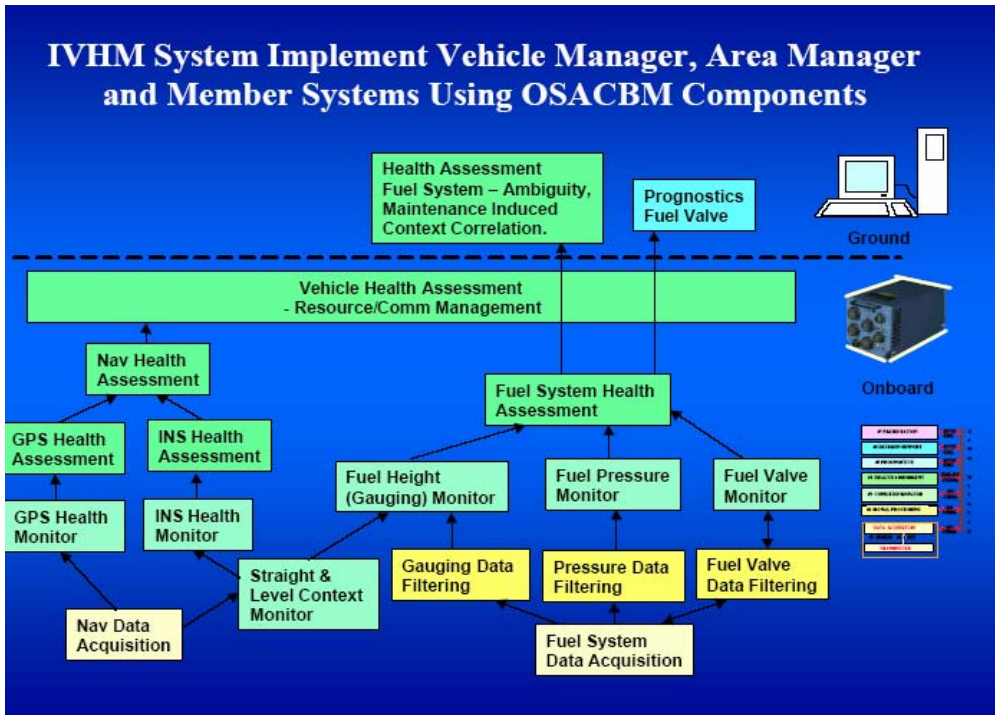


Figure 5.22: [Tustin, A.]: IVHM System and Advanced Diagnostics/Prognostics.

Condition based maintenance (CBM) and prognostics is necessary in military as well as industrial applications [Tustin, A.]. Since the CBM involves the integration of a variety of hardware and software components, a standard to encompass the entire range of functions from data collection to recommendation of specific maintenance actions is being evolved by Boeing, Rockwell and Penn State Applied Research Laboratory, ARL [Tustin, A.]. Figure 5.23 [McRuer, D. D. Graham, and E. Krendel]

shows the system overview for CBM system based on the OSA-CBM concept for health monitoring of an electro hydraulic spoiler actuator. The various elements in the figure are described briefly below.

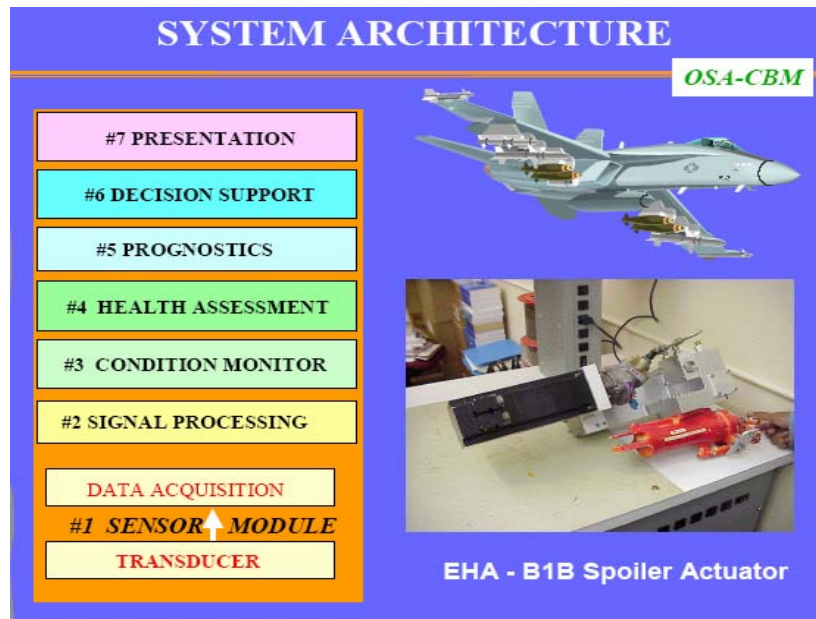


Figure 5.23: Health Monitoring of an Aircraft Actuator.

- #1,2 Signal processing module generates digitally filtered sensor data, frequency spectra, virtual sensor signals and other features using the transducer data from the data acquisition module.
- #3 Condition monitor generates alerts based on preset operational limits using the data from the sensor module, the data manipulation module and other condition monitors.
- #4 Health assessment module determines the health of the monitored component/sub-system or system, generates diagnostic records and proposes fault possibilities. The diagnosis is based upon trends in the health history, operational status, loading and maintenance history.
- #5 Prognostic module calculates the future health of the system based on the output of the health monitoring module.
- #6,7 Decision support module generates recommended actions and alternatives.

The motor system setup and the typical faults considered for the health monitoring of the electro hydraulic actuator described above are shown in Figure 5.24 [McRuer, D., D. Graham, and E. Krendel]. The implementation uses FFT (Fast Fourier Transform) and neural nets for signal processing, limit analysis for condition monitoring and causal net for health assessment.

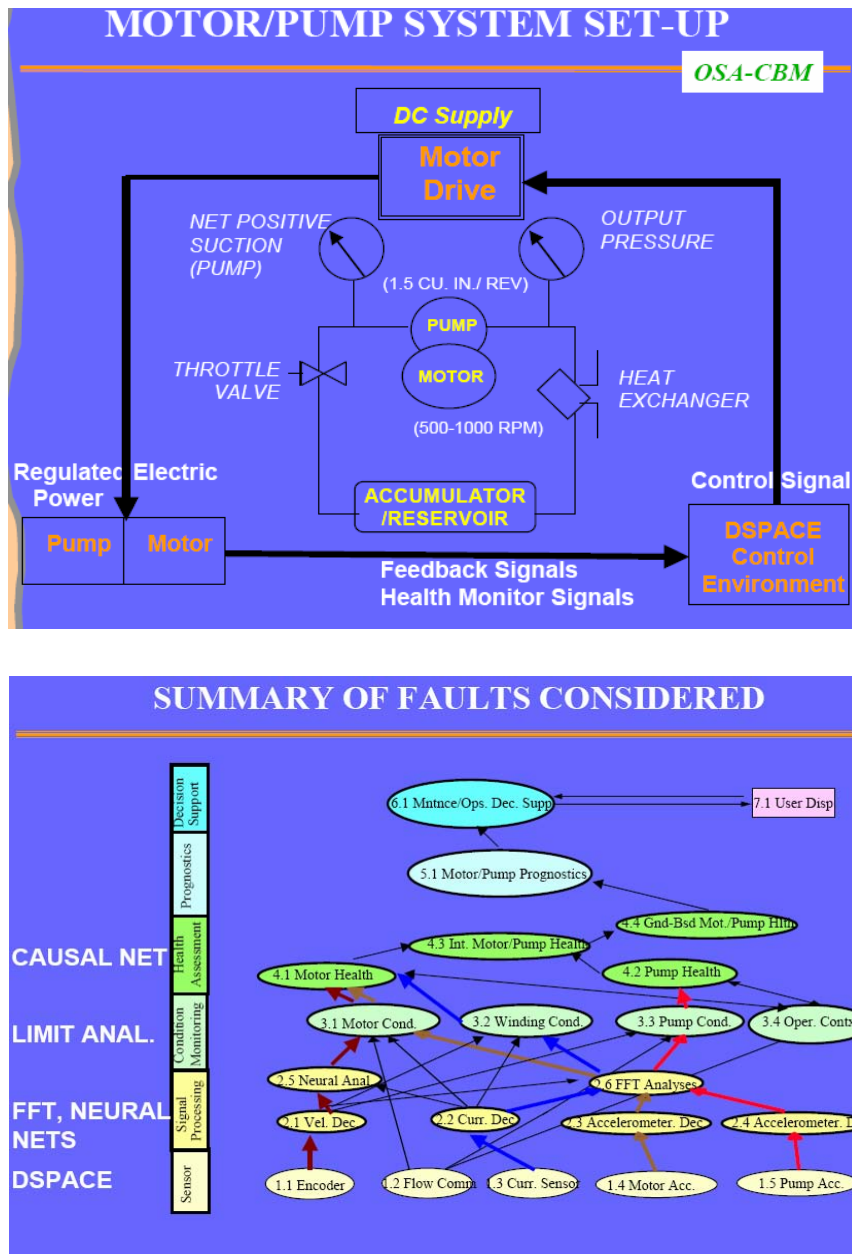


Figure 5.24: Health Monitoring System Example – Motor Pump Setup.

Several standards [McRuer, D. T. and E.S. Krendel] have been evolved for the development of systems based on OSA-CBM concepts.

5.9 INTEGRATED SYSTEMS HEALTH MANAGEMENT

Fully implemented, CBM+ will help predict a system’s remaining operational life span, support operator decision-making, interface with control systems, aid in guiding maintenance repair actions, and provide feedback to the logistics support and system design communities, all of which are areas of concentration being impacted by AFRL’s S&T research in Health Management. The AFRL systems engineering design concept emerging to satisfy CBM+ capability requirements is called Integrated Systems Health Management (ISHM). AFRL has embarked on a collaborative effort to develop a strategy and technology

development plan that will result in ISHM capabilities for Air Force (AF) systems. The AFRL ISHM initiative is planned to continue through fiscal year 2017 as depicted in Figure 5.25. ISHM encompasses the use of engineering, performance, test, inspection, and maintenance data, which is required to extract factors related to the various system and knowledge-based processes called out by CBM+. EHM is that sub-component of Integrated Systems Health Management that concentrates on the propulsion system.

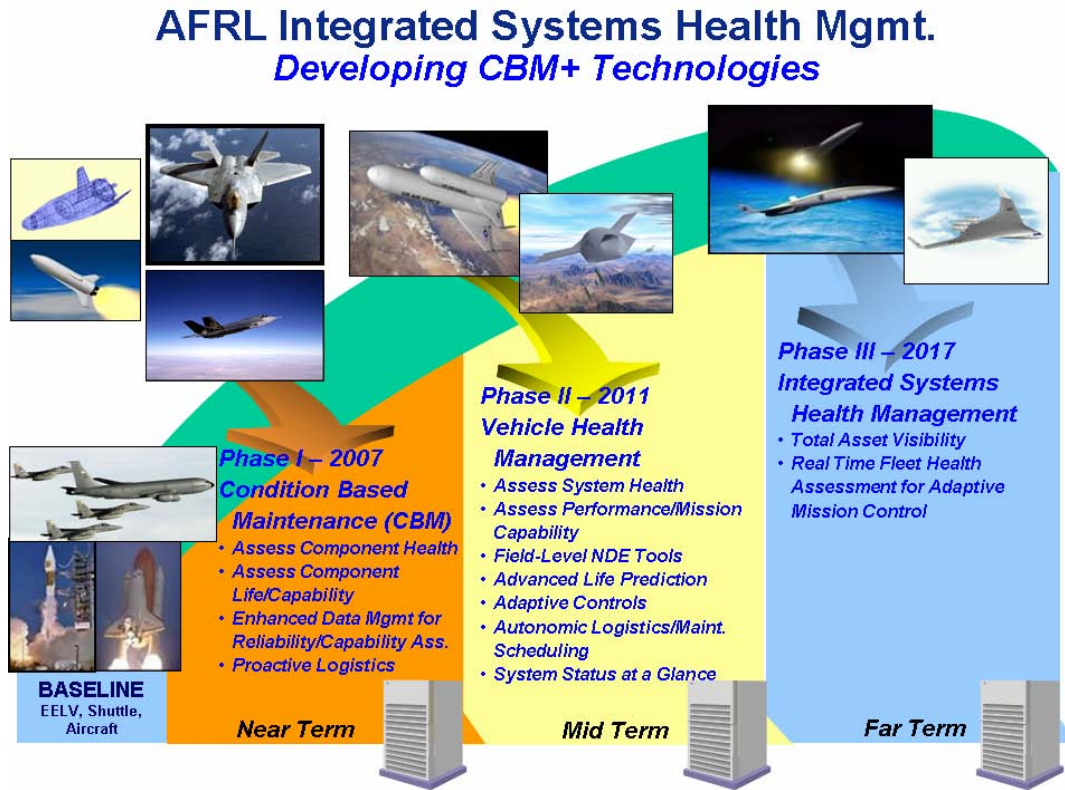


Figure 5.25: AFRL ISHM Program Initiative: Depicts AFRL ISHM CBM+ Technology Programs by Capability Development through Fiscal Year 2017.

Progression to ISHM under CBM+ will require changing business processes throughout the systems engineering process from the inception to the operation stage. To be clear, all disciplines involved, from S&T to development and sustainment, of the weapon system including support equipment and information systems, must forge new relationships and work together to realize the benefits of ISHM.

Under the ISHM umbrella, AFRL is seeking advanced capabilities such as enhanced prognostic and diagnostic techniques, advanced algorithms for failure trend analysis, electronic portable or point of maintenance aids, serial item management, automatic identification technology and data-driven interactive trouble-shooting and maintenance training. The ultimate ISHM vision is to have a fully integrated system operation, providing inputs and outputs for an autonomic system-of-systems, supporting both air and ground operations. The intent of ISHM is to increase operational readiness, and system performance, ensuring mission effectiveness regardless of the degradation state of the system. Payoffs include reduced life cycle costs by enabling a more responsive logistics system.

5.9.1 ISHM Research Requirements

The AF is transforming logistics processes to become lighter and leaner, to develop a more responsive planning and execution capability, to achieve an agile, effective sustainment process, and to develop

responsive, effective, fully integrated joint operations. Military success in the 21st century will require, advanced capabilities, superior speed, power, precision, endurance, and agility. The Science and Technology (S&T) community is working to develop technologies with substantial operational and agile combat support pay-offs. Laboratory technologies, being developed under the ISHM initiative, will revolutionize current AF maintenance philosophy by removing legislated intervals for programmed depot maintenance, by changing maintenance to only be when and where needed, and by implementing unified depot/field mission management concepts. The Laboratory tools and technologies will help Air Force logisticians obtain their desired goals for increasing equipment availability and reducing annual operations and support costs by 10%. The Laboratory is charged with helping the logistics community to reduce sustainment costs, so that more of the AF budget can be used for modernization.

AFRL is developing revolutionary data exploitation techniques, advanced prognostic and diagnostic capabilities, electronic portable maintenance aids, serial item management tools, automatic identification technologies, data-driven interactive trouble-shooting methods, new maintenance training concepts, adaptive planning modules, new sensors, and life prediction models.

In addition to linking AFRL ISHM program objectives to DoD and Air Force Strategic documents this research is conducted to benefit future weapon systems. Within the laboratory environment there are a plethora of programs targeting health management capability gaps that fall under the on-board and/or the off-board system management IT purview. These AFRL programs were born predominantly out of capability needs generated or expressed by weapon system acquisition programs for emerging system concepts, like the Long Range Strike Aircraft (LRS), or systems that are entering the early stages of systems engineering for development. These acquisition programs work with AFRL during their programs requirements development phase on establishing S&T requirements to address their known capability gaps.

A logistics concept being developed for sustaining future Air Force new weapon systems acquisitions is called autonomic logistics. The autonomic logistics concept entails automated automatic support capability needs via IT for sustaining support operations. In this regard, the IT system will be used by all support and operation functions required to manage equipment assets, create and implement mission plans, as well as maintain weapon- system- specific- pilot and -maintenance training records.

Additionally, this system as depicted in Figure 5.26 [McRuer, D. T. and E. S. Krendel] above will collect the gamut of data needed to support the data integration concepts required for ISHM and CBM+ to be fully realized. The prototypes and demonstrations resulting from AFRL ISHM programs will lay the foundation for system engineering of individual AFRL capabilities into the design of this future IT system.

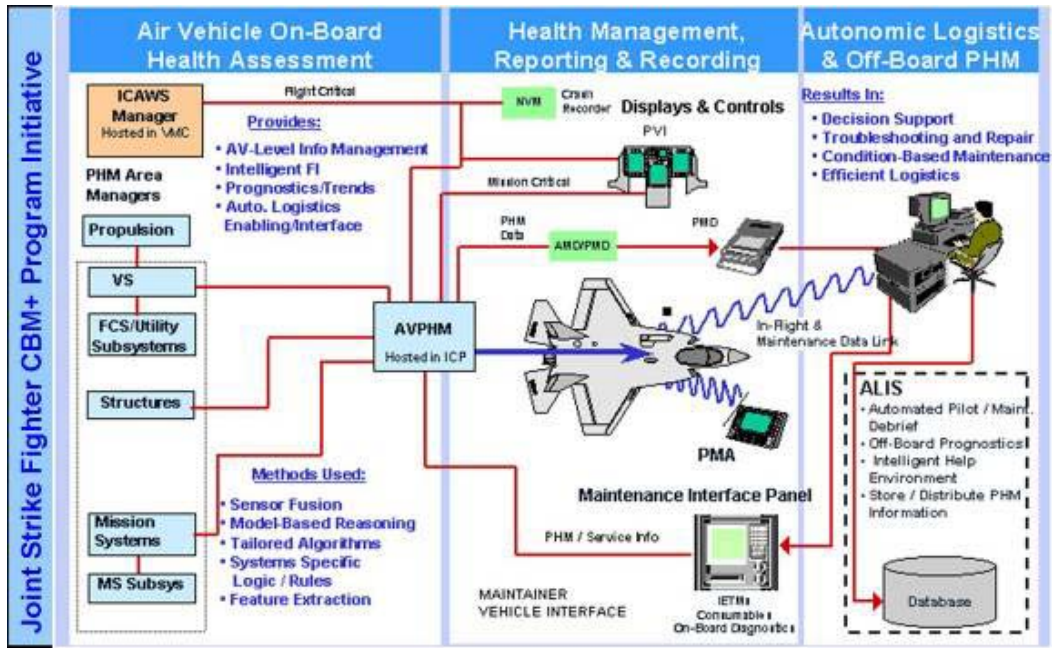


Figure 5.26: Vision for an Autonomic Information Infrastructure that Fully Integrates and Automates Logistics Functions.

The future weapon system programs discussed above are just examples of where AFRL managers obtain project ideas for research initiatives relating to ISHM IT needs. However, for most of these research programs, legacy system data is frequently used because acquisition weapon systems typically do not have or guard their data very closely or the data is not available for research purposes. Another data problem, in respect to new acquisition programs, is the lack of sufficient data needed to perform experiments that fully exercise newly developed AFRL e-tool capabilities. In fact, most of the AFRL IT [knowledge-based capability] research for new acquisition weapon systems is conducted using sample data from legacy weapon systems. Under this practice sufficient data can be obtained to test capabilities to the confidence level required for proving a concept and to satisfy the justification needed for a successful technology transition. Additionally, utilizing legacy systems data provides S&T programs a second target for potential capability transition.

5.9.2 AFRL ISHM Programs

Throughout AFRL, Directorates are creating and managing S&T programs for advanced on-board and ground-based ISHM IT capabilities. These programs focus on developing advanced information fusion and data integration concepts, which use performance data from embedded sensors in combination with test, servicing, and repair data to determine vehicle or system health. Emerging from these concepts is the development of advanced algorithms to satisfy system and Information Technology e-tool capability needs. The resulting prototypes demonstrate innovative capabilities for needs such as trending and predictive failure analysis for maintenance servicing and repair recommendations.

The intention is for these AFRL ISHM developed capabilities to enable the maintenance community in moving forward to a proactive maintenance construct as called out in the new DoD 5000.2. It is envisioned that the health management concepts and/or products resulting from laboratory programs will enable maintainers to quickly assess and determine the best means to prevent or repair weapon system problems. The overarching ISHM ground-based system objective focuses on delivering capabilities for managers of weapon systems to ascertain and assess new information, through data exploitation, about the current condition of the equipment they manage fleet-wide. As AFRL health management concepts

mature, it is expected that new software application capabilities for Air Force logistics systems will also emerge.

The discussion thus far has covered very basic information on how and why AFRL program initiatives are generated and research targets are selected. There is much more planning, detail, and networking that go into those processes for program concept approval. Those process details will not be covered in this paper.

5.9.3 Obstacles to Health Management Research

Researchers require a significant amount of mature systems data to achieve a high level of confidence in the predictive methodologies, so that transition of a capability can occur. Yet, the primary obstacles to health management research are both data and technology transition related. For data, the issues are accessibility, data management policy, and the current data systems used in capturing data. The issues with data are so significant that AFRL managers begin coordinating and planning to meet data needs simultaneously to developing the research program content. The most pressing issue with transitioning advanced health management concepts to information technology programs is there lacks a formal process within the IT systems acquisition community for AFRL to engage in.

5.10 SYSTEM TEST, VERIFICATION AND VALIDATION AND CERTIFICATION ISSUES

Here we present validation, verification technique within the context of mission management.

5.10.1 Basic Definition

Verification: Ensuring that each step in the process yields the right product.

Validation: Ensuring the product will satisfy functional and other requirements.

5.10.2 Verification Techniques

There are many different verification techniques but they all basically fall into 2 major categories – dynamic testing and static testing.

- **Dynamic Testing** – Testing that involves the execution of a system or component. Basically, a number of test cases are chosen, where each test case consists of test data. These input test cases are used to determine output test results. Dynamic testing can be further divided into three categories – functional testing, structural testing, and random testing.
- **Functional Testing** – Testing that involves identifying and testing all the functions of the system as defined within the requirements. This form of testing is an example of black-box testing since it involves no knowledge of the implementation of the system.
- **Structural Testing** – Testing that has full knowledge of the implementation of the system and is an example of white-box testing. It uses the information from the internal structure of a system to devise tests to check the operation of individual components. Functional and structural testing both chooses test cases that investigate a particular characteristic of the system.
- **Random Testing** – Testing that freely chooses test cases among the set of all possible test cases. The use of randomly determined inputs can detect faults that go undetected by other systematic testing techniques. Exhaustive testing, where the input test cases consists of every possible set of input values, is a form of random testing. Although exhaustive testing performed at every stage in the life cycle results in a complete verification of the system, it is realistically impossible to accomplish.

- **Static Testing** – Testing that does not involve the operation of the system or component. Some of these techniques are performed manually while others are automated. Static testing can be further divided into 2 categories – techniques that analyze consistency and techniques that measure some program property.
- **Consistency Techniques** – Techniques that are used to insure program properties such as correct syntax, correct parameter matching between procedures, correct typing, and correct requirements and specifications translation.
- **Measurement Techniques** – Techniques that measure properties such as error proneness, understandability, and well-structuredness.

5.10.3 Validation Techniques

There are also numerous validation techniques, including formal methods, fault injection, and dependability analysis. Validation usually takes place at the end of the development cycle, and looks at the complete system as opposed to verification, which focuses on smaller sub-systems.

- **Formal Methods** – Formal methods is not only a verification technique but also a validation technique. A formal method means the use of mathematical and logical techniques to express, investigate, and analyze the specification, design, documentation, and behaviour of both hardware and software.
- **Fault Injection** – Fault injection is the intentional activation of faults by either hardware or software means to observe the system operation under fault conditions.
- **Hardware Fault Injection** – Can also be called physical fault injection because we are actually injecting faults into the physical hardware.
- **Software Fault Injection** – Errors are injected into the memory of the computer by software techniques. Software fault injection is basically a simulation of hardware fault injection.
- **Dependability Analysis** – Dependability analysis involves identifying hazards and then proposing methods that reduces the risk of the hazard occurring.
- **Hazard Analysis** – Involves using guidelines to identify hazards, their root causes, and possible countermeasures.
- **Risk Analysis** – Takes hazard analysis further by identifying the possible consequences of each hazard and their probability of occurring.

Scope:

For any system development several phases must be followed as shown in Figure 5.27. These phases are part of the verification, and validation process.

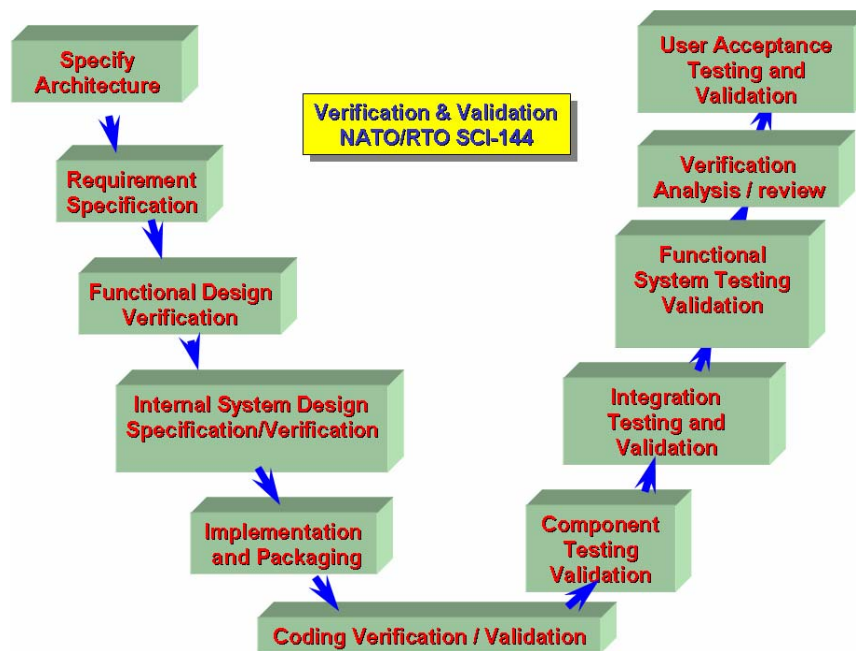


Figure 5.27: Generic Verification and Validation Process shown as “V”.

The IEEE Standard for Software Verification and Validation (IEEE Std 1012-1998) contains information on software integrity levels, the V & V process, the Software V & V reporting, administrative, and documentation requirements, and an outline of the software verification and validation plan.

Verification and validation can be performed by the same organization performing the design, development, and implementation but sometimes it is performed by an independent testing agency. This is called independent verification and validation (IV & V). These agencies usually need to be accredited by a higher organization, to be sure that their results are dependable. For example, in the United Kingdom, the National Measurement Accreditation Service has begun to accredit companies for testing computer software used in safety-critical systems. The first company was accredited in 1994. The testing methods approved include a suite of in-house procedures including static and dynamic testing techniques.

5.10.4 Certification Process

Verification and validation are part of the long certification process for any embedded system. There are different reasons why a product needs certification. Sometimes certification is required for legal reasons. For example, before an aircraft is allowed to fly, it must obtain a license. Being certified would also be important for commercial reasons like having a sales advantage. One of the main reasons for certification is to show competence in specific areas. Certifications are usually carried out by independent agencies or other organizations with a national standing.

Certification can be applied to organizations or individuals, tools or methods, or systems or products. Certification of organizations aims at assuring that the organization achieves a certain level or proficiency and that they agree to certain standards or criteria. This however, is not applicable to all areas because while it is easy to measure the procedures of a company, it is much harder to measure the competence with which they are performed. So certification is usually applied to areas such as quality assurance and testing as opposed to design. Certification may also apply to individuals where workers must be certified in order to be a certain profession. This usually applies to workers such as doctors, lawyers, accountants, and civil engineers. Tools or methods may also be certified. For example, although DO-178B does not specifically define the tools that must be used, it does give certain requirements of tools used to gain certification.

Finally, systems or products may also be certified. [Storey96] In certification, there is always the issue of whether artifacts or methodology be certified. This becomes an issue in the certification of products containing software. Because software testing is so difficult, certification must be based on the process of development and on the demonstrated performance. This is a case where the methodology (development process) is certified instead of the artifact (software).

Review and analyses are performed on the following different components.

- **Requirements Analyses** – To detect and report requirements errors that may have surfaced during the software requirements and design process.
- **Software Architecture** – To detect and report errors that occurred during the development of the software architecture.
- **Source Code** – To detect and report errors that developed during source coding.
- **Outputs of the Integration Process** – To ensure that the result of the integration process is complete and correct.
- **Test Cases and their Procedures and Results** – To ensure that the testing is performed accurately and completely.

The 2 main objectives of the software testing process are to demonstrate that it satisfies all the requirements and to demonstrate that errors leading to unacceptable failure conditions are removed. The testing process includes the following three different types of testing.

- **Hardware/Software Integration Testing** – To verify that the software is operating correctly in the computer environment.
- **Software Integration Testing** – To verify the interrelationships between the software requirements and components and to verify the implementation of the requirements and components in the software architecture.
- **Low-level Testing** – To verify the implementation of software low-level requirements.

5.11 SUMMARY

This chapter describes the overall approach to develop technology for an intelligent mission management system to provide greater robustness. Robustness and fault tolerance are mission-driven requirements. As control technology progresses, improving system performance becomes more complex. Therefore, the price for achieving high performance and robust control is additional complexity of the controller. Fault diagnosis and fault tolerant control are critical component of complex systems.

Autonomous operation requires critical mission management capability for success. Consequently, data must be analyzed and interpreted quickly to yield results that can be implemented to enhance performance in mission operation. Automation can augment or replace human decision making in order to increase reaction speeds, reduce errors, stress, mitigate cognitive overload, enhance safety, and lower costs. While human decision making may not be totally eliminated, the human element fails to offer high capability under continuous stress and integration with logistics infrastructures. However, achieving the goals of automation will require verification and validation techniques to achieve success. Verification and validation is an essential component of a systematic approach.

We live in an information rich environment; however, success often depends on limited access to appropriate information. Information processing speed, accuracy and robustness are critical to achieving successful control and prognostic capability. Control is a critical technology for an integrated system management and is increasingly important in for intelligent operation and management. Control allows the

operation of autonomous and semiautonomous unmanned systems that keep the systems operational, as well as sophisticated command and control systems that enable robust, reconfigurable decision-making systems. The technology is still evolving and in the future we can expect to see the programming platforms for systems placing these techniques within the reach. Until then, it is reasonable to expect future distributed systems capability using electronic control. The increased processing power and intelligence is likely to improve and will lead to robust systems that will be more fault tolerant. The pervasiveness of electronics, communications, computing and sensing will enable many new applications for autonomous operation, but will also require substantial expansion of the current theory and tools.

Chapter 6 – CONCLUSIONS AND RECOMMENDATIONS

The work of SCI Task Group 144 has been presented in detail in the previous chapters. The original title of the activity was “System-Level Integration of Control, Plus Automation”. There was a presumption that development of unmanned vehicles required that the control system would need to be addressed at the system level. As the group members met and exchanged initial ideas, it became a consensus that a general re-focus would be more productive. The emphasis of the work was changed to “Integration of Systems with Varying Levels of Autonomy”. The effort has been to collect and correlate experiences in the design and operation of unmanned vehicles, but also in other areas where autonomy has a primary impact.

The report begins with a historical background starting with man-machine integration. This presents data from the many years of addressing pilot-aircraft interactions. There are then case studies presented discussing examples for land, sea, air and space vehicles. The intent is to discuss success and problems that have been found, plus show similarities across this range of vehicles and any differences. This leads into a chapter discussing systems engineering and then to recommended best practices. Then the report presents supporting information across the complete range of the subject.

This final chapter presents a simple background discussion of the original focus, i.e. the incorporation of an increasing number of automatic functions into flight control systems. There is then a discussion of the issues of systems of systems of unmanned vehicles. The final recommendations are very generic.

6.1 BACKGROUND: ACHIEVEMENTS IN FLIGHT CONTROL SYSTEM DESIGN AND THEIR USE FOR FUTURE AIRCRAFT

The technical progress is defined by the requirements to the production and goals in their development, by achievements in different technologies, sciences and means used for realization of the goals. The main goals and requirements in aviation are the improvement of flight performances (flight effectiveness) and flight safety; all these obtained by improvements in aircraft design, electronics, simulation, feedback control theory, hydraulics, aerodynamics, etc. Many new principles and tools were developed in flight control system design in the area of automation, and will be used in design of the next generation of aircraft. Some of them are briefly summarized below:

- Increased number of functions of control fulfilled by automatic systems. The limitation of maximum bank or pitch angle, normal acceleration, improvement of handling qualities, collision avoidance and others are related to such functions. The number of such control functions for the different periods of recent history in development of civil aircraft is shown in Figure 6.1 for illustration. The tendency shown leads to conclusion that in the near future, if the tendency continues, then aircraft flight will be automatic. The pilot’s function in that case will be limited by monitoring only. This requires consideration of transfer from the automatic system to a human operator as discussed in Chapter 2.1.1.
- Improvement of flight performances by the use of flight control system. It is achieved by increasing static instability up to 10 – 20% allowing decreased weight and increased maneuverability. Flight control system provides necessary stability and controllability, in addition, active control systems can be used for large transport and passenger aircraft to control the lift force in maneuver and to suppress response to atmospheric turbulence.
- Increased number of control surfaces (for example, canard, thrust vectoring control) allows the implementation of super maneuverable flight at low speeds and high angle of attack and new types of maneuvers including 360° loops. New actuation techniques allow the distribution of control signals among the control surfaces in an optimal fashion as a function of piloting task.

CONCLUSIONS AND RECOMMENDATIONS

Moreover damage of any surface has now built-in hardware redundancy allowing continuation of primary mission.

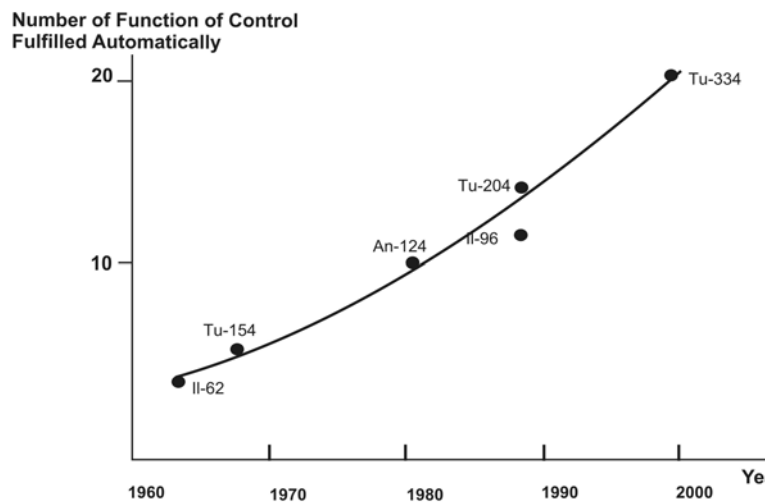


Figure 6.1: Automation Functionalities vs. Time.

- The purpose of a flight control system changes from stability augmentation and flying qualities improvement to provider of necessary/required flying qualities. For the future generation of aircraft this might become the principle of optimization of flying qualities as a function of piloting task and pilot characteristics.
- Usage of fly-by-wire (or fly-by-light) technology. This principle leads to a revolution in flight control system technology: allowing the use of new types of actuators, processors, etc.
- Integration of different control systems: for example integration of flight control with fire control or with critical regime and barrier systems. This principle allows to increase the potentiality of each system, aircraft effectiveness and flight safety.
- Appearance of intelligent systems features, for example the adaptation in flight control system and its elements design. This principle produces increased flight safety and the ability to continue a flight even in case of considerable damage or failure.

6.2 ISSUES AND CHALLENGES

The UAV cooperative team problem can be highly complex, even for relatively small teams. Generally the available theories and approaches can address only one or two aspects of the problem at a time. We are often more interested in a fast, feasible, and robust solution, rather than an optimal one. Since there are many UAV scenarios of moderate size that are of interest, say four to eight vehicles, approaches such as MILP and stochastic dynamic programming may be sufficiently fast at this scale, where a centralized optimal solution is possible. Thus, algorithm scalability may not be the limiting factor.

If more decentralization is desired, the primary limitation is task coupling. Task coupling can be reduced if a myopic or receding horizon procedure is used where not all tasks are addressed up front. However, this can have a significant impact on performance and even feasibility. Also, in the drive for localization, algorithms such as auction and distributed constraint satisfaction can incur extensive message traffic for all but the weakest task coupling. Finally, false information and communication delays can completely negate the benefits of cooperation, similar to losing the benefits of feedback when the sensors are noisy and consequently open loop control is preferable.

It is necessary to study more actively different aspects of human behaviour (manual control, monitoring, decision making) taking into account that integrated, intelligent vehicle control systems has to emulate human capabilities with respect to planning, learning and adaptation.

We have to work for synthesis of concepts, technique and tools defining the information technologies for intelligent system design. The most important goal of such synthesis is to define general scheme that might be called “semi soft computing” (SSC) model. This model must allow to obtain partial models for various branches of the SSC, namely for artificial neural networks, fuzzy logic, generic algorithms, multi-agent systems as well as mathematical modelling by putting into appropriate requirements and conditions.

6.2.1 Systems Level Challenges

Figure 6.2 shows that the work on cooperative control draws from three established disciplines: control, operations research, and computer science, as well as elements of many other disciplines, including estimation, statistics, and economics theory. The research challenge has been to combine these disciplines together to form the beginnings of the new integrated discipline of cooperative control.



Figure 6.2: Cooperation is at the Intersection of Disciplines.

Following is a list of observations made over the course of the research effort that hopefully will provide some insight into what has been done and what yet needs to be done.

- A comprehensive theory of cooperative control must include: uncertainty, communication costs, the consideration of local vs. global information structures, nested vs. non nested information patterns, control decentralization, task coupling, predictive models, adversary action, false information/false targets and false positives, and well reasoned performance measures.
- It is not always beneficial to cooperate, particularly if there are long delays or the sensed or computed information that is communicated is very noisy or error prone.
- It is possible to obtain the optimal solution only for moderately complex operational scenarios.
- Decomposition is a necessary attack upon complexity, but guarantees of optimality are forfeited and, more importantly, oftentimes feasibility is not guaranteed, leading to the possibility of task churning.
- Jacobi auction type assignment algorithms are an iterative form of network flow algorithms which yield a degree of decentralization, but at the expense of possibly much more communications.

CONCLUSIONS AND RECOMMENDATIONS

- The evaluation of different solutions is problematic. Comparisons can be made, but all solutions may be far from the optimum. There is a need to obtain analytical solutions for small scale problems which should help benchmark the array of heuristic “algorithms”/recipes/procedures currently developed.
- Aside from optimality, feasibility is an even more important concern: In general it will not be possible to prove a procedure will not generate infeasible results.
- The “truth” might never be known because in general there are critical states that are not observable. Hence, randomization/mixed strategies are called for, as are Monte Carlo based simulation studies.
- Due to sensitivities to random disturbances, adversary action, operator performance characterization provided in the form of statistical data, and randomized strategies, extensive simulation studies are required for objectively evaluating competing cooperative control schemes.
- As per the employment of feedback, the benefits of cooperative control are questionable when measurement noise and bad or delayed information are dominant factors. Networks are specifically good at rapidly spreading bad information.
- False target attack rate dominates the wide area search and destroy scenario.
- Highly reliable target recognition is *the* critical capability to make autonomous attack vehicles a reality.
- In general, a strongly decentralized controller cannot recover centralized controller performance except if the tasks are nearly independent, that is the optimization problem at hand is virtually decoupled – a rather trivial cooperative control problem.
- For highly coupled tasks, a strongly decentralized controller will need vastly more messages than a centralized controller due to the need to constantly resolve conflicts. This introduces a degree of vulnerability.
- State estimation is the central actor in addressing partial information. In the absence of observability, the illusion is created of being able to provide state estimates from recursive Kalman filters, whereas in reality the provided complete state estimate exclusively hangs on prior information – adding additional states and augmenting the filter does NOT help to update the information about the missing states.
- Only very rarely can optimal performance be achieved with strictly local controllers – unless one can predict the unobserved state.
- Adversary action can be correctly modelled using the game theoretic paradigm. The problem statement is then rigorous, however there are few cases where solutions have been derived.
- Adversaries will resort to information warfare. The objective is to gain information about the true intention of the adversary without revealing any information. This is done by lying, deception, the use of decoys, and diversion tactics/gambits. This further complicates attempts at using game theoretic formulations to realistically address adversarial action.

6.3 CONCLUSIONS

We can start by posing the question: what can be done to support current and future projects? Learning the right lessons from the past can support future projects, by aiming to understand the real reasons for past problems and successes. Exchange of experience, thereby being as open as possible, is strongly recommended. Design cycles are becoming longer, and any designer is faced only with a few designs during his career and, therefore, experience can only partly be gained by learning from the experience of others. To bridge the gaps between projects, an environment has to be established that allows young

engineers to acquire past experience rapidly and reliably. The establishment of databases is recommended that contain bad and good examples of projects from the past. It is important to also consider the establishment of education methods, curricula and training environments in this context.

The system design problem has to be understood as a multidimensional multidisciplinary problem that can only be solved with proper co-operation and mutual understanding between different disciplines and communities. It is therefore important to spend sufficient time at an early stage, to talk to everybody who is involved in the design process, and to consider whether the group has the right constitution. Then the group should understand the 'Best Practices' as documented in Chapter 3. Modern communication and information technology may help to improve the design process, but another real question is how to integrate research communities in order to contribute in this area? New design techniques have been and are being developed, which may aid the designers. An important contribution of the research community could be to make these methods more accessible for the wider design, implementation and testing communities. The gap between science and practical application needs to be narrowed. Modern information and communication technologies could be very helpful in this respect.

CONCLUSIONS AND RECOMMENDATIONS



Chapter 7 – REFERENCES

Ahlstrom, K., J. Torin, K. Fersan, and P. Nobrant, “Redundancy management in distributed flight control systems: experience & simulations”, Dept. of Technol. & Soc., Chalmers Univ., Sweden; Digital Avionics Systems Conference, 2002. Proceedings. The 21st Publication Date: 2002, Volume: 2, On page(s): 13C3-1-13C3-7 Vol.2, ISSN: ISBN: 0-7803-7367-7.

Albus, J.S., “A Reference Model Architecture for Intelligent Systems Design”, published in: *An Introduction to Intelligent and Autonomous Control*, (Antsaklis, P.J., and Passino, K.M., (Eds.)), pp. 27-56, Kluwer Academic, Publishers, Boston, USA. 1993.

Alighanbardi, M., Y. Kuwata, and J. How, “Coordination and Control of Multiple UAVs with Timing Constraints and Loitering,” Proceedings of the 2003 American Control Conference, 2003.

Allen, R. W., and D. T. McRuer, “The effect of adverse visibility on driver steering performance in an automobile simulator”, Proceeding of SAE international Automotive engineering Congress, Detroit, March 1977.

Andersen, P., Lecture notes for 8th Semester Course, “Robust Control User’s Guard/Getting started for Robust Control Toolbox in Matlab”, Department of Control Engineering, Institute of Electronic Systems, Aalborg University, Denmark, U96-4153, April 18, 2001. <http://www.control.auc.dk/~pa/kurser/robust/USnote.pdf>.

Anderson, R. O., “A new approach to the specification and evaluation of flying qualities”, AFFDL-TR-69-120. pp. 60, 1970.

Angelova, A., L. Matthies, D. Helmick, and P. Perona, “Learning slip behavior using automatic mechanical supervision”, IEEE International Conference on Robotics and Automation, 2007.

Anon 1975, “NTSB Assays Iberia Accident at Logan”, 7 April 1975 and “Wind Factor Studied in Iberia Crash”, 14 April 1975, Aviation Week and Space Technology.

Anon 2000, “Flight Control Design – Best Practices”, RTO Technical Report, RTO-TR-29, December 2000.

Anon 2001, *Active Control Technology for Enhanced Performance Operational Capabilities of Military Aircraft, Land Vehicles and Sea Vehicles*, RTO Meeting Proceedings, RTO-MP-051, June 2001.

Anon 2002, “Collaboration for Land, Air, Sea and Space Vehicles: Developing the Common Ground in Vehicle Dynamics, System Identification, Control and Handling Qualities”, RTO Technical Report, RTO-TR-061, November 2002.

Anon 2003a, *Challenges in Dynamics, System Identification, Control and Handling Qualities for Land, Air, Sea and Space Vehicles*, RTO Meeting Proceedings, RTO-MP-095, June 2003.

Anon 2003b, Columbia Accident Investigation Board Report, Volume 1, August 2003, {<http://caib.nasa.gov>}.

Ashkenas, I. L., and D. T. McRuer, “A Theory of Handling Qualities Derived from Pilot-Vehicle System Considerations”, *Aerospace Eng.*, Vol. 21, No. 2, February 1962, pp. 60, 61, 83-102.

REFERENCES

- Avižienis, A., “Framework for a taxonomy of fault-tolerance attributes in computer systems”, Proceedings of the 10th Annual International Symposium on Computer Architecture, Stockholm, Sweden, ISBN: 0-89791-101-6, 1983.
- Bacon, B. J., and Schmidt, D. K., “An Optimal Control Approach to Pilot/Vehicle Analysis and the Neal-Smith Criteria”, AIAA J. of Guidance, Control and Dynamics, Vol. 6, No. 5, 1983.
- Balas, E., and M. Padberg, “Set Partitioning: A Survey”, SIAM Review, Vol.18, pp. 710–760, Society for Industrial and Applied Mathematics. 1976.
- Balas, E. and M.C. Carrera, “A Dynamic Subgradient based Branch and Bound Approach for Set Covering,” Operations Research, Vol. 44, pp. 875–890, 1996.
- Baron S., Kleinman D. and Levison W., “Application of optimal control theory to prediction of human performance in a complex task”, Proceedings of the Fifth NASA-university annual conference on manual control, 1969, NASA-SP 215. P. 367-387.
- Bellutta, P., R. Manduchi, L. Matthies, K. Owens, and A. Rankin, “Terrain Perception for DEMO III” Proceedings of the 2000 Intelligent Vehicles Conference, 2000.
- Belosvet, A., et al., “Stability ‘in gross’ in aircraft-flight control closed loop system for statically unstable aircraft”, Journal of aircraft №1 (Russian).
- Bengtsson, M., “Standardization issues in condition based maintenance”, In Condition Monitoring and Diagnostic Engineering Management – Proceedings of the 16th International Congress, Växjö University, Sweden, 2003.
- Berko, V., et al, “The choice of flight control system parameters with taking into account the effect of flexibility”, Journal of aircraft №1-2 2000 (Russian).
- Bernstein, D.S., S. Zilberstein, and N. Immerman, “The Complexity of Decentralized Control of Markov Decision Processes,” Mathematics of Operations Research, 2002.
- Berthold, M., and D.J. Hand (Eds.), “Intelligent data analysis: An introduction”, 2nd Revised and Extended Edition, Springer-Verlag, Berlin, 2003.
- Bertsekas, D.P., 1988, “The Auction Algorithm: A Distributed Relaxation Method for the Assignment Problem,” Annals of Operations Research, Vol. 14.
- Bertsekas, D.P. and D.A. Castanon, 1989a, “The Auction Algorithm for Transportation Problems,” Annals of Operations Research, Vol. 20.
- Bertsekas, D.P. and J.N. Tsitsiklis, 1989b, “Parallel and Distributed Computation: Numerical Methods,” Prentice-Hall, Englewood Cliffs, N.J.
- Bertsekas, D.P. and D.A. Castanon, 1991, “Parallel Synchronous and Asynchronous Implementations of the Auction Algorithm,” Parallel Computing, Vol.17, pp. 707–732.
- Bertsekas, D., 1992, “Auction Algorithms for Network Flow problems: A Tutorial Introduction,” Computational Optimization and Applications, Vol.1, pp. 7–66.
- Bertsekas, D.P. and D.A. Castanon, 1993a, “Parallel Asynchronous Hungarian Methods for the Assignment Problem,” ORSA Journal on Computing, Vol. 5, No.3.

- Bertsekas, D.P. and D.A. Castanon, 1993b, "Parallel Primal-Dual Methods for the Minimum Cost Network Flow Problem," *Computational Optimization and Applications*, Vol. 2, pp. 319–338.
- Bertsekas, D., D. Castanon, and H. Tsaknakis, 1993c, "Reverse Auction and the Solution of Inequality Constrained Assignment Problems," *SIAM J. on Optimization*, Vol. 3, pp. 268–299.
- Bertsekas, D.P., D.A. Castanon, J. Eckstein, and S. Zenios, 1995, "Parallel Computing in Network Optimization," in *Handbooks in Operations Research: Vol. 7 Network Models*, M. Ball, T. Magnanti, C. Monma, G. Nemhauser editors, pp. 331–396.
- Bertsekas, D.P., 1995, "Nonlinear Programming," Athena Scientific.
- Bushgens, G., et al, 1979, "The longitudinal and lateral dynamics", *Mashinostroenye*, (Russian).
- Bushgens, G., 1990, "Aviation in 21st Century", *Journal of aircraft*, №1 (Russian).
- Bushgens, et al, 1995, "Aerodynamics and dynamics of passenger aircraft", *TSAGI-Chinese Avia published honos*, (Russian).
- Bushgens, G., et al, 2001, "Aerodynamics, stability and controllability of supersonic aircraft", *Nauka*. (Russian).
- Black, G. T. and J. J. Harris, "Interesting Steps Along the Way to a Level 1 Flight Control System Design", *RTO Symposium on Challenges in Dynamic System Identification, Control, and Handling Qualities for Land, Air, Sea, and Space Vehicles*, RTO-MP-095, May 2002.
- Brooks, R. A., "Intelligence without Representation Artificial Intelligence", *Massachusetts Institute of Technology, Artificial Intelligence 47*, pages 139-159, Elsevier, 1991.
- Broten, G., S. P. Monckton, and J. Collier. D. W. Gerhart and R. Grant (ed), "World Representations for Unmanned Ground Vehicles", *Proceedings of SPIE, Unmanned Systems Technology IX, Volume 6561*, May 2007.
- Brusov, V.S., and Yu.V. Tiumentsev, "Advanced information technologies and their application in the aeronautics", 2nd Intern. Sem. Recent Res. and Design Progr. in *Aeron. Eng. and its Influence on Education (RRDPAE'96)*, Warsaw Univ. of Technology, Inst. of Aeron. and Appl. Mech., Warsaw, Poland, Nov.25–26, 1996.
- Brusov V.S., S.A. Piyavsky, and Yu.V. Tiumentsev, "Neural network based tools for uncertainty management with applications to control system design", In: *Generalized Solutions in Control Problems*. Proc. of the IFAC Workshop GSCP-04 / Ed.: Yu.L. Sachkov, Pereslavl-Zalessky, Russia, September 2004.
- Burkard, R.E. and R. Cela, "Linear Assignment Problem and Extensions", *Technical Report 127*, Karl-Franzens University of Graz, Graz, Austria, 1998.
- Burridge, R. R., "Software Architectures for Robots Capable of Intelligent Autonomy: A Survey of the State of the Art", *Annual Forum Proceedings – American Helicopter Society*, Vol. 59, Part 1, pages 396-406, 2003.
- Bushgens, G.S., et al., "Aerodynamics, Stability and Controllability of Supersonic aircraft", pp. 811. *Moscow, Nauka Press* 1998.

REFERENCES

- Cassandras, C.G. and W. Li, "A Receding Horizon Approach for Solving Some Cooperative Control Problems," in Proc. of 41st IEEE Conference on Decision and Control, pp. 3760–3765, 2002.
- Chean, M and J. A. B. Fortes, "A Taxonomy of Reconfiguration Techniques for Fault-Tolerant Processor Arrays", Volume 23, Issue 1 ISBN:0018-9162, January 1990.
- Chen, J., and R. J. Patton, "Robust Model-Based Fault Diagnosis for Dynamic Systems", Kluwer Academic Publishers, Boston, MA, U.S.A., 354 pages, ISBN 0-7923-8411-3, 1999.
- Coste-Manière, É., and R. Simmons, "Architecture, the Backbone of Robotic Systems", Proceedings of the 2000 IEEE International Conference on Robotics & Automation, San Francisco, CA, April 2000.
- Dantzig, G.B., "Linear Programming and Extensions," Princeton University Press, Princeton, N.J., 1963.
- Davidson, J. B., and D. K. Schmidt, "Modified Optimal Control Pilot Model for Computer-Aided Design and Analysis", NASA Technical Memorandum 4384, October 1992.
- Deal, D., "Beyond the Widget: Columbia Accident Lessons Affirmed", published in Air and Space Power, Summer, 2004.
- Deissenberg, C. and F.A. Gonzales, "Pareto Improving Cheating in an Economic Policy Game," Computing in Economics and Finance, April 2001.
- Demirci, U, and F. Kerestecioglu, "Fault Tolerant Control With Re-Configuring Sliding-Mode Schemes", Turkish Journal of Electrical Engineering, Vol. 13, No.1 2005, Tubitak.
- Dillow, J. D., "The Paper pilot – a Digital Computer program to predict pilot rating for the hover task", AFFDL-TR-70-40, 1970.
- Dinnikov, A., et al, 2000, "Analysis of use Neal-Smith criteria for aircraft dynamics estimation in case of flight control system reconfiguration", Journal of aircraft №1-2, (Russian).
- Dinnikov, A., et al, 2001, "The algorithms for fly-by-wire system", TSAGI scientific notes, (Russian).
- Domany, E., J.L. van Hemmen and K. Schultzen (Eds), "Models of neural networks", Springer-Verlag, Berlin, (Physics of Neural Networks Series), 1992.
- Doyle, J.C., B.A. Francis and A.R. Tannenbaum, "Feedback Control Theory", Macmillan Publishing Company, 1992.
- Efremov, A. V., et al, "Evaluation and prediction of aircraft handling qualities", Proceedings of AIAA Atmospheric Flight mechanics conference, Boston. Mass, August 1988.
- Efremov, A.V., et al., "Pilot as a dynamic system", pp. 327. Moscow, Mashinostroenie, 1992.
- Efremov, A. V., and A. V. Ogloblin, "Development and application of the methods for pilot-aircraft system researcher to the manual control tasks of modern vehicles", AGARD Conference Proceedings, *Dual usage in military and commercial technology in Guidance and Control*. pp. 15.1-15.12. Rome. Italy. October 1994.
- Efremov, A. V., and A.V. Ogloblin, 1995, "Development and application of the methods for pilot-aircraft system research to the manual control tasks of modern vehicle", AGARD conference proceedings AGARD CP-556 (English).

- Efremov, A. V., et al, "Investigation of pilot induced oscillation tendency and prediction criteria development", Wright Laboratory WL-TR-96-3109, pp. 1-139, May 1996.
- Ephrath A.R., "Detection of system failure in multi-axis tasks", Proceedings of Eleventh Conference on Manual Control, 1975.
- Feynman, R. P., "Personal Observations on the Reliability of the Shuttle", Report of the Presidential Commission on the Space Shuttle Challenger Accident, Volume II, Appendix F, {<http://science.ksc.nasa.gov/shuttle/missions/51-l/docs/rogers-commission/table-of-contents.html>}, 6 June, 1986.
- Ford, L. R., and D.R. Fulkerson, "Flows in Networks," Princeton University Press, Princeton, N.J., 1962.
- Freeman, P.R., "The Secretary Problem and its Extensions: A Review," *International Statistical Review*, 51 (1983), pp. 189–206.
- Gai E., and R. Curry, "Failure detection of pilots during automatic landing: models and experiments", Proceedings of Eleventh Conference on Manual Control, 1975.
- Gallager, R.J., "Information Theory and Reliable Communication," Wiley 1968, pp. 18.
- Garg, S., and D. Schmidt, "Cooperative Synthesis of control and display augmentation for a STOL aircraft in the approach and landing task", AIAA Paper No. 88-4182. AIAA Guidance, Navigation and Control Conference, Minneapolis, MN, August 1988.
- Gat, E., R. P. Bonnasso and R. Murphy "On Three-Layer Architectures", Artificial Intelligence and Mobile Robots, Aaai Press, 1998.
- Gat, E., "The MDS Autonomous Control Architecture", World Automation Congress, Jet Propulsion Laboratory, California Institute of Technology, April 2000.
- Girard, Anouck R., M. Pachter and P.R. Chandler: "Decision Making Under Uncertainty and Human Operator Model for Small UAV Operations," 2006 AIAA Guidance, Navigation, and Control Conference, August 2006.
- Goldberg, A.V. and R.E. Tarjan, "Solving Minimum Cost Flow Problems by Successive Approximation," *Mathematics of Operations Research*, Vol. 15, pp. 430–466, 1990.
- Goldberg, S.; Maimone, M. & Matthies, L., "Stereo Vision and Rover Navigation Software for Planetary Exploration", IEEE Aerospace Conference Proceedings, 2002.
- Groves, Theodore, "Incentives in Teams," *Econometrica*, Vol. 41, No.4, July 1973.
- Guo, W. and K. Nygard, "Combinatorial Trading Mechanism for Task Allocation," 13th International Conference on Computer Applications in Industry and Engineering, June, 2001.
- Harris, J. J. and G. T. Black, "F-22 Control Law Development and Flying Qualities", AIAA Paper No. 96-3379, August 1996.
- Haykin, S., "Neural networks: A comprehensive foundation", Macmillan College Publishing Company, Inc., New York, 1994.
- Herbert, M. & Krotkov, E., "Local Perception for Mobile Robot Navigation in Natural Terrain: Two Approaches", Workshop on Computer Vision for Space Applications, 1993, 24-31.

REFERENCES

Hess, R., 1978, "Dual Loop Model of the Human Controller", AIAA Journal of Guidance and Control, Vol. 1.

Hess, R., 1979, "Structural model of the adaptive human pilot", AIAA Journal of guidance and control, vol. 3, № 5, pp. 416-423.

Hess, R., 1984, "The effect of time delay, System subject to manual control", Journal of Guidance, Control and Dynamics, Vol. 7, pp. 165-174.

Hess, R., et al, 1998, "Analysis of aircraft handling qualities and pilot induced Oscillation Tendency with actuator Saturation", Paper AIAA-98-4334, AIAA atmospherics Flight Mechanics conference, Boston Mass, USA.

Ho, Y. and K. Chu, "Team decision theory and Information Structures in Optimal Control Problems – Part 1," IEEE Transactions on Automatic Control, Vol. AC-17, pp. 15–22, 1972.

Hoh, R.H., et al, "Development of Handling quality criteria for aircraft with independent control of six degrees of freedom", AFWAL-TR-80 Wight-Patterson AFB, Ohio, p.229, 1980.

Hooks, D., and C. Reising, "Integrated Vehicle Health Management (IVHM)", Presented at NDIA 5th Annual Systems Engineering Conference, 2002.

Jadbabaie, Ali, Jie Lin, and A. Stephen Morse, "Coordination of Groups of Mobile Autonomous Agents Using Nearest Neighbor Rules," IEEE Transactions on Automatic Control, Vol. 48, No.6, June 2003.

Jensen, F., "An Introduction to Bayesian Networks," Springer Verlag, 1996.

Jensen, R. M., Veloso, M. M. and R. E. Bryant, "Fault Tolerant Planning: Toward Probabilistic Uncertainty Models in Symbolic Non-Deterministic Planning", In Proceedings of the 14th International Conference on Automated Planning and Scheduling (ICAPS-04), Vancouver, June 2004.

Jones, R. M., "An Introduction to Cognitive Architectures for Modeling and Simulation", Interservice/Industry Training, Simulation, and Education Conference (I/ITSEC) 2004.

Kanekawa, N, "Dynamic autonomous redundancy management strategy for balanced graceful degradation", Fault-Tolerant Parallel and Distributed Systems, Proceedings of IEEE Workshop on Volume, Issue, June 1994.

Kannan, B., and L. E. Parker, 2006, "Fault-Tolerance Based Metrics for Evaluating System Performance in Multi-Robot Teams", in Proceedings of Performance Metrics for Intelligent Systems Workshop (PerMIS), August 2006.

Kannan, B., and L. E. Parker, 2007, "Metrics for Quantifying System Performance in Intelligent, Fault-tolerant Multi-robot Teams", Carnegie Mellon Univ., Pittsburgh, October - November 2007.

Kelly, A. M., and A. S. Hebert (ed.), "Intelligent Unmanned Ground Vehicles: Autonomous Navigation Research at Carnegie Mellon RANGER: Feedforward Control Approach to Autonomous Navigation", Kluwer Academic Publishers, pages 105-144, 1997.

Kempka, D., J.L. Kennington, and H.A. Zaki, "Performance Characteristics of the Jacobi and the Gauss-Seidel Versions of the Auction Algorithm on the Alliant FX/8," ORSA J. on Computing, Vol. 3, No.2, pp. 92–106, 1991.

- Klein, R. H., and W. F. Clement, "Application of Manual Control Display Theory to the Development of Flight Director Systems for STOL Aircraft", AFFDL-TR-72-152, Wright-Patterson AFB, Ohio.
- Kleinman, D. L., and S Baron, "Analytic Evaluation of Display Requirements for Approach to Landing". NASA-1952, 1971.
- Kluev, M., and I. Kuznetsov, "The development of requirements to dynamic characteristics of maneuverable aircraft actuator", Journal of aircraft №2, 2001 (Russian).
- Konstantinov, S. V., et al, 1999, "The flight safety and some aspects of control system design of advanced maneuver aircraft", New Aviation Technologies of the 21 century, a collection of technical papers 17-22 Aug. 1999, the city of Zukovsky, (English).
- Konstantinov, S. V., and P. Moskalev, 2002, "The peculiarities in development of algorithms and structure of flight control system for perspective modern maneuverable aircraft", Journal of aircraft №3-4, (Russian).
- Kranz, G., "Failure is Not an Option: Mission Control from Mercury to Apollo 13 and Beyond", published by Simon & Schuster, 2000.
- Kuhn, H., "Extensive Games and the Problem of Information," *Annals of Mathematics Studies No.28*, Princeton University Press, Princeton, N.J., 1953.
- Kweon, S., and T. Kanade, "High-Resolution Terrain Map from Multiple Sensor Data", IEEE Transactions on Pattern Analysis and Machine Vision, 1992, 14, 278-2929.
- Lacroix, S.; Mallet, A. & Bonnafous, D., "Autonomous Rover Navigation on Unknown Terrains", Demonstrations in the Space Museum "Cite de l'Espace" at Toulouse Automation, Albuquerque, USA, 1997 7th International Sym. on Experimental Robotics, 2000, 669-683.
- Lala, J. H., and R. E. Harper, "Architectural Principles for Safety-Critical Real-Time Applications," IEEE, Volume 82, Issue 1, Pages 25 – 40, January 1994.
- Lamport, L., "The Part-time Parliament," ACM Transactions on Computer Systems, 16(2):133–169, May 1998. Also appeared as SRC Research Report 49.
- Lehman, J. F., J. Laird and P. Rosenbloom "A Gentle Introduction to Soar, an Architecture for Human Cognition", University Of Southern California Marina Del Rey Information Sciences, Report Number A096413, May 1996.
- Levison, W., Elkind, J., and Ward, J., "Studies of Multivariable Manual Control Systems: A Model of Task Interference", NASA CR-1746, May 1971.
- Llinas, J., and D. Hall, "An Introduction to Multi-sensor Data Fusion", Proceedings of IEEE Int'l Symposium of Circuits and Systems, ISCAS, Vol. 6, May - June 1998.
- Linkens, D.A., and H. O. Nyongess, "Learning systems in intelligent control: An appraisal of fuzzy, neural and genetic algorithm control applications", IEEE Proceedings on Control Theory Application, Vol. 143, No. 4, 1996.
- Lokshin, M., and Y. Shenfinkel, Flight control system for Suhoi aircraft family, The collection of papers of conference 'modern problems in dynamics of flight aerodynamics and flight test', Moscow, Moscow Aviation Institute 2004 (Russian).

REFERENCES

Lou, X-C., A. S. Willsky, and G. C. Verghese, "Optimally Robust Redundancy Relations for Failure Detection in Uncertain Systems" March 1985, M.I.T. Laboratory for Information and Decision Systems, LIDS-P-1447, Automatica (Journal of IFAC) archive Volume 22 , Issue 3, May 1986.

Lovejoy, W., "A Survey of Algorithmic Methods for Partially Observable Markov Decision Processes," Annals of Operations Research, pp. 47–66, 1991.

Luce, R.D. and H. Raiffa, "Games and Decisions: Introduction and Critical Survey," Dover Publications Inc., 1989.

Lussier, B., R. Chatila, F. Ingrand, M-O. Killijian and D. Powell, "On Fault Tolerance and Robustness in Autonomous Systems", LAAS-CNRS, France, In Proceedings of the 3rd IARP-IEEE/RAS-EURON Joint Workshop, 2004.

Lussier, B., A Lampe, R Chatila, J Guiochet, F Ingrand, M-O Killijian, and D Powell, "Fault Tolerance in Autonomous Systems: How and How Much?" www.laas.fr/~mkilliji/docs/workshops/IARP05.pdf

Macedo, J.; Matthies, L. & Manduchi, R., "Ladar-based Discrimination of Grass from Obstacles for Autonomous Navigation", Proceedings of the Seventh International Symposium on Experimental Robotics, 2000.

Marcos, A., S. Ganguli and G. Balas, "Application of H-infinity Fault Detection and Isolation to a Boeing 747-100/200," AIAA Guidance, Navigation and Control Conference, Monterey, CA. 2002.

Marschak, J. and R. Radner, "Economic Theory of Teams," Yale University Press, 1972.

Moravec, H., and A. Elfes, "High-resolution maps from wide-angle sonar", In Proceedings of IEEE Int'l Conf. on Robotics and Automation, St. Louis, Missouri, March 1985.

Mayne, D.Q. and L. Michalska, "Receding Horizon Control of Nonlinear Systems," IEEE Transactions on Automatic Control, Vol. AC-35, No.7, pp. 814–824, 1990.

McLain, T. and R. Beard, "Coordination Variables, Coordination Functions, and Cooperative Timing Missions," AIAA Journal of Guidance, Control, and Dynamics, Vol. 28, No.1, pp. 150–161, January-February 2005.

McRuer, D., D. Graham, E. Krendel, and W. Reisener, Jr., 1965, "Human Pilot Dynamics in Compensatory Systems: Theory, Models, and Experiments with Controlled Element and Forcing Function Variations", AFFDL-TR-65-15.

McRuer, D. T., D. Graham, and E. Krendel, 1967, "Manual Control of Single-Loop Systems", *J. Franklin Inst.*, Vol. 238, No. 1, pp. 1-29; No. 2, pp. 145-168.

McRuer, D. T., H. R. Jex, W. F. Clement, and D. Graham, 1968, "A Systems Analysis Theory for Displays in Manual Control", Systems Technology, Inc., Tech. Report. 163-1.

McRuer, D. T. and E. S. Krendel, "Mathematical Models of Human Pilot Behavior". Advisory Group For Aerospace Research And Development, Paris (France), Descriptive Note : AGARDograph, January 1974.

McRuer, D., 1973a, "Development of Pilot-in-the-Loop Analysis", AIAA J. of Aircraft, Vol. 10, No. 9.

McRuer, D., et al, 1973b, "Measurement of driver/vehicle multiloop response properties with the single disturbance input", Proceedings of 9th annual conference on Manual control.

- McRuer, D., and R. Klein, 1974, "Automobile controllability-driver/vehicle response for steering control", Systems Technology Inc. Report TR 1040-1-1.
- McRuer, D., and R. Klein, 1976, "Comparison of human driver dynamics in an automobile on the road with those in simulators having complex and simple visual displays", STJ paper No. 173A.
- McRuer, D. et al, 1988, "Advanced piloted aircraft flight control system design methodology Vol. 1. Knowledge base", p. 228, NASA CR-191726.
- Michalewicz, Z., "Genetic algorithms + data structures = evolution programs", Third, Revised and Extended Edition, Springer-Verlag, Berlin, 2001.
- Modi, P.J., W. Shen, M. Tambe, and M. Yokoo, "An Asynchronous Complete Method for General Distributed Constraint Optimization," In Proc of Autonomous Agents and Multi-Agent Systems Workshop on Distributed Constraint Reasoning, 2002.
- Moorhouse, D. J., 1977, "Airspeed Control Under Wind Shear Conditions", AIAA Journal of Aircraft, Vol. 14, No. 12, December 1977.
- Moorhouse, D. J., 2004, "Integration of Aerodynamic and Propulsive Flight Control Systems", published in *Vehicle Propulsion Integration*, RTO-MP-AVT-100, August 2004.
- Moravec, H., and A. Elfes, "High-resolution maps from wide-angle sonar", In Proc. IEEE Int'l Conf. on Robotics and Automation, St. Louis, Missouri, March 1985.
- Mullane, M., "Riding Rockets: The Outrageous tales of a Space Shuttle Astronaut", published by Scribner, 2006.
- Murphy, Robert A., "An Approximate Algorithm for a Weapon Target Assignment Stochastic Program," in *Approximation and Complexity in Numerical Optimization: Continuous and Discrete Problems*, Kluwer Academic Publishers, 1999.
- Neal, T. P., and R. E. Smith, "Development of flying qualities criterion for the testing of a fighter flight control systems", AIAA paper № 70-927, 1970, p. 803-809.
- Nemhauser, G. and B. Wolsey, "Integer and Combinatorial Optimization," Wiley, 1999.
- Norton, W. J., "Balancing Modelling & Simulation with Flight Test in Military Aircraft Development", published in AGARD-CP-593, December 1997.
- Nygaard, K., P. Chandler, and M. Pachter, "Dynamic Network Optimization Models for Air Vehicle Resource Allocation," ACC 2001, Arlington, VA, June 2001.
- Obolensky, Y., "Increase of maneuverability by multiaxis thrust vectoring control", The collection of papers of conference 'modern problems in dynamics of flight aerodynamics and flight test', Moscow, Moscow Aviation Institute, (Russian).
- Olfati-Saber, R. and R.M. Murray, "Consensus Protocols for Networks of Dynamic Agents," in Proceedings of the American Control Conference, 2003.
- Olesen, H.R., "A Platform for Model Evaluation", National Environmental Research Institute (NERI), 7th International conference on Harmonisation within Atmospheric Dispersion Modelling for Regulatory Purposes, Belgirate, May 2001. {<http://www.dmu.dk/AtmosphericEnvironment/harmoni/belgirate.htm>}.

REFERENCES

- Pal, S.K., L. Polkowsky, and A. Skowron (Eds.), "Rough-neural computing: Techniques for computing with words", Springer-Verlag, Berlin, (Cognitive Technologies Series) 2003.
- Papadimitriou, C.H. and K. Steiglitz, "Combinatorial Optimization: Algorithms and Complexity," Prentice-Hall, Englewood Cliffs, N.J., 1982.
- Parzen, E., Passino, Kevin M., "Biomimicry for Optimization, Control, and Automation," Springer, 2005.
- Patton, R. J., "Fault Tolerant Control Systems: The 1997 Situation", Proc. IFAC Safe Process, Hull, United Kingdom, pp.1033-1055, 1997. IEEE.
- Pew, R. and A. S. Mavor "Modeling Human and Organizational Behavior," National Academy Press, 1998.
- Piegat, A., "Fuzzy modeling and control", Springer-Verlag, Berlin, (Studies in Fuzziness and Soft Computing, Vol. 69), 2001.
- Polkowsky, L., S. Tsumoto, and T.Y. Lin (Eds.), "Rough set methods and applications: New developments in knowledge discovery in information systems", Springer-Verlag, Berlin, (Studies in Fuzziness and Soft Computing, Vol. 56) 2000.
- Pospelov, D.A., "Situational Control: Theory and practice". Published by Nauka press, (Problems of Artificial Intelligence Series) (In Russian), 1986.
- Puterman, M.L., 1994, "Markov Decision Processes," Wiley Inter Science.
- Puterman, M.L., 2005, "Markov Decision Processes: Discrete Stochastic Dynamic Programming," Wiley Series in Probability and Statistics, Wiley Inter Science.
- Rachinayani, P. K., "Robust Fault-Tolerant Control For Aircraft Systems", Jawaharlal Nehru Technological University, M. S. in Electrical Engineering, August 2006.
- Rasmussen, J., 1983, "Skills, Rules, and Knowledge: Signals, signs, and symbols, and other distinctions in human performance models", IEEE Transactions on Systems, Man and Cybernetics.
- Rasmussen, S.J., P.R. Chandler and C.J. Schumacher, 2002, "Investigation of Single vs. Multiple Task Tour Assignments for UAV Cooperative Control," Proceedings of AIAA Guidance, Navigation & Control Conference, Monterey, CA, August 2002.
- Rasmussen, S.J., P.R. Chandler, J.W. Mitchell, C.J. Schumacher, and A.G. Sparks, 2003, "Optimal vs. Heuristic Assignment of Cooperative Autonomous Unmanned Air Vehicles," Proceedings of the AIAA Guidance, Navigation, and Control Conference, August 2003.
- RayChaudhuri, T., and L.G.C. Hamey, "From Conventional Control to Intelligent Neurocontrol Methods: A Survey of the Literature", Macquarie University, New South Wales, Australia, Technical Report No.95/170C, 1995 {URL: <http://citeseer.nj.nec.com/raychaudhuri95from.html>}.
- Reinelt, G., "The Traveling Salesman Problem: Computational Solns for TSP Applications," Springer Verlag, Berlin, 1994.
- Richards, A. and J. How, "Aircraft Trajectory Planning with Collision Avoidance Using Mixed Integer Linear Programming," American Control Conference, 2002.

- Richards, A., J. Bellingham, M. Tillerson and J. P. How, “Coordination and Control of Multiple UAVs”, AIAA Guidance, Navigation, and Control Conference, AIAA Paper 2002-4588, August 2002.
- Ronco, E., and P. J. Gawthrop, “Neural networks for modelling and control”, University of Glasgow, Centre for System and Control, Technical Report csc97008, November 1997.
- Rosenblatt, J., “DAMN: A Distributed Architecture for Mobile Navigation”, Proceedings of the 1995 AAAI Spring Symposium on Lessons Learned from Implemented Software Architectures for Physical Agents, H. Hexmoor and D. Kortenkamp (Eds.), AAAI Press, Menlo Park, CA, 1995.
- Ross, S.M., “Introduction to Stochastic Dynamic Programming,” Academic Press, 1983.
- Saffiotti, A., “Handling Uncertainty in Control of Autonomous Robots”, Application of Uncertainty Formalisms in Information Systems, Computer Science, Published in A. Hunter and S. Parsons, editors, Uncertainty in Information Systems, Lecture Notes in Computer Science. Springer-Verlag, Berlin, Germany / Heidelberg, Germany / London, UK, 1998.
- Samarin, A. I., “Neural networks with preliminary adjustment”, 7th All-Russia Conference “Neuroinformatics–2005”. Lectures on Neuroinformatics / Ed.: Yu.V. Tiumentsev. – Moscow, MEPhI Press, (In Russian) 2005.
- Sandholm, Thomas, “An Implementation of the Contract Net Protocol Based on Marginal Cost Calculations,” In Eleventh National Conference on Artificial Intelligence (AAAI-93), pp. 256–262, Washington DC, 1993, AAAI.
- Schmidt, D. K., “Pilot Optimal Augmentation for the Air-to-Air Tracking Task”, AIAA Journal of Guidance and Control, Vol. 3, September - October 1980.
- Schumacher, C.J., M. Pachter, P.R. Chandler, and L. Pachter, 2004, “UAV Task Assignment with Timing Constraints via Mixed-Integer Linear Programming,” Proc. AIAA 3rd Unmanned Unlimited Systems Conference.
- Schumacher, C.J., P.R. Chandler, M. Pachter, and L.S. Pachter, 2007, “Air Vehicles Operations Optimization,” Journal of the Operations Research Society.
- Shenfinkel, Y., “Flight control system for aircraft SU-27”, Journal of aircraft №2, 1990 (Russian).
- Siewiorek, D. P., and R. S. Swarz, 1982, “The Theory and Practice of Reliable System Design” Digital Equipment Corporation, Bedford, MA, USA.
- Siewiorek, D. P., and R. S. Swarz, 1998, “Reliable Computer Systems: Design and Evaluation”, A K Peters Ltd; 1998 ISBN: 156881092X.
- Simmons, R., “Structured Control for Autonomous Robots”, IEEE Transactions on Robotics and Automation, 1994.
- Sha, L., “Using Simplicity to Control Complexity”, University of Illinois at Urbana-Champaign, <https://guinness.cs.steventech.edu/~lbernste/papers/Sha%20Reliability%20Equation%20Simplicity%20IEEE%20final.pdf>.
- Smith, R. G. and R. Davis, “Frameworks for Cooperation in Distributed Problem Solving,” IEEE Transactions on Systems, Man, and Cybernetics, January 1981.

REFERENCES

- Sontag, E., “Neural Networks for Control”, In: H. Trentelman and J. Willems, Eds., “Essays on Control: Perspectives in the Theory and its Applications”. Vol.14 of “Progress in Systems and Control Theory”. – Boston, MA: Birkhauser, 1993. { URL: <http://citeseer.nj.nec.com/sontag93neural.html> }
- Stengel, R.F., “Toward intelligent flight control”, IEEE Trans. on Systems, Man, and Cybernetics, Vol. 23, No.6, 1993.
- Suykens, J.A.K., and H. Bersini, “Neural Control Theory: An Overview”, Journal A. Vol. 37, No.3, 1996.
- Swihart, D., and F. Barfield, “An Advanced Automatic Ground Collision Avoidance System for Fighter Aircraft”, presented at the RTO Symposium, *Flight in a Hostile Environment*, Solomons Island, Maryland, USA, October 1999.
- Sycara, K. and J.S. Liu, “Multiagent Coordination in Tightly Coupled Task Scheduling”, International Conference on Multi-Agent Systems, 1996.
- Thompson, P. M., “Minimum Flying Qualities, Vol. III: Program CC’s Implementation of the Human Optimal Control Model”, Systems Technology, Inc., 13766 South Hawthorne Blvd Hawthorne, CA 90250-7083, January 1990.
- Tiumentsev, Yu.V., 2002, “Intelligent autonomous systems as a challenge for information technologies”, Proceedings of the 8th National Conference on Artificial Intelligence (CAI’2002). Vol. 2 – Moscow: Fizmatlit Publishers, (In Russian).
- Tiumentsev, Yu.V., 2004, “Intelligent autonomous systems”, Aerospace Instrumentation, No.10, (In Russian).
- Thrun, S.; Montemerlo, M.; Dahlkamp, H.; Stavens, D.; Aron, A.; Diebel, J.; Fong, P.; Gale, J.; Halpenny, M.; Hoffmann, G.; Lau, K.; Oakley, C.; Palatucci, M.; Pratt, V.; Stang, P.; Strohband, S.; Dupont, C.; Jendrossek, L.; Koelen, C.; Markey, C.; Rummel, C.; van Niekirk, J.; Jensen, E.; Alessandrini, P.; Bradski, G.; Davies, B.; Ettinger, S.; Kaehler, A.; Nefian, A. & Mahoney, P. Stanley, “The Robot That Won the DARPA Grand Challenge”, Journal of Field Robotics, 2006, 23, 661-692.
- Toth, P., and D. Vigo, “The Vehicle Routing Problem,” SIAM, DT09, Softcover Book, · ISBN no.978-0-898715-79-8, 2001.
- Tran, B., and B. Chidambaram, “Health monitoring of an electro hydraulic system”, OSA-CBM Milestone # 5 Review, December 7-8, 2000.
- Tustin, A. (1944), “*An Investigation of the Operator’s Response in Manual Control of a Power Driven Gun*”, Metropolitan-Vickers Electrical Co., Ltd., Attercliffe Common Works, Sheffield, England, C.S. Memorandum No. 169, 22 August 1944.
- Tustin, A. (1947), “The Nature of the Operator’s Response in Manual Control and Its Implications for Controller Design”, *J. IEE*, Vol. 94, Part IIA, No. 2, 1947.
- Vaughan, D., “The Challenger Launch Decision: Risky Technology, Culture, and Deviance at NASA”, published by University of Chicago Press, 1996.
- Veldhuyzen, W., and H. G. Stassen, 1976a,” The internal model: What does it mean in human control?”, Published In: T.B. Sheridan; G. Johanssen. Monitoring Behavior and Supervisory Control. NATO Conf. Series, Plenum, Plenum Press, pp. 139-157.

- Veldhuyzen, W., 1976b, "Ship maneuvering under human control: Analysis of the helmsman's control behavior", Ph.D. Thesis, Department of Mechanical Engineering, Delft University of Technology, 104p.
- Vorobev, N. N., "Game Theory" - Lectures for Economists and Systems Scientists (Applications of Mathematics 7), published by Springer, 1977.
- Walley, M.; M. Freed, M. Takahashi, D. Christian, A. Petterson-Hine, G. Schulein, R. Harris, "The NASA/Army Autonomous Rotorcraft Project", American Society 59th Annual Forum, Phoenix, Arizona, May 6-8, 2003.
- Wein, J. and S.A. Zenios, "Massively Parallel Auction Algorithms for the Assignment Problem," Proc. of 3rd Symposium on the Frontiers of Massively Parallel Computation, Md., November 1990.
- Weir, D. H., and D. T. McRuer, 1970, "Dynamics of driver/vehicle steering control", Automatica. Vol. 6 №1, pp. 87-98.
- Weir, D. H., R. H. Klein, and D. T. McRuer, 1971, "Principles for the Design of Advanced Flight Director Systems Based on the Theory of Manual Control Displays", NASA CR-1748.
- Wellman, M. and P. Wurman, "Market-aware Agents for a Multiagent World," Robotics and Autonomous Systems, 24:115–125, 1998.
- Xie, L., "Output Feedback H-Infinity Control of Systems with Parameter Uncertainty", International J. of Control, 63, pp. 741-750, 1996.
- Yokoo, Makoto, and K. Hirayaman, "Algorithms for Distributed Constraint Satisfaction: A review," Autonomous Agents and Multi-Agent Systems, Vol. 3, No.2, pp. 198–212, 2000.
- Zhou, K., and J. C. Doyle, 1997, "Essentials Of Robust Control", Published by Prentice Hall, Web: <http://www.ee.lsu.edu/kemin/essentials.htm>. ISBN 0-13-525833-2.
- Zhou, D.H. and P.M. Frank, 1998, "Fault Diagnostics and Fault Tolerant Control", IEEE Transactions of Aeronautical Electric Systems 34, pages 420-427.
- Zhou, K, 2005, "A New Approach to robust and Fault Tolerant Control", ACTA Automatica Sinica.
- WEB SITE A, "Control systems related research in Framework Programme 7 (2007-2013), Recommendations for a research agenda in Europe", Publication sponsored by the HYCON Network of Excellence, D R A F T DECEMBER 10, 2005. http://www.ist-hycon.org/documents/ Report_Control_for_FP7-rev5.-jl-hycon-1.pdf.
- WEB SITE B, <http://www.dtic.mil/ndia/2002systems/reisig2b3.pdf>
- WEB SITE C, <http://www.osacbm.org/mission.html>
- WEB SITE D, URL: <http://citeseer.nj.nec.com/suykens96neural.html>
- WEB SITE E, URL: <http://citeseer.nj.nec.com/sontag93neural.html>
- WEB SITE F, http://www.ece.cmu.edu/~koopman/des_s99/verification/

REFERENCES



| REPORT DOCUMENTATION PAGE | | | | | | | | | | | | | | | | | | | | | |
|---|--|--|---|------------------|--------|-------------------|------------|--------------------|------------------|-----------------------|---------------------------|---------------------|-----------------------|--------------|-------------------------------|--------------------------------------|----------|----------------------------|-------------------|--|--|
| 1. Recipient's Reference | 2. Originator's References RTO-TR-SCI-144 AC/323(SCI-144)TP/144 | 3. Further Reference ISBN 978-92-837-0054-8 | 4. Security Classification of Document UNCLASSIFIED/ UNLIMITED | | | | | | | | | | | | | | | | | | |
| 5. Originator Research and Technology Organisation North Atlantic Treaty Organisation BP 25, F-92201 Neuilly-sur-Seine Cedex, France | | | | | | | | | | | | | | | | | | | | | |
| 6. Title Integration of Systems with Varying Levels of Autonomy | | | | | | | | | | | | | | | | | | | | | |
| 7. Presented at/Sponsored by This Report was prepared by Task Group SCI-144 on "System-Level Integration of Control plus Automation" and has been sponsored by the Systems Concepts and Integration Panel. | | | | | | | | | | | | | | | | | | | | | |
| 8. Author(s)/Editor(s) Multiple | | | 9. Date September 2008 | | | | | | | | | | | | | | | | | | |
| 10. Author's/Editor's Address Multiple | | | 11. Pages 150 | | | | | | | | | | | | | | | | | | |
| 12. Distribution Statement There are no restrictions on the distribution of this document. Information about the availability of this and other RTO unclassified publications is given on the back cover. | | | | | | | | | | | | | | | | | | | | | |
| 13. Keywords/Descriptors <table style="width: 100%; border: none;"> <tr> <td style="width: 33%;">Adaptive systems</td> <td style="width: 33%;">Design</td> <td style="width: 33%;">System of systems</td> </tr> <tr> <td>Automation</td> <td>Integrated systems</td> <td>Systems analysis</td> </tr> <tr> <td>Autonomous navigation</td> <td>Operational effectiveness</td> <td>Systems engineering</td> </tr> <tr> <td>Autonomous operations</td> <td>Requirements</td> <td>UAV (Unmanned Aerial Vehicle)</td> </tr> <tr> <td>AUV (Autonomous Underwater Vehicles)</td> <td>Robotics</td> <td>Vehicle management systems</td> </tr> <tr> <td>Control equipment</td> <td></td> <td></td> </tr> </table> | | | | Adaptive systems | Design | System of systems | Automation | Integrated systems | Systems analysis | Autonomous navigation | Operational effectiveness | Systems engineering | Autonomous operations | Requirements | UAV (Unmanned Aerial Vehicle) | AUV (Autonomous Underwater Vehicles) | Robotics | Vehicle management systems | Control equipment | | |
| Adaptive systems | Design | System of systems | | | | | | | | | | | | | | | | | | | |
| Automation | Integrated systems | Systems analysis | | | | | | | | | | | | | | | | | | | |
| Autonomous navigation | Operational effectiveness | Systems engineering | | | | | | | | | | | | | | | | | | | |
| Autonomous operations | Requirements | UAV (Unmanned Aerial Vehicle) | | | | | | | | | | | | | | | | | | | |
| AUV (Autonomous Underwater Vehicles) | Robotics | Vehicle management systems | | | | | | | | | | | | | | | | | | | |
| Control equipment | | | | | | | | | | | | | | | | | | | | | |
| 14. Abstract The report begins with a historical background and the evolution of systems engineering. The report then addresses issues of controlling land, sea and air vehicles in a system of systems. There is a chapter discussing both negative and positive aspects of systems engineering, followed by a presentation of recommended best practices. There is discussion of complexity and automation, man/machine interface, single vs. multiple vehicles, and mission management, especially robust design of autonomous systems. This leads to addressing certification issues, such as verification and validation of non-deterministic systems, followed by discussion of issues and challenges. The report ends with conclusions and suggestions for required future research. | | | | | | | | | | | | | | | | | | | | | |





BP 25

F-92201 NEUILLY-SUR-SEINE CEDEX • FRANCE
Télécopie 0(1)55.61.22.99 • E-mail mailbox@rta.nato.int



DIFFUSION DES PUBLICATIONS
RTO NON CLASSIFIEES

Les publications de l'AGARD et de la RTO peuvent parfois être obtenues auprès des centres nationaux de distribution indiqués ci-dessous. Si vous souhaitez recevoir toutes les publications de la RTO, ou simplement celles qui concernent certains Panels, vous pouvez demander d'être inclus soit à titre personnel, soit au nom de votre organisation, sur la liste d'envoi.

Les publications de la RTO et de l'AGARD sont également en vente auprès des agences de vente indiquées ci-dessous.

Les demandes de documents RTO ou AGARD doivent comporter la dénomination « RTO » ou « AGARD » selon le cas, suivi du numéro de série. Des informations analogues, telles que le titre et la date de publication sont souhaitables.

Si vous souhaitez recevoir une notification électronique de la disponibilité des rapports de la RTO au fur et à mesure de leur publication, vous pouvez consulter notre site Web (www.rto.nato.int) et vous abonner à ce service.

CENTRES DE DIFFUSION NATIONAUX

ALLEMAGNE

Streitkräfteamt / Abteilung III
Fachinformationszentrum der Bundeswehr (FIZBw)
Gorch-Fock-Straße 7, D-53229 Bonn

BELGIQUE

Royal High Institute for Defence – KHID/IRSD/RHID
Management of Scientific & Technological Research
for Defence, National RTO Coordinator
Royal Military Academy – Campus Renaissance
Renaissancelaan 30, 1000 Bruxelles

CANADA

DSIGRD2 – Bibliothécaire des ressources du savoir
R et D pour la défense Canada
Ministère de la Défense nationale
305, rue Rideau, 9^e étage
Ottawa, Ontario K1A 0K2

DANEMARK

Danish Acquisition and Logistics Organization (DALO)
Lautrupbjerg 1-5, 2750 Ballerup

ESPAGNE

SDG TECEN / DGAM
C/ Arturo Soria 289
Madrid 28033

ETATS-UNIS

NASA Center for AeroSpace Information (CASI)
7115 Standard Drive
Hanover, MD 21076-1320

FRANCE

O.N.E.R.A. (ISP)
29, Avenue de la Division Leclerc
BP 72, 92322 Châtillon Cedex

GRECE (Correspondant)

Defence Industry & Research General
Directorate, Research Directorate
Fakinos Base Camp, S.T.G. 1020
Holargos, Athens

HONGRIE

Department for Scientific Analysis
Institute of Military Technology
Ministry of Defence
P O Box 26
H-1525 Budapest

ITALIE

General Secretariat of Defence and
National Armaments Directorate
5th Department – Technological
Research
Via XX Settembre 123
00187 Roma

LUXEMBOURG

Voir Belgique

NORVEGE

Norwegian Defence Research
Establishment
Attn: Biblioteket
P.O. Box 25
NO-2007 Kjeller

PAYS-BAS

Royal Netherlands Military
Academy Library
P.O. Box 90.002
4800 PA Breda

POLOGNE

Centralny Ośrodek Naukowej
Informacji Wojskowej
Al. Jerozolimskie 97
00-909 Warszawa

PORTUGAL

Estado Maior da Força Aérea
SDFA – Centro de Documentação
Alfragide
P-2720 Amadora

REPUBLIQUE TCHEQUE

LOM PRAHA s. p.
o. z. VTÚLaPVO
Mladoboleslavská 944
PO Box 18
197 21 Praha 9

ROUMANIE

Romanian National Distribution
Centre
Armaments Department
9-11, Drumul Taberei Street
Sector 6
061353, Bucharest

ROYAUME-UNI

Dstl Knowledge and Information
Services
Building 247
Porton Down
Salisbury SP4 0JQ

SLOVENIE

Ministry of Defence
Central Registry for EU and
NATO
Vojkova 55
1000 Ljubljana

TURQUIE

Milli Savunma Bakanlığı (MSB)
ARGE ve Teknoloji Dairesi
Başkanlığı
06650 Bakanlıklar
Ankara

AGENCES DE VENTE

NASA Center for AeroSpace Information (CASI)

7115 Standard Drive
Hanover, MD 21076-1320
ETATS-UNIS

The British Library Document Supply Centre

Boston Spa, Wetherby
West Yorkshire LS23 7BQ
ROYAUME-UNI

Canada Institute for Scientific and Technical Information (CISTI)

National Research Council Acquisitions
Montreal Road, Building M-55
Ottawa K1A 0S2, CANADA

Les demandes de documents RTO ou AGARD doivent comporter la dénomination « RTO » ou « AGARD » selon le cas, suivie du numéro de série (par exemple AGARD-AG-315). Des informations analogues, telles que le titre et la date de publication sont souhaitables. Des références bibliographiques complètes ainsi que des résumés des publications RTO et AGARD figurent dans les journaux suivants :

Scientific and Technical Aerospace Reports (STAR)

STAR peut être consulté en ligne au localisateur de ressources
uniformes (URL) suivant: <http://www.sti.nasa.gov/Pubs/star/Star.html>
STAR est édité par CASI dans le cadre du programme
NASA d'information scientifique et technique (STI)
STI Program Office, MS 157A
NASA Langley Research Center
Hampton, Virginia 23681-0001
ETATS-UNIS

Government Reports Announcements & Index (GRA&I)

publié par le National Technical Information Service
Springfield
Virginia 2216
ETATS-UNIS
(accessible également en mode interactif dans la base de
données bibliographiques en ligne du NTIS, et sur CD-ROM)



BP 25

F-92201 NEUILLY-SUR-SEINE CEDEX • FRANCE
Télécopie 0(1)55.61.22.99 • E-mail mailbox@rta.nato.int



DISTRIBUTION OF UNCLASSIFIED RTO PUBLICATIONS

AGARD & RTO publications are sometimes available from the National Distribution Centres listed below. If you wish to receive all RTO reports, or just those relating to one or more specific RTO Panels, they may be willing to include you (or your Organisation) in their distribution.

RTO and AGARD reports may also be purchased from the Sales Agencies listed below.

Requests for RTO or AGARD documents should include the word 'RTO' or 'AGARD', as appropriate, followed by the serial number. Collateral information such as title and publication date is desirable.

If you wish to receive electronic notification of RTO reports as they are published, please visit our website (www.rto.nato.int) from where you can register for this service.

NATIONAL DISTRIBUTION CENTRES

BELGIUM

Royal High Institute for Defence – KHID/IRSD/RHID
Management of Scientific & Technological Research
for Defence, National RTO Coordinator
Royal Military Academy – Campus Renaissance
Renaissancelaan 30
1000 Brussels

CANADA

DRDKIM2 – Knowledge Resources Librarian
Defence R&D Canada
Department of National Defence
305 Rideau Street, 9th Floor
Ottawa, Ontario K1A 0K2

CZECH REPUBLIC

LOM PRAHA s. p.
o. z. VTÚLaPVO
Mladoboleslavská 944
PO Box 18
197 21 Praha 9

DENMARK

Danish Acquisition and Logistics Organization (DALO)
Lautrupbjerg 1-5
2750 Ballerup

FRANCE

O.N.E.R.A. (ISP)
29, Avenue de la Division Leclerc
BP 72, 92322 Châtillon Cedex

GERMANY

Streitkräfteamt / Abteilung III
Fachinformationszentrum der Bundeswehr (FIZBw)
Gorch-Fock-Straße 7
D-53229 Bonn

GREECE (Point of Contact)

Defence Industry & Research General Directorate
Research Directorate, Fakinos Base Camp
S.T.G. 1020
Holargos, Athens

HUNGARY

Department for Scientific Analysis
Institute of Military Technology
Ministry of Defence
P O Box 26
H-1525 Budapest

ITALY

General Secretariat of Defence and
National Armaments Directorate
5th Department – Technological
Research
Via XX Settembre 123
00187 Roma

LUXEMBOURG

See Belgium

NETHERLANDS

Royal Netherlands Military
Academy Library
P.O. Box 90.002
4800 PA Breda

NORWAY

Norwegian Defence Research
Establishment
Attn: Biblioteket
P.O. Box 25
NO-2007 Kjeller

POLAND

Centralny Ośrodek Naukowej
Informacji Wojskowej
Al. Jerozolimskie 97
00-909 Warszawa

PORTUGAL

Estado Maior da Força Aérea
SDFA – Centro de Documentação
Alfragide
P-2720 Amadora

ROMANIA

Romanian National Distribution
Centre
Armaments Department
9-11, Drumul Taberei Street
Sector 6
061353, Bucharest

SLOVENIA

Ministry of Defence
Central Registry for EU and
NATO
Vojkova 55
1000 Ljubljana

SPAIN

SDG TECEN / DGAM
C/ Arturo Soria 289
Madrid 28033

TURKEY

Milli Savunma Bakanlığı (MSB)
ARGE ve Teknoloji Dairesi
Başkanlığı
06650 Bakanlıklar – Ankara

UNITED KINGDOM

Dstl Knowledge and Information
Services
Building 247
Porton Down
Salisbury SP4 0JQ

UNITED STATES

NASA Center for AeroSpace
Information (CASI)
7115 Standard Drive
Hanover, MD 21076-1320

SALES AGENCIES

NASA Center for AeroSpace Information (CASI)

7115 Standard Drive
Hanover, MD 21076-1320
UNITED STATES

The British Library Document Supply Centre

Boston Spa, Wetherby
West Yorkshire LS23 7BQ
UNITED KINGDOM

Canada Institute for Scientific and Technical Information (CISTI)

National Research Council Acquisitions
Montreal Road, Building M-55
Ottawa K1A 0S2, CANADA

Requests for RTO or AGARD documents should include the word 'RTO' or 'AGARD', as appropriate, followed by the serial number (for example AGARD-AG-315). Collateral information such as title and publication date is desirable. Full bibliographical references and abstracts of RTO and AGARD publications are given in the following journals:

Scientific and Technical Aerospace Reports (STAR)

STAR is available on-line at the following uniform resource
locator: <http://www.sti.nasa.gov/Pubs/star/Star.html>
STAR is published by CASI for the NASA Scientific
and Technical Information (STI) Program
STI Program Office, MS 157A
NASA Langley Research Center
Hampton, Virginia 23681-0001
UNITED STATES

Government Reports Announcements & Index (GRA&I)

published by the National Technical Information Service
Springfield
Virginia 2216
UNITED STATES
(also available online in the NTIS Bibliographic Database
or on CD-ROM)