Immunity-based detection, identification, and evaluation of aircraft sub-system failures

Hever Y. Moncayo

West Virginia University

Follow this and additional works at: https://researchrepository.wvu.edu/etd

Recommended Citation
Moncayo, Hever Y., "Immunity-based detection, identification, and evaluation of aircraft sub-system failures" (2009). Graduate Theses, Dissertations, and Problem Reports. 2928.
https://researchrepository.wvu.edu/etd/2928

This Dissertation is protected by copyright and/or related rights. It has been brought to you by the The Research Repository @ WVU with permission from the rights-holder(s). You are free to use this Dissertation in any way that is permitted by the copyright and related rights legislation that applies to your use. For other uses you must obtain permission from the rights-holder(s) directly, unless additional rights are indicated by a Creative Commons license in the record and/or on the work itself. This Dissertation has been accepted for inclusion in WVU Graduate Theses, Dissertations, and Problem Reports collection by an authorized administrator of The Research Repository @ WVU. For more information, please contact research.repository@mail.wvu.edu.
Immunity-Based Detection, Identification, and Evaluation of Aircraft Sub-System Failures

Hever Y. Moncayo

Dissertation submitted to the College of Engineering and Mineral Resources at West Virginia University in partial fulfillment of the requirements for the degree of

Doctor of Philosophy in Aerospace Engineering

Committee Members: Bojan Cukic, Ph.D. Powsiri Klinkhachorn, Ph.D. Marcello Napolitano, Ph.D. Mario Perhinschi, Ph.D., Chair Jacky Prucz, Ph.D.

Department of Mechanical and Aerospace Engineering

Morgantown, West Virginia 2009

Keywords: Artificial Immune Systems, Artificial Intelligence, Fault Tolerant Control Systems, Failure Detection, Identification and Evaluation.
ABSTRACT

Immunity-Based Detection, Identification, and Evaluation of Aircraft Sub-System Failures

Hever Y. Moncayo

This thesis describes the design, development, and flight-simulation testing of an integrated Artificial Immune System (AIS) for detection, identification, and evaluation of a wide variety of sensor, actuator, propulsion, and structural failures/damages including the prediction of the achievable states and other limitations on performance and handling qualities. The AIS scheme achieves high detection rate and low number of false alarms for all the failure categories considered. Data collected using a motion-based flight simulator are used to define the self for an extended sub-region of the flight envelope. The NASA IFCS F-15 research aircraft model is used and represents a supersonic fighter which includes model following adaptive control laws based on non-linear dynamic inversion and artificial neural network augmentation. The flight simulation tests are designed to analyze and demonstrate the performance of the immunity-based aircraft failure detection, identification and evaluation (FDIE) scheme. A general robustness analysis is also presented by determining the achievable limits for a desired performance in the presence of atmospheric perturbations.

For the purpose of this work, the integrated AIS scheme is implemented based on three main components. The first component performs the detection when one of the considered failures is present in the system. The second component consists in the identification of the failure category and the classification according to the failed element. During the third phase a general evaluation of the failure is performed with the estimation of the magnitude/severity of the failure and the prediction of its effect on reducing the flight envelope of the aircraft system.

Solutions and alternatives to specific design issues of the AIS scheme, such as data clustering and empty space optimization, data fusion and duplication removal, definition of features, dimensionality reduction, and selection of cluster/detector shape are also analyzed in this thesis. They showed to have an important effect on detection performance and are a critical aspect when designing the configuration of the AIS.

The results presented in this thesis show that the AIS paradigm addresses directly the complexity and multi-dimensionality associated with a damaged aircraft dynamic response and provides the tools necessary for a comprehensive/integrated solution to the FDIE problem. Excellent detection, identification, and evaluation performance has been recorded for all types of failures considered. The implementation of the proposed AIS-based scheme can potentially have a significant impact on the safety of aircraft operation. The output information obtained from the scheme will be useful to increase pilot situational awareness and determine automated compensation.
DEDICATION

This work is dedicated to my dear wife Paola, my parents Bernardo and Nancy and my siblings Alejo, Daniel and Lady.
ACKNOWLEDGEMENTS

I would like to give special thanks to my advisor, Dr. Mario Perhinschi, for his vision, continuous encouragement, patient guidance and support, and for giving me the opportunity of doing a PhD and introduce me to this field of research.

Great thanks go to my Dissertation Committee members, Dr. Bojan Cukic, Dr. Powsiri Klinkhachorn, Dr. Marcello Napolitano, and Dr. Jacky Prucz, for their important comments and helpful suggestions which have been assimilated in this dissertation.

My greatest thanks goes to my wife Paola, my parents Bernardo and Nancy, and the rest of our family, for their unconditional love and for supporting me during my whole academic and non-academic life.

Many and special thanks to the volunteer pilots Dr. Alejandro Posada, Steven Mullins, Steven Hard, and Ondrej Karas for performing the tests in the flight simulator. They have helped me for achieving the results present in this thesis.

Finally, the financial support provided by NASA Aviation Safety Program through a grant within the Integrated Resilient Aircraft Control project is gratefully acknowledged.
# Table of Contents

Abstract ......................................................................................................................................................... ii  
Dedication ................................................................................................................................................... iii  
Acknowledgements .................................................................................................................................... iv  
List of Tables .................................................................................................................................................. xii  
Nomenclature ................................................................................................................................................ xiii  

Chapter 1 Introduction .................................................................................................................................... 1

Chapter 2 Literature Review .......................................................................................................................... 5

Chapter 3 Aircraft Sub-System FDIE Problem ......................................................................................... 11

Chapter 4 Aircraft Sub-System Failure Modeling .................................................................................... 15

  4.1 - Aircraft Model .................................................................................................................................... 15

  4.2 - Failure Modeling .................................................................................................................................. 15

    4.2.1 - Actuator Failure Modeling ......................................................................................................... 15

    4.2.2 - Sensor Failure Modeling ............................................................................................................ 16

    4.2.3 - Structural Failure Modeling ........................................................................................................ 16

    4.2.4 - Engine Failure Modeling ............................................................................................................ 17

  4.3 - Outline of Simulated Failures ............................................................................................................. 17

Chapter 5 Flight Simulation Testing and Data Acquisition ......................................................................... 19

  5.1 - WVU 6-DOF Flight Simulator ........................................................................................................... 19

  5.2 - Interface of Aircraft Model with the 6 DOF Flight Simulator ....................................................... 19

  5.3 - Simulation Scenarios ....................................................................................................................... 21

Chapter 6 General Architecture of AIS ....................................................................................................... 25

  6.1 - AIS Paradigm ..................................................................................................................................... 25

  6.2 - AIS for Aircraft Sub-System FDIE ................................................................................................... 26

Chapter 7 Antibodies Generation Algorithms ............................................................................................ 28

  7.1 - Binary Valued Matching Rules ........................................................................................................ 28

  7.2 - Real Valued Matching Rules .......................................................................................................... 29

    7.2.1 - Detector Generation via Negative Selection ............................................................................. 30
Chapter 11 Conclusions and Future Work ................................................................. 104
References .................................................................................................................. 109
List of Figures

Figure 2.1 Generation of T-Cells in the Thymus ................................................................. 6
Figure 2.2 Outline of a Typical Negative Selection Algorithm (Forrest et al., 1994) ....... 7
Figure 3.1 Aircraft Sub-System FDIE Problem (Perhinschi, 2009) ............................... 14
Figure 5.1 The WVU 6 – DOF Flight Simulator System ................................................... 20
Figure 5.2 The WVU 6 – DOF Flight Simulator Cabin .................................................... 20
Figure 5.3 Interface of the WVU Flight Simulator with External Models (Perhinschi and
Napolitano, 2009) .......................................................................................................... 20
Figure 5.4 Top Level Simulink Model Interfaced with WVU 6-DOF Flight Simulator
(Sagoo, 2008) ................................................................................................................ 21
Figure 5.5 Testing Flight Envelope ................................................................................... 22
Figure 6.1 Artificial Immune System-Based Abnormal Condition Detection ............... 26
Figure 6.2 AIS – Based FDIE .......................................................................................... 27
Figure 7.1 The Basic NSA-R with Fixed Detectors for a 2-Dimensional Self Version Data
Set ................................................................................................................................... 31
Figure 7.2 The Modified NSA-R with Mobile Detectors for a 2-Dimensional Self Version
Data Set .......................................................................................................................... 32
Figure 7.3 The Variable Detector NSA-R for a 2-Dimensional Self Version Data Set
(1000 detectors) ............................................................................................................ 33
Figure 7.4 The Variable Detector NSA-R for a 3-Dimensional Self Version Data Set...... 34
Figure 7.5 Enhanced NSA-R with Variable Detectors ..................................................... 36
Figure 7.6 Interactive Design Environment of the IFDOT (Davis and Perhinschi, 2009)
........................................................................................................................................ 39
Figure 7.7 Main Menu of the IFDOT Design Environment (Davis and Perhinschi, 2009)
........................................................................................................................................ 40
Figure 7.8 Hyper-Spheres Detectors Coverage ............................................................... 42
Figure 7.9 Hyper-Spheres Overlapping ......................................................................... 42
Figure 7.10 Hyper-Rectangle Detectors Coverage ......................................................... 43
Figure 9.4 Block Diagram of the Identification Phase of the Proposed HMS Strategy.... 68
Figure 9.5 Signature of Failure of Different Magnitudes in 2-Dimensional Space ........ 69
Figure 9.6 Indirect Evaluation Using the AIS Paradigm ........................................ 71
Figure 10.1 Integrated Artificial Immune System Scheme for the WVU IFCS F-15 Simulink Model................................................................. 72
Figure 10.2 Top Level Simulink Model of the HMS Strategy Implementation .......... 73
Figure 10.3 HMS Detection, Identification and Evaluation Simulink Modules .......... 73
Figure 10.4 HMS Detection Phase Simulink Module.............................................. 74
Figure 10.5 HMS Identification Phase Simulink Module........................................ 74
Figure 10.6 Failure Warning Window for Pilot Interface .................................... 75
Figure 10.7 Defined Self for Envelope Reduction Due to Aileron Failure ............ 83
Figure 10.8 Defined Self for Envelope Reduction Due to Stabilator Failure ... 84
Figure 10.9 Defined Self for Envelope Reduction Due to Rudder Failure ......... 84
Figure 10.10 Evaluator Distribution and Activation when an Aileron Failure is Declared
(Left Aileron Surface Control Locked at 8deg) ............................................. 86
Figure 10.11 Evaluator Distribution and Activation when a Stabilator Failure is Declared
(Right Stabilator Surface Control Locked at 5deg) ......................................... 87
Figure 10.12 Evaluator Distribution and Activation when an Rudder Failure is declared
(Left Rudder Surface Control Locked at 4deg) ............................................ 87
Figure 10.13 Dynamic Behavior for Left Aileron Failure (Locked at 8deg) ...... 89
Figure 10.14 Dynamic Behavior for Left Stabilator Failure (Locked at 8deg) ..... 89
Figure 10.15 Dynamic Behavior for Right Rudder Failure (Locked at 8deg) ......... 90
Figure 10.16 Evaluator Distribution and Activation when a Sensor Failure is declared
(Large Step Bias in the Roll Rate Sensor) ................................................ 91
Figure 10.17 Evaluator Distribution and Activation when a Sensor Failure is declared
(Medium Step Bias in the Pitch Rate Sensor) ............................................. 92
Figure 10.18 Evaluator Distribution and Activation when a Sensor Failure is declared
(Small Step Bias in the Yaw Rate Sensor Failure) ......................................... 92
Figure 10.19 Indirect Evaluation Scheme for Structural Failure ......................... 94
Figure 10.20 Defined Self for Envelope Reduction Due to Structural Failure ....... 95
Figure 10.21 Evaluator Distribution and Activation when a Structural Failure is Declared ................................................................. 95
Figure 10.22 F-15 Flight Envelope for Different Power Settings........................................... 97
Figure 10.23 Defined Self for Envelope Reduction Prediction in the Presence of Propulsion Failures........................................................................................................................................ 97
Figure 10.24 Distribution of Evaluators for Estimation of Flight Envelope Reduction Due to Propulsion System Failure ............................................................................................................. 99
Figure 10.25 Propulsion Failure Impact on Flight Envelope Reduction
(Right Engine Stuck at 60% of Its Total Power)....................................................................... 99
List of Tables

Table 4.1 Simulated Failures ............................................................................................................. 18
Table 8.1 Definition of Eight Different Self Configurations .......................................................... 55
Table 10.1 Feature Configurations for Self Definition ................................................................. 76
Table 10.2 Detection Performance of Different Self Configurations (3 Points of the Flight Envelope) ................................................................................................................. 76
Table 10.3 Detection Performance of Different Self Configurations (Extended Flight Envelope) ................................................................................................................................. 77
Table 10.4 Detection Performance of the HMS Strategy for the Extended F-15 Flight Envelope ................................................................................................................................. 78
Table 10.5 Pre-identification Performance of the HMS Strategy .................................................. 79
Table 10.6 Identification Performance of the HMS Strategy ......................................................... 80
Table 10.7 Evaluation Performance of the HMS Strategy ............................................................. 81
Table 10.8 Nominal Operational Ranges ....................................................................................... 85
Table 10.9 Roll Rate Failure Operational Ranges .......................................................................... 88
Table 10.10 Pitch Rate Failure Operational Ranges ...................................................................... 88
Table 10.11 Yaw Rate Failure Operational Ranges ...................................................................... 88
Table 10.12 Nominal Operational Ranges ...................................................................................... 91
Table 10.13 Roll Rate Sensor Failure Operational Ranges .............................................................. 92
Table 10.14 Pitch Rate Sensor Failure Operational Ranges ............................................................ 93
Table 10.15 Yaw Rate Sensor Failure Operational Ranges ............................................................ 93
Table 10.16 Operational Ranges Under Structural Damage ......................................................... 96
Table 10.17 Flight Envelope Reduction Due to Propulsion Failures at 90000ft ......................... 100
Table 10.18 Flight Envelope Reduction due to Propulsion Failures at 20000ft ......................... 100
Table 10.19 Flight Envelope Reduction due to Propulsion Failures at 31000ft ......................... 100
Table 10.20 Flight Test Condition in the Presence of Turbulence .............................................. 101
Table 10.21 AIS Detection and Identification Performance under Atmospheric Turbulence ................................................................................................................................. 102
Nomenclature

**English**

- \( a \) Acceleration, ft/sec\(^2\)
- \( \text{NN}_{w(x)} \) Specific neural network weight on all three channels
- \( \text{NN}_{\text{outx}} \) Specific neural network output on all three channels
- \( p,q,r \) Measured roll, pitch and yaw rate, rad/sec
- \( R_{\text{py}} \) Roll-pitch cross correlation coefficient
- \( R_{rr} \) Yaw autocorrelation coefficient
- \( x_{\text{TE}} \) Angular rate tracking error on all the three channels
- \( \delta_a \) Aileron deflection (deg)
- \( \delta_e \) Elevator deflection (deg)
- \( \delta_r \) Rudder deflection (deg)

**Greek**

- \( \alpha \) Angle of attack, rad or deg
- \( \beta \) Sideslip angle, rad or deg
- \( \lambda \) Minkowski distance parameter
- \( \varepsilon \) Integration error
- \( \eta \) Confidence level
- \( \delta \) Deflection of control surface

**Subscripts**

- \( T \) T cells
- \( B \) B cells

**Acronyms**

- AIS Artificial Immune System
- ANN Artificial Neural Networks
- DE Direct Evaluation
- DQEE\(_x\) Decentralized Quadratic Estimation Error
- DR Detection Rate
<table>
<thead>
<tr>
<th>Acronym</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ENSA-RV</td>
<td>Enhanced Negative Selection Algorithm - Real Valued with</td>
</tr>
<tr>
<td></td>
<td>Variable Detectors</td>
</tr>
<tr>
<td>FA</td>
<td>False Alarms</td>
</tr>
<tr>
<td>FDI</td>
<td>Failure Detection and Identification</td>
</tr>
<tr>
<td>FDIE</td>
<td>Failure Detection, Identification, and Evaluation</td>
</tr>
<tr>
<td>FEE</td>
<td>Flight Envelope Estimators</td>
</tr>
<tr>
<td>FN</td>
<td>False Negatives</td>
</tr>
<tr>
<td>FP</td>
<td>False Positives</td>
</tr>
<tr>
<td>GA</td>
<td>Genetic Algorithm</td>
</tr>
<tr>
<td>GNC</td>
<td>Guidance, Navigation and Control</td>
</tr>
<tr>
<td>ID</td>
<td>Indirect Evaluation</td>
</tr>
<tr>
<td>IFC</td>
<td>Intelligent Flight Control</td>
</tr>
<tr>
<td>LFDB</td>
<td>Large Fast Drifting Bias</td>
</tr>
<tr>
<td>LSB</td>
<td>Large Step Bias</td>
</tr>
<tr>
<td>MILA</td>
<td>Multilevel Immune Learning Algorithm</td>
</tr>
<tr>
<td>MQEE</td>
<td>Main Quadratic Estimation Error</td>
</tr>
<tr>
<td>NN</td>
<td>Neural Network</td>
</tr>
<tr>
<td>NS</td>
<td>Negative Selection</td>
</tr>
<tr>
<td>NSA</td>
<td>Negative Selection Algorithm</td>
</tr>
<tr>
<td>NSA-RVF</td>
<td>NSA-Real Valued with Fixed Detectors</td>
</tr>
<tr>
<td>NSA-RVM</td>
<td>NSA-Real Valued with Mobile Detectors</td>
</tr>
<tr>
<td>NSA-RVV</td>
<td>NSA-Real Valued with Variable Detectors</td>
</tr>
<tr>
<td>NSDR</td>
<td>Negative Selection with Detection Rules</td>
</tr>
<tr>
<td>OQEE</td>
<td>Output Quadratic Estimation Error</td>
</tr>
<tr>
<td>PS</td>
<td>Positive Selection</td>
</tr>
<tr>
<td>RNS</td>
<td>Real-valued Negative Selection</td>
</tr>
<tr>
<td>RRNS</td>
<td>Randomized Real-valued Negative Selection</td>
</tr>
<tr>
<td>TF</td>
<td>Transfer Function</td>
</tr>
<tr>
<td>TN</td>
<td>True Negatives</td>
</tr>
<tr>
<td>TP</td>
<td>True Positives</td>
</tr>
</tbody>
</table>
Chapter 1 Introduction

The development of fault-tolerant flight control systems has emerged in recent years as a key aspect to ensure increased safety and enhanced performance for both civilian and military aircraft (White, 2006). This new research focus has produced a variety of techniques (KrishnaKumar, 2002a; KrishnaKumar, 2002b; KrishnaKumar, 2003), many of which rely on high performance real-time failure detection and identification (FDI) schemes. A comprehensive integrated solution to the failure detection, identification and evaluation (FDIE) problem for aircraft sub-systems is extremely complex and multi-dimensional requiring adequate tools. A promising candidate in this respect is the Artificial Immune System (AIS) concept (Dasgupta, 1997). The AIS-based fault detection (D’haeseleer, 1996; Dasgupta, 2002b; Forrest, 1994) operates in a similar manner as does the biological immune system - according to the principle of self-non-self discrimination - when it distinguishes between entities that belong to the organism and entities that do not. The basic idea is that an abnormal situation (i.e. failure of one of aircraft sub-systems) can be declared when a current configuration of “features” does not match with any configuration from a pre-determined set known to correspond to normal situations. This paradigm can potentially address directly the complexity and multi-dimensionality of aircraft dynamic response in the context of abnormal conditions and provide the tools necessary for a comprehensive/integrated solution to the FDIE problem.

The AIS concept has shown a promising potential for a variety of applications (Dasgupta, 1999), including fault detection of aerospace systems (KrishnaKumar, 2003; Dasgupta, 2004; Karr, 2005). However, these research efforts have focused on single classes and magnitudes of failure for limited regions of the flight envelope.

The first phase in the immunity-based fault detection process consists of defining the set of features whose values characterize the self – or normal conditions and – for that matter – the non-self, or the abnormal conditions. These features can include any
information expected to be relevant to the behavior of the system. They can be instantaneous samples or time histories over constant or variable windows.

An integrated set of methodologies for AIS-based detection, identification, and evaluation of a wide variety of aircraft sensor, actuator, propulsion, and structural failures/damages is currently under development at West Virginia University (WVU) within NASA’s Aviation Safety Program. As part of this effort, the new artificial intelligence paradigm in combination with other bio-inspired techniques such as evolutionary algorithms and artificial neural networks is currently used to develop comprehensive schemes for aircraft sub-system FDIE (Perhinschi, 2009). Throughout this design process, several critical issues have been identified, such as: data clustering and empty space optimization, data fusion and duplication removal, definition of features and dimensionality reduction, and selection of cluster/detector shape. They are primarily due to the extremely large amounts of experimental data that are necessary to determine the self and to the potentially very high-dimensional feature space. These design issues and their impact on detection performance are described and analyzed in this thesis.

The AIS-based FDIE scheme is capable of detecting and identifying several categories of sub-system abnormal conditions over an extended area of the flight envelope (Moncayo, 2009). The effectiveness of the approach in terms of high detection rate and low number of false alarms for the four categories of failures is tested using data from the WVU motion based flight simulator. The aircraft model represents a supersonic fighter including model following adaptive control laws based on non linear dynamic inversion and artificial neural network augmentation (Perhinschi, 2003).

A brief review of the AIS paradigm and its use for failure detection is presented in Chapter 2. A description of the proposed AIS-based FDIE scheme and a general framework for the AIS-based FDIE scheme are presented in Chapters 3 through 6 including aircraft sub-system failure modeling. Details on the design of the FDIE scheme are provided in Chapters 7 and 8 followed in Chapter 9 and 10 by the test results, analysis, and evaluation of the FDIE scheme performance. Finally, some conclusions are summarized followed by a bibliographical list.

The main contributions of this research effort consist of:
Proposing and demonstrating a novel integrated and comprehensive solution to the problem of FDIE for aircraft sub-systems. This solution addresses a variety of types and severities of several main aircraft sub-systems over large areas of the flight envelope.

Using the AIS for identification of faulty aircraft sub-systems and failure/damage evaluation.

Formulating a novel hierarchical multi-self strategy in developing the AIS for FDIE to mitigate multi-dimensionality issues.

Providing computational tools for AIS design by including all necessary steps within a multi-optional environment for development, optimization, and testing.

The research effort presented in this thesis has resulted in a number of publications or submissions:

*Conference proceedings*


Journals


Chapter 2  Literature Review

The AIS emerged in recent years as a new computational paradigm in artificial intelligence. The concept has shown a very promising potential for a variety of applications such as anomaly detection (Forrest, 1995; Dasgupta, 1996; Dasgupta, 2004; Zhi-tang, 2005; Sanchez, 2009), data mining (Dasgupta, 1996; Dasgupta, 2002a; Dasgupta, 2002c), computer security (De Castro, 2003; Forrest, 1994; Dasgupta, 2002b; Gonzalez, 2002; Harmer, 2002; Kim, 2005), adaptive control (Farmer, 1986; Karr, 2005; Ko, 2004), and pattern recognition (Dasgupta, 2000; De Castro, 2002a).

The immunity-based fault detection (Dasgupta, 1997; KrishnaKumar, 2003; Dasgupta, 2006) operates in a similar manner as does the biological immune system - according to the principle of self-non-self discrimination - when it detects microbial and non-microbial exogenous antigens while not reacting to the self cells. T-cells are the component of the system with the most important role in this process (Benjamini, 1992). T-cells are first generated through a pseudo-random genetic rearrangement mechanism, which ensures high variability of the new cells in terms of biological features (typically proteins or polysaccharides). A censoring process then takes place in the thymus resulting in the destruction of the T-cells that react against self proteins (see Figure 2.1). Eventually, only those T-cells that do not bind to self proteins are allowed to leave the thymus. For obvious reasons this process is referred to as negative selection NS (Forrest, 1994). The matured T-cells can circulate throughout the body to detect antigens and mark them for destruction. Only those cells that recognize the antigen are allowed to proliferate or to produce copies (clones) of themselves. This mechanism is also referred to as clonal selection. Therefore, the NS consists of three phases: defining self, generating detectors, and monitoring the occurrence of anomalies. Alternative mechanisms such as network model (Olivetti, 2005; De Castro, 2002a; De Castro, 2002b) and the danger theory (Aickelin, 2002; Aickelin, 2003) have been explored for AIS. The network model concept was proposed in the mid-seventies and it is based on the interconnections of B cells for
antigen recognition. The stabilization of the network is guaranteed by the stimulation and suppression of these cells. The danger theory refers to the fact that danger and damage can be detected by using a combination of sensing molecules belonging to the invaders and the safe signals within the body.

![Figure 2.1 Generation of T-Cells in the Thymus](image)

Although the NS has been modeled in several ways, the main characteristics of the phenomenon are still being investigated. Figures 2.2 (a) and (b) represent the original conception and describe the major steps in such an algorithm. During generation stage, the detectors are created by some random process and censored by trying to match self samples. The candidates that match are eliminated and the rest are stored as detectors. In the detection phase, the set of detectors is used to classify whether an incoming data instance is self or non-self. If it matches any detector, it is claimed as non-self or an anomaly.
The basic idea of using this new artificial intelligence technique for failure/malfunction detection is that an abnormal situation (i.e. failure of one of aircraft sub-systems) can be declared when a current configuration of “features” does not match with any configuration from a pre-determined set known to correspond to normal situations. Extensive experimental data are necessary to determine the self or the hyperspace of normal conditions. Adequate numerical representations of the self/non-self must be used and the data processed such that they are manageable.

Data representation has an important impact on algorithm effectiveness and performance. It determines the possible matching rules, the detector generation mechanisms, and the detection process. In general, the data to be processed include numeric data, categorical data, Boolean data, and textual data. Data representation schemes have been grouped into two basic types: string representation and real-valued vector representation. In string representation, a detector is represented as a string over a finite alphabet. The length of the string is usually fixed, but it can also be variable. All four data types mentioned above can be processed using the string representation. Binary representation is a special case of string representation that is widely used for AIS applications (Forrest, 1995; Doyne, 1986). Some deterministic generation algorithms have been also designed based on this type of representation. In many cases, they are implemented to study analytically the algorithm complexity or detector coverage (D’haeseler, 1996; Timmis, 2002). Timmis reviewed and compared five detector generation scheme using binary negative selection algorithms (Timmis, 2002): exhaustive,
linear, greedy, binary template, and NS mutation. Some optimization processes were implemented by introducing elimination redundancy to exhaustive NS algorithm for better performance. Gonzalez analyzed also different binary matching rules in negative selection: r-contiguous matching, r-chunk matching, Hamming distance matching, and its variation the Rogers and Tanimoto (R&T) matching (Gonzalez, 2003b). It thus provided a guideline of selecting different matching rules for any negative selection algorithm. New algorithms of detector generation in string representation are still being proposed (Luo, 2006).

Due to the binary implementation within computers, the binary representation may be considered to subsume all other representations. However, a matching rule defined on a high-level representation generally does not translate into a binary matching rule or rules in a straightforward way. It is important for the proper processing of the data that the matching rule reflects the actual Euclidean distance between the components tested. However, other types of distances such as the Minkowski and Manhattan distances have been considered and implemented with promising results (Ji, 2007). Within the real-valued vector representation, each data item is a vector of real numbers (Gonzalez, 2003a). The matching rules and the measure of difference or similarity are based on the numeric elements of the vector. Hybrid representations combining strings and real-valued vectors exist also (Zhang, 2004; Balachandran, 2005) where each data instance may consist of several features of different data types such as integer, real value, categorical information, Boolean value, text information, etc.

On the next level, the type of data representation does not necessarily define the representation or format of detectors. The detector is usually described in the same data space, but more details may be involved unless the detector is purely a point. For example, if a detector is a hyper-sphere in $n$-dimensional real space, then it can be represented by an $n$-dimensional point, the center, and a radius. In one dimension such a detector is a value range, and it can be represented as a pair of values, the two limits of the range.

Based on the representation schemes, different NS algorithms for detector generation have been explored. In particular, real valued representations using hyper-shapes such as hyper-rectangles (Gonzales, 2003c), hyper-spheres (Gonzalez, 2003a; Gonzalez, 2003c), hyper-ellipses (Shapiro, 2005) or a combination of them (Balachandran, 2005) have been studied. Among the detector generation algorithms most discussed are
(Balachandran, 2005; Gonzalez, 2003d): Negative Selection with Detection Rules (NSDR) for hyper-cube detectors generation using evolutionary algorithms, Real-valued Negative Selection (RNS) to generate hyper-spherical detectors using a heuristic algorithm, and Randomized Real-valued Negative Selection (RRNS), an algorithm for generating hyper-spherical detectors using Monte Carlo methods. Also, hybrid immune learning algorithms have been developed as combination of RNS (or RRNS) and classification algorithms (Balachandran, 2005). Although these algorithms include optimization concepts for coverage and overlapping, none of them addresses the issue of the number of detectors necessary to cover the non-self because of the constant size considered for the detectors.

An alternative algorithm called V-detector (Ji, 2004; Ji, 2006), considers variable size of detector during the generation process. By estimating the coverage during the process, this algorithm avoids the situation where the number of detectors needs to be pre-calculated before a certain space is covered. Unlike other real-valued NS algorithms, the size of detectors is maximized until a large and desired coverage is achieved. Thus a lower number of detectors is necessary to cover the same space.

Other algorithms such as the Multilevel Immune Learning Algorithm MILA have been proposed (Dasgupta, 2003a). This algorithm utilizes different mechanisms of AIS in an integrated hybrid model to increase the anomaly detection efficiency.

In general, to make the AIS a practical fault/anomaly detection technique, some specific aspects must be addressed: computational efficiency improvement of the algorithms, enhancement of the representation, and development of unified architectures that can integrate several AIS mechanisms.

Although new models are currently being developed (Dasgupta, 2003b; Dasgupta, 2006) and existing methods are improved continuously, the entire field of AIS including negative selection algorithms is still relatively young and not well defined. Theoretical issues have been occasionally addressed in the attempt to assess and prove AIS applicability (Stibor, 2006); however, there is no systematic theoretical background yet to support the AIS.

Within this thesis, a thorough literature review on AIS and their use for failure detection was performed to identify the most effective algorithms and techniques and directions for performance improvement. Brownlee et al., (2007) have recorded a total of
27 doctoral dissertations and 36 master theses related to AIS (Brownlee, 2007). The immunological computational paradigms most investigated are the negative selection (26%) and the cloning (15%). In preparing the literature review of this thesis, 72 technical papers on AIS were used. An approximate assessment reveals that 33 of them address general AIS design; 10 provide surveys on the development and use of AIS; 19 address computer security applications; 3 address robot control and industrial applications; and 7 papers present applications of the AIS to specific aerospace problems. These focus primarily on aircraft systems fault detection and identification; however, they only have considered single classes and high magnitudes of failure for limited regions of the flight envelope. Therefore, the availability of failure detection, identification, and evaluation schemes with high rates of success, with comprehensive coverage, integrating all aircraft sub-systems and operational modes is a critical objective of this research work.
Chapter 3 Aircraft Sub-System FDIE Problem

Physical redundancy for the actuators of the primary control surfaces is rarely available due to cost and added complexity. Therefore, actuators failures of primary control surfaces may represent major threats to flight safety. On the other side, due to the lower cost and weight of the sensors, flight control systems typically rely on multiple physical redundancy in the sensors. However, for specific type of aircraft for which reduced complexity, reduced weight, and/or reduced costs are critical design issues, it may be appealing to design a flight control system without physical redundancy in the sensor components with some form of analytical redundancy from real time on-line estimates of the dynamic variables. Regulations require that multiple engine aircraft be capable of safe operation if one engine fails. However, information regarding the occurrence of the failure, location, and evaluation of the effect on reducing the envelope are necessary for the pilot and the control system. Structural failures/damages can vary from minor to catastrophic and can potentially have a significant effect on the aerodynamics and hence the control and performance of the aircraft.

Due to the specific dynamic signature of each failure, it is not feasible to train pilots in an exhaustive manner to handle all the non-nominal situations associated with each class of failures/damages/malfunctions. The existence of an FDIE scheme supports automatic accommodation as part of a fault-tolerant control system and it can also improve human accommodation through increased pilot situational awareness.

Most of the research efforts in the area have focused on individual classes of failure and did not address the failure evaluation aspect. State estimation or observer-based schemes have been widely proposed for actuator failure FDI relying on Kalman or other classes of filters (Wilsky, 1980; Marcos, 2002; Shin, 2002; Narendra, 1997). Artificial Neural Networks (ANN) have also been extensively used to solve the FDI problem for aerospace systems (Napolitano, 1996; Napolitano, 2000; Jakubek, 2002; Lou, 2002).
Alternative approaches for FDI and pilot awareness enhancement were also proposed based on inductive learning (Iverson, 2004).

The issue of sensor FDI has been addressed to a lower extent, since triple and quadruple physical redundancy of aircraft sensors is a common practice. However, sensor FDI schemes based on ANN estimations of sensor outputs have been proposed (Napolitano, 1995; Perhinschi, 2003).

Research regarding the dynamic impact and accommodation of structural damage to main aircraft components (wing, horizontal tail) has recently been focused on the development of fault-tolerant control laws with indirect failure assessment through parameter identification without explicit FDI (Nguyen, 2006). The use of large networks of sensors for global structural health monitoring has been also investigated (Tessler, 2007).

The failure evaluation process must address several distinct aspects such as to determine the type of the failure, its magnitude or severity, and evaluate failure effect on reducing the flight envelope, in the most general sense. These issues are important to enhance pilot situational awareness and provide necessary information to the automatic control system to avoid commands that might lead to loss of control and other dangerous/catastrophic situations.

The attempt to integrate FDIE for a large diversity of aircraft sub-systems and over extended areas of the flight envelope poses significant challenges. From a failure accommodation point of view, the differentiation between – for example - a sensor and an actuator failure is a critical task because different types of compensation are necessary in each case. If the specific signal associated with the sensor failure is used in the control laws, it might be challenging for a pilot to distinguish between sensor and actuator failures. Furthermore, very often, it is important but difficult to determine - within each of the two categories - which particular element has failed. Such issues related to the integration of FDIE for different classes of failure have only been addressed on a limited basis and comprehensive and systematic methodologies have yet to be developed (Perhinschi, 2003; Perhinschi, 2007).

The need for a solution to the FDIE problem for aerospace vehicles that includes all sub-systems over the entire flight envelope has been widely acknowledged (Totah, 2007; Srivastava, 2008; Young, 2007a; Young 2007b) and has become a major objective of
NASA’s Aviation Safety Program (White, 2006). The complexity and extremely high dimensionality of the problem require adequate tools. Recently, a new concept inspired from the biological immune system was proposed for aerospace systems FDI (KrishnaKumar, 2003; Dasgupta, 2004). The AIS-based fault detection can potentially address directly the complexity and multi-dimensionality of aircraft dynamic response in the context of abnormal conditions and provide the tools necessary for a comprehensive/integrated solution to the FDIE problem.

On the other hand, it is desirable to perform the aircraft sub-system FDIE with the necessary level of detail to allow for adequate specific reformulation of the control tactics and strategies such that the control of the vehicle is maintained or recovered and the mission is continued with the same or amended objectives and requirements. The three processes grouped under the acronym FDIE must be performed in subsequent phases to increase efficiency and reliability. Detection represents the process of declaring that a generic malfunction of the system has occurred. Any one or several of the total $N_S$ sub-systems can be subject to the failure. These sub-systems can be actuators, sensors, propulsion, structural elements, etc. The identification process has two phases or more, depending on the complexity of the sub-system. The first phase consists of determining in which of the $N_S$ categories the failure falls or, in other words, determining what is the failed sub-system. The outcome of the second phase specifies the failed element (e.g. roll rate sensor, or rudder actuator, or left wing). In certain situations, an intermediate phase could be defined to distinguish between groups within the sub-system. For example, if an actuator failure is declared, an intermediate phase would determine which of the three control channels are affected. It is important to clarify that the term identification used in this thesis refers to the classification of failures without any difference with the meaning of the term isolation as used for other authors. The evaluation of the failure addresses three aspects. One is of a qualitative nature and involves determining the type of the failure. For example, the qualitative evaluation is expected to determine if an actuator failure consists of a locked actuator, or a freely moving control surface, or a reduction of control efficiency. The other two aspects are of a quantitative nature and can be defined as direct and indirect. The direct failure evaluation consists of estimating the magnitude or severity of the failure (e.g. left aileron locked at +10deg). The indirect failure evaluation includes
the re-assessment of the flight envelope and prediction of the limitations and constraints on
the performance and handling qualities inflicted by the presence of the failure. The general
aspects of the aircraft sub-system FDIE problem are illustrated in Figure 3.1 (Perhinschi,
2009).

![Aircraft Sub-System FDIE Problem](image)

Figure 3.1 Aircraft Sub-System FDIE Problem (Perhinschi, 2009)

It is envisioned that the role of such a FDIE system will be twofold: 1) it will trigger
compensatory actions from the fault-tolerant flight control system to maintain stability and
control of the aircraft, and 2) it will provide information to the pilot and/or control system
for decision making regarding modification of control and navigation strategies.
Chapter 4 Aircraft Sub-System Failure Modeling

4.1 - Aircraft Model

The aircraft aerodynamic model is derived from a non-linear model of a high performance military aircraft distributed by NASA to academic institutions in 1990 within a student design competition (Antoniewicz, 1988). This generic model was customized through the addition of the aerodynamics modeling of canard surfaces for the purpose of simulating the NASA IFCS F-15 research aircraft (Perhinschi, 2004). The aerodynamic and thrust characteristics are provided through 42 look-up tables, that is 16 tables for the longitudinal dynamics as functions of Mach number, angle of attack and stabilator deflection; 20 tables for the lateral-directional dynamics as functions of Mach number, angle of attack, sideslip angle and rudder; 2 tables for engine thrust and fuel flow as functions of Mach number and altitude. Additional look-up tables have been added for the modeling the canards. The look-up tables have been subdivided to isolate the contribution of individual aerodynamic surfaces and control surfaces in order to be able to simulate structural damage and control surface failure.

4.2 - Failure Modeling

Four types of failures were modeled to support the development and testing of the AIS-based FDIE scheme: actuator, sensor, propulsion, and structural failures/damages. A brief description of the modeling approach is presented next.

4.2.1 - Actuator Failure Modeling

Within this effort, failure on left or right individual stabilator, aileron, or rudder - since the aircraft considered is equipped with a dual fin - have been considered. Two types of control surface failure are modeled: stuck aerodynamic control surface and physically damaged aerodynamic control surface. The first failure type corresponds to an actuator
mechanism failure and results in a locked surface; in fact, at the failure occurrence, the
control surface remains fixed in the current position/deflection or moves to a pre-defined
position and remains fixed there. A failure involving a blockage of the control surface at a
fixed deflection does not alter the aerodynamic properties of the control surface. However,
each surface in a pair (left and right) will have different deflections and the resulting
moments and forces are computed individually. The second failure type corresponds to a
physical destruction and/or deformation of the control surface. It consists of a
deterioration of the aerodynamic “efficiency” of the control surface starting at the failure
occurring moment. A control failure that involves physical destruction of the control
surface may alter the aerodynamic properties in manners that can be both qualitative
(affecting the nature of the aerodynamic phenomena involved) and quantitative (affecting
the magnitude of characteristic parameters). More details and complete models are
presented in the references (Perhinschi, 2006; Perhinschi, 2008).

4.2.2 - Sensor Failure Modeling

Failures of the gyros on the three channels have been considered within this effort.
The simulated sensor failure implemented consists of an output bias. The transition to the
biased sensor output can be instantaneous (step bias) or over a certain transient (drifting
bias). Different transients as well as different sizes of the bias can be defined. Thus, two
types of sensor failures are implemented: Large Step Bias and Large Fast Drifting Bias in
the angular rate sensors (Perhinschi, 2006).

4.2.3 - Structural Failure Modeling

For the specific aircraft considered, only the damage of the wing must be modeled
separately. The stabilators and rudders coincide with the entire respective aerodynamic
surfaces. Therefore, damages of these surfaces are modeled as described in 4.2.1.

A simple model of wing damage is developed considering both aerodynamic and
gravimetric effects. The failure type corresponds to a total or partial physical destruction
and/or deformation of the wing and different percent values along the wing can be selected
as damage affected area. In addition, the effect of this type of failure on the ailerons control
(physically damaged aerodynamic control surface) has been modeled for certain wing damage percentages.

At post failure conditions, the non-symmetry of aerodynamic forces produced by the left and right aerodynamic surfaces requires that the aircraft forces and moment computation be adjusted. An additional rolling moment is introduced due to the non-symmetric lift produced by the damaged semi-wing, and an additional yawing moment due to the non-symmetric drag produced by the damaged semi-wing.

4.2.4 - Engine Failure Modeling

Simple models for the following engine failures/malfunctions have been modeled:

- stuck throttle
- thrust runaway
- power/thrust reduced control efficiency

The “stuck throttle” failure implies normal operation of the engine but no response to power lever actuation. The throttle \( \kappa(t) \) remains constant at the value reached at the moment of failure occurrence \( t_f \).

The “thrust runaway” failure models a malfunction of the fuel control system, which causes the increase of the fuel flow to maximum and the increase of the thrust as a result. This is modeled by increasing the throttle to maximum with first order dynamics and time constant set-up by the user.

Finally, the “power/thrust reduced control efficiency” is modeled by scaling down the throttle input by a constant factor selected by the user.

4.3 - Outline of Simulated Failures

Table 4.1 outlines the characteristics of the preliminary set of simulated failures considered in this thesis. Different magnitudes for these types of failure have been selected in order to perform the failure evaluation phase of the AIS-based FDIE scheme.

The training data set is created by taking all of the data from nominal conditions, and the test data were generated from flight tests with failure. Additional validation set of data at normal conditions is acquired to evaluate the level of false alarms.
<table>
<thead>
<tr>
<th>Failure Category</th>
<th>Failure Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Actuator</td>
<td>Stabilator</td>
<td>Blockage of any, left or right, control stabilator surface at 8deg and 2deg.</td>
</tr>
<tr>
<td></td>
<td>Aileron</td>
<td>Blockage of any, left or right, control aileron surface at 8deg and 2.5deg.</td>
</tr>
<tr>
<td></td>
<td>Rudder</td>
<td>Blockage of any, left or right, control rudder surface at 8deg and 4deg.</td>
</tr>
<tr>
<td>Sensor</td>
<td>Large Step Bias LSB</td>
<td>Step bias of 10 deg/s and 5 deg/s in the roll and pitch rate gyro sensors and 3 deg/s and 1 deg/sec in the yaw rate gyro sensor</td>
</tr>
<tr>
<td></td>
<td>Large Fast Drifting Bias LFSB</td>
<td></td>
</tr>
<tr>
<td>Structural</td>
<td>Wing damage</td>
<td>Loss of 15% of any, left or right wing, affecting the “efficiency” of the aileron control surface. Loss of 6% of any, left or right wing, without affecting the “efficiency” of the aileron control surface.</td>
</tr>
<tr>
<td>Engine</td>
<td>Power/thrust reduced control efficiency</td>
<td>Loss of the 98% and 60% of the power in any left of right engine.</td>
</tr>
</tbody>
</table>
Chapter 5 Flight Simulation Testing and Data Acquisition

5.1 - WVU 6-DOF Flight Simulator

The WVU Motus 600 Flight Simulator (Figure 5.1) manufactured by Fidelity Flight Simulation, Inc., Pittsburgh, PA includes the following components (Anonymous, 2006):

- 6 DOF motion platform driven by electrical induction motors
- Laminar Research X-Plane flight simulation software
- LCD mosaic wall four-monitor external visual display
- Instructors operating station
- Computer and control cabinet

The motion platform provides adequate six-degrees-of-freedom translational and rotational motion cues. Electrical motors are used to drive the motion base (Figure 5.1), which represents a very versatile and inexpensive solution to this type of application. Motion drive algorithms convert the motion of the aircraft as resulting from the dynamic model into motion of the platform such that the perception of the pilot is optimized within the physical limitations of the ground based simulator.

5.2 - Interface of Aircraft Model with the 6 DOF Flight Simulator

The WVU Flight Simulator has been interfaced (Sagoo, 2008) with an external computer on which the WVU IFCS F-15 research aircraft model can be run within the Matlab/Simulink environment to drive the entire simulator system (see Figure 5.3). Pilot input signals are transferred from the simulator cockpit into the Matlab/Simulink model. The outputs of this model are sent to X-Plane (Meyer and Van Kampen, 2002), for the control of all the simulator sub-systems including the generation of visual cues. However, the connection of X-Plane to the motion computer is deactivated and the signals from the
external computer are sent directly to the motion computer, which drives the motion base. This set-up allows the use of any Simulink aircraft model including customized failures to drive the simulator.

Figure 5.1 The WVU 6 – DOF Flight Simulator System

Figure 5.2 The WVU 6 – DOF Flight Simulator Cabin

Figure 5.3 Interface of the WVU Flight Simulator with External Models (Perhinschi and Napolitano, 2009)
Figure 5.4 shows the top level Simulink diagram of the model interfaced with the WVU Flight Simulator. The model includes the non-linear dynamics of a supersonic fighter (as described in section 4.1) and models failure/damages of actuator, sensors, wing, and engine. The three large blocks at the bottom of Figure 5.4 include the computation of specific variables to be provided to the flight simulator to drive the generation of visual and aural cues as well as the motion of the simulator platform.

![Figure 5.4 Top Level Simulink Model Interfaced with WVU 6-DOF Flight Simulator (Sagoo, 2008)](image)

### 5.3 - Simulation Scenarios

To define the self as completely and accurately as possible, adequate coverage of the state space must be achieved. Different flight scenarios are considered to be performed over a wide area of the flight envelope, which is first defined based on nine specific reference points (see Figure 5.5) for Mach numbers between 0.6 and 0.9 and altitudes between 9000 ft and 31000ft. Flight tests start at steady state flight condition #1 and continue to cover the nine points as described by the arrows in Figure 5.5. For example, one flight test starts at #1, the aircraft is accelerated at constant altitude to point #4,
descended at constant speed to point #5, and then returned to point #4 and #1. A total of eight such tests are necessary to cover the testing flight envelope. The data from these tests are used to develop the failure detection and identification scheme. Additional tests at intermediate points (A, B, C, D in the Figure 5.5) are performed to be used as validation data.

The set of flight scenarios, lasting between 10 and 20 minutes each, are designed to include steady state flight conditions, transitions between steady state conditions, and mild to moderate maneuvers. These flight scenarios are simulated under normal flight conditions. They are repeated under various failure scenarios for both design/development and validation purposes. Only one failure at a time is considered to capture/isolate the dynamic fingerprint of each type of failure and generate antibodies appropriately. The data from the simulator were acquired at the rate of 50 Hz.

The following list of maneuvers is an example of one of the 8 nominal flight tests. They are repeated for every segment until all the nine reference points defined in the Figure 5.5 are covered. Starting at the trim condition Mach 0.75 and altitude 20000 ft, the maneuvers can be described as follows:
**At point #1:**

- Steady level flight at 20000 ft and Mach 0.75. (Maintain steady state level, symmetrical flight for 5 seconds).
- Perform left and right coordinated turn, half progressive circle at 5, 10, 15 and 20 degrees bank angle.
- Maintain steady state level symmetrical flight for 5s at Mach 0.75.
- Perform three sets of pitch attitude doublets (+15degrees and –15degrees, +10degrees and –10degrees, +5degrees and –5degrees) while maintaining altitude, velocity, and heading; and steady state level symmetrical flight for 5s at Mach 0.75 between the set of doublets.
- Perform three sets of roll attitude doublets (+15degrees and –15degrees, +10degrees and –10degrees, +5degrees and –5degrees) while maintaining altitude, velocity, and heading; and steady state level symmetrical flight for 5s at Mach 0.75 between the set of doublets.
- Perform three yaw attitude doublets at different angles while maintaining altitude, velocity, and heading.
- Maintain steady state level symmetrical flight for 5s at Mach 0.75.
- Perform left and right coordinated turn, half progressive circle at 5, 10, 15 and 20 degrees bank angle.
- Maintain steady state level symmetrical flight for 5s at Mach 0.75.
- Climb up to 22000ft at 10 degrees bank angle 5 degrees pitch angle and come back to 20000ft at the same but negative angles.
- Maintain steady state level symmetrical flight for 5s at Mach 0.75.
- Accelerate up to Mach 0.9 in 60 seconds (reach point #4).

**At point #4:**

- Steady level flight at 20000ft and Mach 0.90. (Maintain steady state level, symmetrical flight for 5 seconds).
- Perform the same maneuvers as at the point 1.
- Maintain steady state level symmetrical flight for 5s at Mach 0.90.
- Descent to 9000ft and Mach 0.9 within 60 seconds with constant velocity (reach point #5).

**At point #5:**
- Steady level flight at 9000ft and Mach 0.90. (Maintain steady state level, symmetrical flight for 5 seconds).
- Maintain steady state level symmetrical flight for 5s at Mach 0.90.
- Perform the same maneuvers as at the point 1.
- Maintain steady state level symmetrical flight for 5s at Mach 0.90.
- Climb up to 20000ft and Mach 0.9 within 60 seconds with constant velocity (return to point #4)
- 

**At point #4:**
- Steady level flight at 20000ft and Mach 0.90. (Maintain steady state level, symmetrical flight for 5 seconds).
- Perform left and right coordinated turn, half progressive circle at 5, 10, 15 and 20 degrees bank angle.
- Maintain steady state level symmetrical flight for 5s at Mach 0.90.
- Decelerate back Mach 0.75 at 20000ft within 60 seconds with constant velocity. (Point # 1)

**At point #1:**
- Steady level flight at 20000ft and Mach 0.75. (Maintain steady state level, symmetrical flight for 5 seconds).
Chapter 6 General Architecture of AIS

6.1 - AIS Paradigm

The biological immune system has the capability to detect microbial and non-microbial exogenous entities while not reacting to the self cells. T-cells (Benjamini, 1992) are the component of the system with the most important role in this process. These cells are first generated through a pseudo-random genetic rearrangement mechanism, which ensures high variability of the new cells in terms of biological features. Typically, these features are specific molecular strings of organic compounds such as proteins or polysaccharides. A selection process takes place in the thymus resulting in the destruction of the T-cells whose features match the self. Eventually, only those T-cells that are “different” are allowed to leave the thymus and proliferate. This process is referred to as negative selection. The surviving T-cells can now circulate throughout the body to detect intruders and mark them for destruction.

The mechanisms and processes of the biological immune system are the inspiration for the AIS, as a new artificial intelligence technique for fault detection (Farmer, 1986; Forrest, 1994; Dasgupta, 1999). The basic idea of this new computational paradigm is that an abnormal situation (i.e. failure of one of the aircraft sub-systems) can be declared when a current configuration of “features” does not match with any configuration from a predetermined set known to correspond to normal situations. These “features” can include any information expected to be relevant to the behavior of the system and able to capture the signature of abnormal situations. Extensive experimental data are necessary to determine the self or the hyper-space of normal conditions.

Adequate numerical representations of the self/non-self must be used and the data processed such that they are manageable given the computational and storage limitations of the available hardware. The artificial counterpart of the T-cells - the detectors - must then be generated and optimized. This process may be repeated to generate several sets of
detectors as part of a hierarchical scheme that allows failure isolation and evaluation. At this point, the obtained selves can be organized and classified based on the capability of each one to detect and identify every type of failure. Finally, a detection logic must be designed for real time operation with high detection rate and low number of false alarms. The block diagram of the AIS design process for fault detection is presented in Figure 6.1.

![Block Diagram of AIS Design Process for Fault Detection](image)

**Figure 6.1 Artificial Immune System-Based Abnormal Condition Detection**

### 6.2 - AIS for Aircraft Sub-System FDIE

Using the flowchart in Figure 6.1 as a starting point, the aircraft sub-system FDIE based on the AIS paradigm can be considered to include three main processes:

- pre-processing of information and flight data
- on-line detection, identifications, and evaluation
- post-processing of FDIE outcomes

The general flowchart of the AIS-based FDIE is presented in Figure 6.2.

The **Pre-processing of information and flight data** has as outcomes the sets of detectors for the various phases of the FDIE. This process includes key activities such as the definition of the features, data acquisition, data reduction, and detector generation and optimization.
The **On-line FDIE** process implies the development and operation of an FDIE scheme. Sets of current values of the features measured in flight at a certain sampling rate are compared against the detectors. For each sample, a binary output results, 0 if the current values of the features are outside the detector (“normal” situation) or 1 if the current values are inside the detector (“abnormal” situation). Alternatively, an output between 0 and 1 can result if fuzzy instead of crisp boundaries are considered for the detectors. To reduce the number of false alarms, sets of output values over moving time windows are typically used to produce the FDIE outcome and declare a failure.

![Diagram of FDIE Process](image)

**Figure 6.2 AIS – Based FDIE**

The **post-processing** of FDIE outcomes and the analysis of false alarms and failed detections can potentially be used on or off-line to alter the detector sets and improve the overall performance.
Chapter 7 Antibodies Generation Algorithms

A process that is of absolute importance for the AIS is the matching between the artificial antibodies or detectors and the explored data or candidates (data subject to the detection process). This is the equivalent of the biological matching between the antibodies and antigen, which is the basis for the recognition and selective elimination mechanism of foreign elements. In general, the matching rules rely on metrics for comparison and a logic to produce a binary output – match or not-match. They depend on the type of data representation. The matching rules are used in two instances. The first is the detector generation process, in which candidate detectors are compared with the self. The second is the detection process, during which the explored data are compared with the established detector to check for abnormalities.

7.1 - Binary Valued Matching Rules

One of the first matching rules based on binary representation implemented for AIS is the so-called r-contiguous matching (Forrest, 1994). The matching between detectors and candidates is declared positive if there is a window of pre-selected size $r$ in which all the bits are identical. Alternative matching rules have been proposed using binary programming (D’haeseleer, 1996) and binary measures of distance or similarity (Harmer, 2002). However, some unsatisfactory results have been reported (Gonzalez, 2003) when using binary representation in the context of negative selection, apparently caused by incompatibilities between the nature of the data used to define the self/non-self space and the representation of these data (Balthrop, 2002).

In this thesis, time histories of continuous variables are primarily used to define the self/non-self. Thus, it is expected that a string extension of the self may be necessary or useful for the identification phase and the integration of multiple failure detection schemes. However, since some tests reveal that the performance of negative selection algorithm
combined with binary representation of time history data is modest (Gonzalez, 2003), this research work is only based on the Real-Valued Matching Rules which are described next.

### 7.2 - Real Valued Matching Rules

The real valued vector representation describes the self/non-self as sets of $n$-dimensional points; therefore, in general:

- **Current test point**: $C = [c_1, c_2, \ldots, c_n]^T$, $C \in \mathbb{R}^n$  
  \hspace{1cm} (7.2.1)
- **Detector**: $D = [d_1, d_2, \ldots, d_n]^T$, $D \in \mathbb{R}^n$  
  \hspace{1cm} (7.2.2)

The matching rules are based on the “distance” $\Delta$ between $C$ and $D$. $\Delta$ can be defined as the Minkowski distance:

$$\Delta(C, D) = |C - D|_p = \left( \sum_{i=1}^{n} |c_i - d_i|^p \right)^{\frac{1}{p}}$$

\hspace{1cm} (7.2.3)

with its particular case, the Euclidean distance, for which $p=2$. The matching rule can be defined in terms of a distance threshold $\delta_d$ such that:

$$\Delta \leq \delta_d \Rightarrow \text{match}$$

\hspace{1cm} (7.2.4)

Alternatively, a detection radius $\rho_d$ is defined and the matching rule formulated as:

$$\Delta < \rho_d \Rightarrow \text{match}$$

\hspace{1cm} (7.2.5)

An equivalent matching rule can be formulated by defining the self/non-self as a set of hyper-spheres. In this case, a current test point can be represented by:

- **Current test hyper-sphere**: $C = \{O_C, R_C\}$, $O_C \in \mathbb{R}^n$, $R_C \in \mathbb{R}$  
  \hspace{1cm} (7.2.6)
- **Detector**: $D = \{O_D, R_D\}$, $O_D \in \mathbb{R}^n$, $R_D \in \mathbb{R}$  
  \hspace{1cm} (7.2.7)

where $O$ are the centers and $R$ the radii of the respective hyper-spheres. A detector is said to be activated when the following condition is satisfied:

$$\Delta(O_C, O_D) < R_C + R_D \Rightarrow \text{match}$$

\hspace{1cm} (7.2.8)

Alternative shapes for the self/non-self representation are hyper-rectangles (Dasgupta, 2003) and hyper-ellipsoids (Shapiro, 2005).
7.2.1 - Detector Generation via Negative Selection

Forrest et al., (1994) proposed the negative selection algorithm (NSA) for the generation of detectors, which is inspired by the mechanism used by the natural immune system to train the T-cells to recognize antigens (non-self) and prevent them from recognizing body’s own cells (self). The AIS attempts in this way to differentiate between what is normal and what is abnormal.

There has been, to date, no research reporting on deterministic generation mechanisms for real-valued vector representation. The current approaches rely on random initialization of candidate detectors and subsequent censoring to eliminate overlapping with the self and additional random generation for non-self coverage. Several algorithms based on negative selection have been studied, tested, and evaluated for the purpose of this research effort.

7.2.1.1 - NSA-Real Valued with Fixed Detector Position NSA-RVF

Initially, a set of detectors is generated randomly. Each individual detector then goes through a ‘training’ process whereby any detectors that are found to match any element of a set of self samples – based on the Euclidean distances - are discarded and replaced by other randomly generated detectors. This ensures that the set of detectors remains a constant size. These replacement detectors are also checked against the set of detectors previously defined. A detector that goes through this process and is not eliminated becomes a mature detector. The procedure is repeated until all of the detectors in the detector set are mature, non-self detectors. Instead of the Euclidean distance to the nearest element of the self, the median Euclidean distance to the \( k \) nearest self elements is calculated and used as a matching criterion (Gonzalez and Dasgupta, 2002). This strategy makes the algorithm more robust to outliers and noise in the data sets.

In Figure 7.1, a typical execution of the algorithm for a 2-dimensional self version data set (time series with 512 elements) is shown where 800 detectors were considered. The blue circles represent the self samples and the red ones the generated detectors. Figure 7.1a shows the set of initial candidate detectors and Figure 7.2b shows the set of matured detectors. A radius (threshold) of 0.02 for the self points and detectors was considered. Only one nearest neighbor is selected for this experiment. The results are modest because
the algorithm has reduced capability to avoid overlapping with the self and does not ensure adequate coverage of the non-self.

7.2.1.2 - NSA-Real Valued with Mobile Detectors NSA-RVM

As opposed to the basic version, where a candidate detector matching the self is discarded, in the modified version, the detector is moved away from the self-sample. At each iteration, if the detector is still within the threshold, it is stepped further away from the self-sample. Every detector is assigned an age which is increased every time it is relocated (Gonzalez and Dasgupta, 2002). Its position continues to be updated in this way until the detector either moves outside the threshold distance or reaches a pre-specified maturity level. The size of the step that a detector takes is determined by the parameter $\eta$. This parameter must be reduced at each iteration in order to ensure that the algorithm converges to a stable state (Gonzalez and Dasgupta, 2002). The following updating rule is used:

$$\eta = \eta_0 e^{-\tau i}$$  \hspace{1cm} (7.2.9)

where $\eta_0$ is the initial adaptation rate value, $\tau$ is its decay parameter and $i$ is the current iteration.
As in the case of the basic NSA-R with fixed detectors, tests using a 2-dimensional self version data set and a similar general procedure were performed. In Figure 7.2, a typical execution of this modified algorithm is presented with 800 antibodies and radius (threshold) of 0.02 for the self points and detectors. Although the coverage of the non-self is improved as compared to the basic NSA-R, the issue of the overlapping with the self space persists. Note that the size of all detectors is the same. This creates difficulties in covering small non-self spaces. Also, using a constant size, the number of detectors needed to cover a certain area will increase. It is necessary to explore new algorithms where the number of detectors and the area covered are optimized, giving more flexibility to the detector generation process.

![Initial Candidate Set of Detectors](image1)

![Final Set of Mature Detectors](image2)

**Figure 7.2 The Modified NSA-R with Mobile Detectors for a 2-Dimensional Self Version Data Set**

### 7.2.1.3 - NSA-Real Valued with Variable Detectors NSA-RVV

This algorithm allows the radius of each detector to be variable. It is expected that this will result in fewer detectors and better non-self coverage. While the previous two NSA-R are susceptible to have ‘holes’ or ‘gaps’ in the area covered by its detectors, the variable detector algorithm is likely to avoid this drawback as smaller detectors may be generated to fill them while keeping the same number of detectors. The use of variable detector radii permits a small number of detectors with large radii to cover large areas of
non-self space and makes the coverage of “holes” more feasible using very small detectors (Ji and Dasgupta, 2004).

The algorithm can be enhanced by computing the radius of each candidate detector that does not overlap with the self, at the initial stage, as the distance to the nearest self element.

A maximum number of detectors is preset to the largest acceptable number of detectors and will terminate the algorithm when reached if another stopping criterion has not already been called.

In Figure 7.3, a typical execution of the algorithm for a 2-dimensional self version is shown where 1000 detectors were considered. The blue circles represent the self samples and the red ones the generated detectors.

The coverage of the non-self is improved with respect to the results shown for the previous versions of the NSA-R. Since this approach allows small size of detectors, the small non-self space can be explored and covered. Although this method shows a high capability for anomaly detection, some additional objectives for increased efficiency need to be reached such as:

1) Optimization of the number of detectors

2) Optimization of the overlap among detectors.
These objectives can be reached using alternative methods for optimization. Within this effort, the implementation of evolutionary algorithms has been explored (Davis and Perhinschi, 2009).

In Figure 7.4, a typical execution of the same algorithm for a 3-dimensional self version and same parameters is shown for illustration.

![Figure 7.4 The Variable Detector NSA-R for a 3-Dimensional Self Version Data Set](image)

7.2.1.4 - Enhanced NSA-Real Valued with Variable Detectors ENSA-RV

Starting with an initial set of candidate detectors, located randomly in the non-self of an $n$-dimensional hyper-space, the algorithm performs a selection process based on two criteria: no overlapping with the self and maximum coverage of the non-self. At every iteration, the radius of each detector is computed using the distance between the candidate detector and the nearest self cluster. Since a minimum radius $r_m$ is permitted for detectors, the distance between centers $d(c_i,c_j)$ must be greater than or equal to the sum of $r_m$ and the radius of the cluster $r_s$. 
Because a better coverage may be achieved when a minimum overlapping among detectors is allowed, an overlapping measure $w_i$ of a detector with respect to the others is calculated during the maturation process (Wong and Dasgupta, 2004).

$$w_j = \sum_{j=1}^{m} \left( e^{\delta_{ij}} - 1 \right)^{n-1} \quad (7.2.10)$$

$$\delta_{ij} = \left( \frac{r_i + r_j - d(c_i, c_j)}{2r_i} \right) \quad (7.2.11)$$

For an overlapping threshold value $w_{thr}$, every detector is selected as mature if the condition $w_i \leq w_{thr}$ is satisfied. Eventually, if $w_i = 0$, that particular detector is selected to have a number of $N_{clon} = 2n$ clones around it. The center of the first clone is placed at a distance equal to one radius and at a random unitary direction. The remaining clone centers are generated at $90^\circ$ angles with respect to the first one at $n$ different planes.

If $0 < w_i \leq w_{thr}$, only one center clone is generated at a direction opposite to nearest element (mature detector or cluster self), according to:

$$c_{\text{clone}} = c_{\text{mature}} + (1 + \eta_{\text{clone}}) r_{\text{mature}} \frac{c_{\text{mature}} - c_{\text{nearest}}}{\left\| c_{\text{mature}} - c_{\text{nearest}} \right\|} \quad (7.2.12)$$

where $\eta_{\text{clone}}$ corresponds to a decay parameter which determines how far the clone element is located at every iteration and is defined by:

$$\eta_{\text{clone}} = \eta_{\text{clone}} e^{-\text{iter}/\tau_{\text{clone}}} \quad (7.2.13)$$

Additionally, the $N_{MOV}$ smallest rejected detectors are selected to be moved in opposite direction of the mean center of the $k$-nearest elements (equation 7.2.14). Similar to the cloning operation, the moving process is performed with the same decay criteria of the cloning operation by replacing $\eta_{\text{mov}}$ and $\tau_{\text{mov}}$ in the equation 7.2.13.

$$c_{\text{mov}} = c_{\text{rej}} + \eta_{\text{clone}} \frac{c_{\text{rej}} - c_{\text{nearest}}}{\left\| c_{\text{rej}} - c_{\text{nearest}} \right\|} \quad (7.2.14)$$

Finally, a set of $N_{RD}$ random centers is inserted; the radius of the mature detectors calculated, and the coverage and overlapping computed using the Monte Carlo method described in the section 7.3.
The process can be stopped after a prescribed number of iterations, when a prescribed maximum number of acceptable detectors has been reached, or when a desired coverage of the non-self has been achieved. The algorithm can optimize the requirements for no overlapping among non-self detectors and self and minimum un-covered areas in the non-self.

In Figure 7.5, a typical execution of the algorithm for a 2-dimensional self version is presented after the 2 iterations (Figure 7.5a) and after 50 iterations (Figure 7.5b).

7.2.2 - Detector Generation via Positive Selection

Through positive selection (PS) strategy, the detectors are generated to coincide with the self and the process is equivalent to clustering the self data. This time, an abnormal situation is declared if the explored current configuration does not match any of the detectors. However, this is not suitable compared with the NS where the activation of a single negative antibody is enough to declare the presence of abnormal situation. Using PS instead, it is necessary to test the complete set of positive antibodies before classifying a sample as abnormal.

From another point of view, since in most of the anomaly detection applications, a large amount of of positive (healthy) samples are always necessary for training, a high
number of positive detectors (clusters) must be generated in order to preserve the self signature and reduce the empty space. This implies a high computing cost when testing the positive detectors against online current data. With NS, however, a smaller set of negative detectors with variable radius can be generated to cover most of the non-self space. This improves considerably the computational efficiency for online applications.

The focus during the design of the FDIE detection scheme on the non-self and – for that matter – the initial use of NS is necessary for the identification and evaluation phases. A structure of the non-self hyperspace relative to the known and unknown failures is necessary. However, it should be noted that labeling the detectors for identification purposes is performed using PS-type of algorithms.

Different issues regarding the application of negative and positive selection have been explored in the literature. Stibor et al, (2006) showed that through PS, the complexity of algorithms to optimize the generation of negative detectors and the high dimensionality issues related to it are mitigated. Ji et al, (2006) argued however, that these issues are not only characteristics to the NS approach, but instead are presented in all learning strategies.

In general, it is important to notice that conventional classification algorithms need both positive and negative samples. Therefore, in this thesis, the two approaches have been implemented to design the FDIE scheme. With NS, different sets of antibodies are generated for detection purposes; then, PS is used to generate antibodies for identification and evaluation of anomaly samples.

7.2.3 - Two-Phase Evolutionary Algorithm for Detector Generation

Based on the analysis of available algorithms, the real valued data representation and geometric hyper-body shaped detectors have been selected for the development of the FDIE scheme. To ensure optimal efficiency and good detection performance, the detector generation process must be driven by the following optimization criteria (OC):

- OC#1 - Minimum overlapping with the self;
- OC#2 - Maximum coverage of the non-self;
- OC#3 - Minimum number of detectors;
- OC#4 - Minimum overlapping among detectors.
- OC#5 – Optimum empty space
As long as the self is completely defined as a set of clusters, zero overlapping among detectors and self can be obtained through analytical constraints on the detectors location. The other four objectives can be reached through evolutionary optimization.

Genetic or evolutionary algorithms (EA) are parameter iterative search techniques that rely on analogies to natural biological processes (Davis, 1991; Michalewicz, 1994). They simulate the evolution of species and individual selection based on Darwin’s “survival of the fittest” principle to perform parameter optimization. EAs work simultaneously on a set (population) of potential solutions (individuals) to the problem under investigation. A performance index based on the optimization criteria pertinent to the problem to be solved is defined to assess the “fitness” of each individual. The degree to which solutions meet the performance requirements and constraints is evaluated and used to select “surviving” individuals that will “reproduce” and generate a new population. Individuals will now undergo alterations similar to the natural genetic mutation and crossover and possibly other genetic operators. The next iteration starts with this new population (new set of possible solutions). The process continues until there is no more significant increase in the performance of the best solution and/or the maximum number of iterations set-up by the designer is reached.

For the purpose of generating detectors, an individual consists of a set of values of all the parameters that define the non-self. For example, aircraft states and correlation coefficients of selected states at every moment in time could form an “individual”. These data are actually clustered and an individual is set of – let’s say – hyper-spheres that cover the non-self.

The initial population is determined using a variable detector algorithm that ensures that OC#1 and OC#2 are satisfied. This is the first phase of the evolutionary algorithm. Performance with respect to these criteria is preserved throughout the generations by appropriate design of genetic operators. For the second phase of the EA, a performance index including OC #3 and #4 is considered. Before these two phases are performed, the OC#5 must be met by the minimization of the non-self covered by the generated self-clusters.

Specific genetic operators, that is customized mutation and crossover operators are designed to accommodate the real-valued data representation and the representation of the
detectors as geometric hyper-bodies. More details about the design of the two-phase EA for the generation of detectors are presented in the reference (Davis and Perhinschi, 2009).

After different NSA have been explored and analyzed, all of the developed algorithms needed for the design, optimization and testing of antibodies have been integrated within an interactive environment called IFDOT (WVU Immunity-Based Failure Detector, Optimization and Testing). This user-friendly graphical interface has been implemented using Simulink/Matlab tools. The Figures 7.6 and 7.7 show the portal to the tool and the main menu, respectively. Details about the description of this interactive utility and some examples of its functionality can be found in the reference (Davis, 2009; Davis and Perhinschi, 2009).

Figure 7.6 Interactive Design Environment of the IFDOT (Davis and Perhinschi, 2009)
7.3 - Calculation of Overlapping and Coverage Using the Monte Carlo Algorithm

The goal of this technique is to estimate the volume and overlapping of hyper-bodies without using analytical formulas that are impractical for high dimensional spaces. As a particular case, using this volume estimation, it is possible to calculate how many antibodies of a given radius are needed to cover the non-self space. The estimation is reliable with a defined standard error.

The Monte Carlo method integrates a function over a complicated domain, where analytical expressions are very difficult to be applied – e.g. the calculation of the volume of overlapping hyper-spheres in higher dimensions. Given integrals of the form

\[ I = \int_{\chi} h(x)f(x)dx, \]

where \( h(x) \) and \( f(x) \) are functions for which \( h(x)f(x) \) is integrable over the space \( \chi \), and \( f(x) \) is a non-negative valued, integrable function satisfying \( \int_{\chi} f(x)dx = 1 \), the Monte Carlo method picks \( n \) random points \( x_1, x_2, \ldots, x_n \), over \( \chi \) and approximates the integral as
The absolute error of this method is independent of the dimension of the space $\chi$ and decreases as $1/\sqrt{n}$ (Fishman, 1995). By applying this integration method, two fundamental questions arise:

- How many observations should one collect to ensure a specified statistical accuracy?

- Given $n$ observations from a Monte Carlo Experiment, how accurate is the estimated solution?

Using the Chebyshev’s inequality and specifying a confidence level $1 - \delta$ (Fishman, 1995), one can determine the smallest sample size $n$ that guarantees an integration error no larger than $\varepsilon$. This specification is called the $(\varepsilon, \delta)$ absolute error criterion and leads to the worst-case sample size.

$$N := \frac{1}{4\delta\varepsilon^2}$$

(7.3.2)

This equation shows that there is always a tradeoff between the accuracy of the solution and the sample size. The standard deviation of the error is inversely related to $\delta$. Also the error $\varepsilon$ is kept small by making $n$ large enough.

An straightforward algorithm can be developed which estimates the total space (volume) covered by the hyper-spheres or hyper-cubes inside the unitary hypercube $[0, 1]^n$; which represents the universe or the union of self and non-self as well as the overlapping among clusters or detectors. The results shown in the Figure 7.8 to 7.11 are obtained by sampling the points in 2 dimensions. The total number of samples for achieving a confidence level of 95% (0.95) required the following (Perhinschi and Moncayo, 2008):

- Standard Error $\varepsilon = 0.01$.

- The probability outside the Confidence interval, $\delta = 0.05$ (Target = 1 - \delta)

- The number of Points, ($n \gg 1/4 \delta\varepsilon^2$) or approximately 50,000 samples (or points).

It can be seen, that the accuracy of the integrated space depends on the absolute error of the estimated volume and the confidence level, a higher confidence level $\delta$ or a
smaller absolute error $\varepsilon$ bias, the required sample size and therefore the algorithm runtime complexity.

It is important to notice that the analysis of the available algorithms show that the algorithms associated with the real valued representation perform better than those associated to the binary representation for applications where the self/non-self is described by time histories of continuous variables (Gonzalez, 2003). Based on this conclusion, a two phases evolutionary algorithm has been designed designed for the generation and optimization of AIS detectors (Davis and Perhinschi, 2009). The first phase, based on the variable size detector algorithm ENSA-RV, ensures that there is no overlapping with the self and that the non-self is covered to a desired predetermined level. The second phase ensures that a minimum number of detectors are used and that the overlapping with other detectors is minimal.

Figure 7.8 Hyper-Spheres Detectors Coverage

Figure 7.9 Hyper-Spheres Overlapping
Chapter 8 Design Issues of AIS

Pre-processing of the data has the purpose to condense the data and reduce the amount of storage and computing resources while preserving the information content. As a part of this pre-processing, feature selection, clustering, normalization and data fusion algorithms to update the database without duplication when new tests are available have been developed. The reduced information is then represented as a collection of hyper-bodies, which form the self set (normal patterns) that can be used to generate a diverse set of detectors.

However, throughout the design process of the AIS-based FDI system, several critical issues have been identified, such as: data clustering and empty space optimization, data fusion and duplication removal, definition of features and dimensionality reduction, and selection of cluster/detector shape. They are closely related to the general framework (Perhinschi, 2009) of the FDIE and are primarily due to the extremely large amounts of experimental data that are necessary to determine the self and to the potentially very high-dimensional feature space. These design issues are described and analyzed in this chapter. Solutions and alternatives are proposed and their impact on detection performance is illustrated using data obtained from the WVU motion based flight simulator.

For the purpose of illustrating the issues analyzed in this chapter, only a subset of the flight envelope described in Section 5.3, and the high magnitude failures outlined in the Table 4.1, are considered.

The Enhanced Negative Selection Algorithm for Real-Valued representation with Variable Detector Radius (ENSA-RV) is used. As described in Section 7.2.1, the algorithm ensures that there is no overlapping with the self and that the non-self is covered to a desired predetermined level. The quantitative evaluation has been defined by using a specific metric. Assuming typical binary outcomes, the results of the detection can be categorized as:

- **TP** - True Positives: abnormal data points detected as abnormal
- TN - True Negatives: normal data points NOT detected as abnormal
- FP - False Positives: normal data points detected as abnormal
- FN - False Negatives: abnormal data points NOT detected as abnormal

The detection rate (DR) is defined as the ratio between abnormal data points detected as abnormal divided by the total amount of abnormal data points:

\[
DR = \frac{TP}{TP + FN} \times 100
\]  
(8.1)

The false alarm rate (FA) is defined as the ratio between normal data points detected as abnormal divided by the total amount of normal data points:

\[
FA = \frac{FP}{TN + FP} \times 100
\]  
(8.2)

8.1 - Data Clustering and Empty Space

When the clustering process is performed, some issues need to be addressed to ensure that the dynamic signature will be captured without loss of significant information. A cluster will include a (potentially) large number of data points (self) but also a certain amount of “empty space”, points between the actual data, which are assumed to be self but can actually be either self or non-self. When evaluating the amount of “empty space”, a certain confidence radius is assumed around every self point within which all points are considered self. This is another parameter that must be selected with caution. For illustration, Figure 8.1 shows the amount of empty space that is generated around two nominal points when a single hyper-sphere or hyper-cube cluster is build around them. Here, the clustering process is based on the k-means (Elkan, 2003) method to reduce the training dataset and improve the time complexity of the detector generation.
Pertinent values for the radius around a single nominal point can be obtained by analyzing the distance between points of the self measured at different sampling rates. If a small number of clusters are generated, in order to cover all data points, the size of every cluster will likely be large compared to the radius around each data point and a large amount of “empty space” – potentially non-self - is included. In Figure 8.2, the variation of the empty space with the number of clusters is presented. A number of 26831 self data points and a constant self point confidence radius of 0.0005 – normalized with respect to the unit hyper-cube – are considered. The empty space decreases when the number of clusters is increased. Note that the empty space can reach significantly large values.
The presence of empty space affects the detection rate and the number of false alarms. For example, if a small number of clusters is generated for a given set of data; many clusters are likely to cover some non-self space. Since no detectors can be generated in that particular space, the detection rate will decrease. On the other hand, if the test data do not perfectly cover the self and a big number of clusters are considered, the size of the clusters will not permit to cover some space that could be part of the self but will be covered by detectors instead. In this case, the false alarms are likely to increase and the computational effort may increase due to the large number of clusters. In Figures 8.3 and 8.4, the DR and FA performance of three different sets of clusters is presented for a four dimensional self defined with the following parameters: NN roll channel weight and NN compensation on all three channels. The training data set used in these experiments is limited to a small region of the flight envelope and the number of detectors was the same for all cases. The false alarms have been calculated from a set of nominal data different from the one used in the training process. These results confirm the fact that increasing the number of clusters, the number of activated detectors is increased (higher detection rate). As the number of clusters is increased for the same set of data, their radius is decreased, giving the opportunity for new detectors to be generated in that particular non-self space that lies between clusters. However, using a larger number of clusters apparently increases the number of FA. This means that some points of the self were not present in the training data and were only included in the self by increasing the radius of the clusters.

It is clear from this example that an adequate number of clusters, which ensures a desirable balance between DR and FA, needs to be determined prior to detector generation. An imposed amount of empty space can be achieved by varying the number of cluster using the algorithm presented in Figure 8.5. The algorithm starts with a set of initial clusters and a given confidence radius $r_s$ for the self points. At each iteration, the total empty space is calculated using a Monte Carlo method (Fishman, 1995). If the empty space is less than or equal to a desired value, the algorithm stops. Otherwise, the number of cluster is increased and the iterations continue.
Figure 8.3 Detection Rate Using Different Number of Clusters

Figure 8.4 False Alarms Using Different Number of Cluster
8.2 - Data Fusion and Duplicate Elimination

Whenever new data to characterize the self/non-self is available, the clustering process must be repeated with the entire set. It would be more efficient to be able to do the clustering only on the newly acquired data and then put old and new clusters together eliminating any duplication. Data fusion algorithms for this purpose have been developed.

The first step for data fusion process is eliminating any duplicate points in the defined self space before clustering the flight data set. Thus, two points will be considered the same if the Euclidian distance between them is less or equal to a fixed threshold value. This threshold has to be determined in such a way that the features represented in the self are preserved. Applying the duplicate elimination process, the computational time required for clustering is reduced considerably.

After the duplication is eliminated, the clustering of the new data set is performed next and then the fusion of the new and old sets. This fusion algorithm allows the updating of the database when new flight tests are available. The approach is based on the measured overlap between clusters, which is calculated in a similar way as the overlapping between the detectors. The algorithm favors clusters with bigger radii and will preserve for the final self representation those clusters with more efficient coverage.

Figure 8.6 presents an example of the fusion of two different data clustered sets. Each set of data represent a different flight test but processed with the same number of clusters.
clusters. The two-dimensional self space in this case is represented by the non-dimensional roll and yaw rate. Figures 8.6a and 8.6b show the 500 clusters of the data flight set 1 and 2 respectively. Figure 8.7a represents the integration of the two data sets before the fusion process is applied. Finally, 689 best clusters are obtained with acceptable overlap among them as shown in Figure 8.7b.

The impact of the data fusion process in the detection performance can be analyzed in terms of the DR and FA parameters before and after the fusion operation is applied. For
example, the same 5000 clusters of the four-dimensional self space defined in the Section 8.1 (NN roll channel weight and NN compensation on all three channels), are fused here with 5000 clusters of a new set of flight data. The updated self, with 9696 clusters resulted from the fusion operation, is then used to generate 500 detectors. The results of the detection of the four failure categories considered are presented in Figures 8.8 and 8.9. Note how the fusion process improves significantly the FA maintaining around the same percent of DR for all failures. Thus, when new data is fused to old ones, new self space uncovered before is now covered reducing the space where false alarms can potentially occur.

![Figure 8.8 Detection Rate Comparison – Before and After Applying Data Fusion](image)

*Figure 8.8 Detection Rate Comparison – Before and After Applying Data Fusion*
8.3 - Definition of Features

8.3.1 - Self Configurations

One of the most critical phases in the design of AIS schemes consists of defining the set of “features” whose values characterize the self – or normal conditions and – for that matter – the non-self, or the abnormal conditions. These “features” can include various sensor outputs, states estimates, statistical parameters, or any other information expected to be relevant to the behavior of the system. They can be instantaneous samples or time histories over constant or variable windows. The candidate parameters for self/non-self definition can be grouped in the following five categories:

- Aircraft state variables
- Pilot input variables
- Stability and control derivatives
- Variables generated within the control laws
- Derived variables

The aircraft state variables are a natural choice since measurements of aircraft angular rates have been used for self/non-self definition and failure detection (Dasgupta, 2004) with promising results.
It is possible that the dynamic fingerprint of a failure be reproduced through intentional pilot input. For example, it is well known that an elevator failure induces a coupling between the longitudinal and lateral channel. This characteristic may be used for detection by using as features the roll and pitch rates. However, it is possible to achieve similar coupling under normal conditions through simultaneous pilot input on both channels. The locus of non-dimensionalized roll and pitch rates is presented in Figure 8.10 for decoupled lateral and longitudinal inputs at nominal conditions. Note that the cross-like shape is due to the decoupling – input on one channel produces an angular rate response on that channel only. The stabilator failure will be responsible for non-zero roll rate in response to a longitudinal input, as shown in Figure 8.11. However, a similar pattern of the locus can be obtained if simultaneous inputs are provided on both the lateral and longitudinal channel, as shown in Figure 8.12.

![Figure 8.10 Longitudinal and Lateral Decoupled Input – Nominal Condition](image)

![Figure 8.11 Longitudinal and Lateral Decoupled Input – Stabilator Failure](image)

![Figure 8.12 Longitudinal and Lateral Coupled Input – Nominal Condition](image)
This example suggests that information about the pilot input may be needed for correct failure detection. Several signals can be used to provide such information: stick and pedals displacement, aerodynamic control surfaces deflections, and reference state variables generated by model following control laws.

The stability and control derivatives can provide useful information regarding subsystem failures. It should be noted that determining them on-line is not a trivial problem and they have not been considered in this thesis.

The artificial neural network (ANN) control augmentation implemented within the model provides useful signals with significant FDIE capabilities. The NN output, its derivative, the neural weights and their derivatives have FDIE potential. Since the adaptation activity increases after the occurrence of a failure, these signals capture the increased adaptation activity and thus detect the failure. More details on the NN architecture, inputs, outputs, and training are presented in the references (Perhinschi and Napolitano, 2004).

Finally, previous studies have put into evidence promising parameters for FDI based on correlations between state variables (angular rates) - \( \overline{R}_{rr}, \overline{R}_{pq} \) (Perhinschi, 2007) - and based on neural estimates of the angular rates - MQEE, OQEE, and DQEE \(_x\) (Napolitano, 2000). A brief description of these parameters is given next, but more details are provided in the references.

- parameters based on NN estimates of angular rates:

\[
MQEE(k) = \frac{1}{2} \left[ (p(k) - \hat{p}_{MNN}(k))^2 + (q(k) - \hat{q}_{MNN}(k))^2 + (r(k) - \hat{r}_{MNN}(k))^2 \right]
\]  

(8.3.1)

where \( p(k), q(k), \) and \( r(k) \) are measurements of angular rates at sample \( k \) and \( \hat{p}_{MNN}(k), \hat{q}_{MNN}(k), \) and \( \hat{r}_{MNN}(k) \) are neural estimates of the angular rates based on sensor measurements including the respective gyro, over a specified time window.

\[
OQEE(k) = \frac{1}{2} \left[ (\hat{p}_{DNN}(k) - \hat{p}_{MNN}(k))^2 + (\hat{q}_{DNN}(k) - \hat{q}_{MNN}(k))^2 + (\hat{r}_{DNN}(k) - \hat{r}_{MNN}(k))^2 \right]
\]

(8.3.2)

where \( \hat{p}_{DNN}(k), \hat{q}_{DNN}(k), \) and \( \hat{r}_{DNN}(k) \) are neural estimates of the angular rates based on sensor measurements that do NOT include the respective gyro, over a specified time window.
\[ DQEE_x(k) = \frac{1}{2} (\hat{x}_{DNN}(k) - x(k))^2, \quad x = p, q, r \]  

(8.3.3)

- angular rates correlation parameters, \( \bar{R}_{rr} \) and \( \bar{R}_{pq} \):

\[
\bar{R}_{rr}(k) = \mu_{rr} \cdot OQEE(k) + \sum_{i=k-n}^{k} R_{rr}(i) \]  

(8.3.4)

\[
\bar{R}_{pq}(k) = \sum_{i=k-n}^{k} R_{pq}(i) \]  

(8.3.5)

where \( \mu_{rr} \) is a scaling factor and \( n \) defines the width of the time window over which the angular rate correlation coefficients are summed.

To study the impact of selecting different features on the performance of the detection, eight different self configurations were used considering several dimensional space sizes. After generating the corresponding detectors for every self (around 500 hypersphere antibodies), the performance of the abnormal detection was analyzed for every type of failure (high magnitude of failure according to Table 4.1). The Table 8-1 summarizes the definition of every self space with their respective features.

<table>
<thead>
<tr>
<th>Self Number</th>
<th>Definition</th>
<th>Dimension Space</th>
</tr>
</thead>
<tbody>
<tr>
<td>Self#1</td>
<td>NN roll channel weight, NN compensation on all three channels, derived parameters (MQEE, OQEE, DQEE(_p), DQEE(_q), DQEE(_r)), Rpq, Rrr, and tracking errors on all three channels</td>
<td>14</td>
</tr>
<tr>
<td>Self#2</td>
<td>NN compensation on all three channels, derived parameters (MQEE, OQEE, and DQEE(_p), DQEE(_q), DQEE(_r)), Rpq, Rrr, and tracking errors on all three channels</td>
<td>13</td>
</tr>
<tr>
<td>Self#3</td>
<td>NN roll channel weight, NN compensation on all three channels, derived parameters (MQEE, OQEE, DQEE(_p), DQEE(_q), DQEE(_r))</td>
<td>9</td>
</tr>
<tr>
<td>Self#4</td>
<td>NN compensation on all three channels, derived parameters (MQEE, OQEE, DQEE(_p), DQEE(_q), DQEE(_r))</td>
<td>8</td>
</tr>
<tr>
<td>Self#5</td>
<td>NN roll channel weight, NN compensation on all three channels</td>
<td>4</td>
</tr>
<tr>
<td>Self#6</td>
<td>derived parameters MQEE and OQEE, Rpq, Rrr,</td>
<td>4</td>
</tr>
<tr>
<td>Self#7</td>
<td>NN compensation on all three channels</td>
<td>3</td>
</tr>
<tr>
<td>Self#8</td>
<td>tracking errors on all three channels</td>
<td>3</td>
</tr>
</tbody>
</table>

In Figures 8.13, 8.14, and 8.15 results of the experiments for five different failures are shown. Only Self#5 achieves an acceptable performance with high detection rate for all failure cases, but with a high number of false alarms. However, note that every self case presents at least one acceptable performance in one of the failures considered. This result
can be used to develop an integrated scheme where different self configurations will ensure high detection rate and low number of false alarms for all considered anomalies (Moncayo, 2009b). For example, Self#1 would become a good candidate because presents high DR for the stabilator and structural failures maintaining at the same time a very low number of FA. Another candidate would be the Self#8 because the correct detection of the sensor failure with a low percent of FA.

![Figure 8.13 Detection Rate for Different Self Configurations – Actuator Failure (High Magnitude)](image1)

**Figure 8.13 Detection Rate for Different Self Configurations – Actuator Failure (High Magnitude)**

![Figure 8.14 Detection Rate for Different Self Configurations – Sensor, Structural, and Engine Failures (High Magnitude)](image2)

**Figure 8.14 Detection Rate for Different Self Configurations – Sensor, Structural, and Engine Failures (High Magnitude)**
Except for the stabilator and structural failures, one of the worst detection was recorded for Self#1. This test confirms that adding parameters or considering more possible states of flight operation not always will improve the final results in terms of detection and false alarms rates. Similar results are obtained for the Self#2 which represents a relative high dimensional space. Differentiating between numerous classes of failures requires a large number of features thus increasing the dimensionality of the detector space and exposing the entire process to specific issues that can potentially have a negative impact on the performance of the FDIE scheme (Charu, 2000; Verleysen, 2005).

When increasing the dimensionality of the self/non-self space, an exponentially larger number of clusters/detectors are necessary to maintain the same resolution. This requirement was not met in the tests presented in Figures 8.13 and 8.14 as the same number of clusters was used for all cases, which resulted in larger clusters for the 14-D set (Self#1) and in more empty space/non-self included in them leading to the lower performance of the highest dimensionality feature set. Dimensionality reduction techniques (Van Der Maaten, 2007) may be used to avoid effects of high dimension spaces while preserving the quality of information.
In general, a larger number of detectors is expected to achieve a better coverage of the non-self, hence better detection performance (as shown in Figure 8.16 for Self#5) at the expense of computational effort. However, it should be noted that a larger number of detectors is typically accompanied by a reduction in the size of (some) detectors. The implication is that smaller size detectors are more likely to cover small areas in the vicinity and between the self clusters, areas that very often are actually self for which data is not available. As a consequence, the false alarm rate may increase along the number of detectors, as shown in Figure 8.17.

![Figure 8.16 Detection Rate Using Different Number of Detectors](image1)

*Figure 8.16 Detection Rate Using Different Number of Detectors*

![Figure 8.17 False Alarms Using Different Number of Detectors](image2)

*Figure 8.17 False Alarms Using Different Number of Detectors*
8.3.2 - Shape Configurations

Data representation has an important impact on algorithm effectiveness and performance. It determines the possible matching rules, the detector generation mechanisms, and the detection process. In general, the data to be processed may include numeric data, categorical data, Boolean data, and textual data. With the real-valued vector representation, a condensed data version using a reduced number of parameters can be achieved by replacing clusters of data by circumscribed geometrical hyper-bodies. However, the characteristics (shape) of these bodies have impact on the efficiency of the detector generation process and on the detection itself. They determine how well the non-self is covered, how many detectors are necessary, and how intensive the computational process is. The following shapes for the self/non-self are analyzed to assess their advantages and disadvantages:

- Hyper-rectangles – determined by an n-dimensional center and one value for the size;
- Hyper-spheres – determined by an n-dimensional center and one value for the radius.

**Hyper-Spheres**

The hyper-sphere detectors are defined as the pair \((c, r_d)\), where \(c = [c_1, c_2, \ldots, c_n]\) is the center and \(r_d\) is the radius. The representation assumes an n-dimensional “feature” space defining the self. In general, the volume of the hyper-sphere is computed as:

\[
V = \frac{(\pi/2)^k (2R)^n}{n!!}
\]  

(8.3.1)

This calculation is difficult to perform especially for edge hyper spheres and is inefficient in overlap volume estimation. A statistical estimate of the volume is used instead.

In order to test the location of a point inside a hyper-sphere, the Minkowski distance measure is used for the distance \(D\) between two points \(x\) and \(y\):

\[
D(x, y) = (\sum |x_i - y_i|^k)^{1/k}
\]  

(8.3.2)
The Minkowski distance with $\lambda = 2$ is equivalent to Euclidean Distance. Let $r_t$ be a threshold distance around any data point $p$. When $D \leq r_d + r_t$, the test point $p$ is assumed to fall inside the detector $d$ and the detector is said to be activated or triggered.

**Hyper-Rectangles**

The hyper-cube detectors are defined as the pair $d = (c, s_d)$, where $c = [c_1, c_2, \ldots, c_n]$ is the center and $s_d$ is the side. For hyper-cubes, $s_d$ is a scalar since it has the same value for all dimensions. In general, for hyper-rectangles, $s_d$ is an n-dimensional vector, since the side on each dimension may be different. In general, the volume of the hyper-rectangle is given by the expression:

$$\text{volume}(R) = \prod_{i=1}^{n} (\text{high}_i - \text{low}_i)$$  

(8.3.3)

The geometrical volume estimation involves the multiplication of the difference of lower and higher planes in each dimension. In order to test the location of the point $(x_j)$ within a closed bounded subspace represented as hyper-rectangle, L1- norm is used which in simplified set notation indicates the limits within which each of the dimensions of the point exists for the corresponding range of the subspace.

$$x^i_j \in [\text{low}^i, \text{high}^i] \quad \forall i \in n$$  

(8.3.4)

Figures 8.18 and 8.19 present a typical set of hyper-sphere and hyper-rectangle detectors for a particular self definition, respectively. Note that no overlapping is permitted in the hyper-rectangles case, but it is allowed for the hyper-spheres. In general, it is not possible to cover a space with spherical detectors without allowing some overlapping. However, when the calculation of the covered space volume of antibodies is necessary for optimization purposes, the computing cost will become a critical aspect. For example, for hyper-spheres, the Monte-Carlo Randomized method is used to calculate the total volume covered by antibodies. For hyper-rectangles, since they are not overlapping, simple algebraic calculations are enough.
The Self#6 and Self#7 described in Table 8.1, were used to determine the impact of selecting the hyper-shape on the effectiveness and performance of detection. 500 antibodies were generated using hyper-spheres and hyper-rectangles for the two self configurations. The Figures 8.18, 8.19 and 8.20 illustrate results of the experiments considering different types of failures. In general, the percent of detection rate remains around the same for the two hyper-shape cases. However, a lower number of false alarms are achieved using hyper-rectangles antibodies. This is due to the fact that because of the allowed overlapping, the spherical detectors achieve better coverage of the non-self including areas that are attributed to the non-self simply because of imperfections of the self. Thus, for the same number of hyper-rectangles, the hyper-spheres produce an increase of detection rate, but an increase of the number of false alarms as well. In the literature, a combination of different hyper-shapes in an integrated scheme has been proposed (Balachandran, 2005). As compared to existing single shaped detection strategy, the approach utilizes the geometric properties of each shape to improve the detection performance in terms of coverage, number of detectors generated and overlap constraints by adding geometric flexibility and diversity.
Figure 8.20 Detection Rate for Different Hyper-Shape Antibodies for Self#6

Figure 8.21 Detection Rate for Different Hyper-Shape Antibodies for Self#7
The analysis and examples provided in this chapter show that a careful selection of the self space configuration must be made in order to properly balance numerous contradictory effects and increase the capabilities of the AIS for failure detection. It is important to notice that every self case studied in this chapter presents at least one acceptable performance for one of the failures considered. This result is used to develop an integrated scheme where different self configurations will ensure high detection rate and low number of false alarms for all considered anomalies (Moncayo, 2009). The scheme is based on a Multi-Self Strategy and is presented and described in the following chapter.
Chapter 9 Hierarchical Multi-Self Strategy

This chapter describes the development of an AIS-based scheme which is capable of detecting, identifying and evaluating the four categories of failures considered. The proposed approach is based on a Hierarchical Multi-Self (HMS) strategy where different self configurations are selected and integrated to achieve low number of false alarms and high detection rates for the sub-system abnormal conditions. In other words, the failure detection, identification and evaluation performance achieved by a complete set of features of dimension $n$ may be matched by a collection of sets, each of dimension $n_i$, where $n_i << n$, properly integrated, such that the issues with the multi-dimensionality of the hyper-space are mitigated.

As shown in Figure 9.1, the on-line detection, identification, and evaluation process is performed using three main components. The first one uses an integrated block of self patterns which performs the detection phase. The second component, where the identification phase is performed, attempts to ensure the correct identification of category and sub-categories of failures at different levels. In the third one, an evaluation based on the magnitude estimation of the failure is performed. In this phase, an indirect failure evaluation includes also the re-assessment of the flight envelope and prediction of the limitations and constraints on the performance inflicted by the presence of the failure.
9.1 - Detection

Detection is considered to be the process leading to declaring that an abnormal condition is affecting any of the sub-systems. During this phase, sets of current values of the features measured in flight at a certain sampling rate are compared against the detectors that have been generated for every self configuration as shown in Figure 9.2. A detection parameter $\zeta$ is calculated, which represents the number of consecutive points over a window $\omega$ that trigger detectors, summed over all selves. If $\zeta$ is within a certain range, a failure warning is issued, but if $\zeta$ exceeds the upper bound of the range, a failure is declared and the identification phase starts.
9.2 - Identification

Within the non-self detectors, sub-sets can be identified to correspond to specific categories of abnormal conditions for identification purposes. The approach implies the use of the negative selection strategy and a priori knowledge of specialized detectors. In Figure 9.3, this concept is illustrated for the 2-dimensional case. However, positive selection can also be applied to determine identifiers instead of detectors. This new identifier term implies that it is not necessary to use for identification the same antibodies used in the detection phase (called detectors). This is because the detectors are designed with a variable size in such a way that they cover the most of the non-self space. Thus, the bigger the detectors are, the smaller the detection time. However, bigger detectors are not convenient during the identification phase, since they can be activated for two or more types of failure at the same time, so they need to be redefined using a finer resolution. The process of selecting the identifiers is performed applying positive selection method to the failure flight tests. In this way, the candidate identifiers are labeled according to the type of failure and will be selected as identifiers when the activation is guaranteed only for that particular failure. With this approach, the number of false identifications is reduced considerably.
The identification phase of the HMS based on identifiers is performed in two steps:

- **Pre-identification**: The failure is attributed to one of the four categories: control surface, sensor, structure, or engine failure. As shown in the Figure 9.4, the category is determined based on the number of times each set of identifiers is activated for a particular failure category. For instance, if one set of specialized identifiers is activated more often, the failure category corresponding to those specific identifiers is identified. The preliminary result is compared with the output of the other individual/isolated identifier sets. The most repetitive result will be considered the identified category.

- **Identification**: If the failure is classified as actuator failure, the failure is identified to be a left or right stabilator, aileron, or rudder failure. If pre-identified as a sensor failure, it is identified to be a roll, pitch, or yaw rate sensor failure. If pre-identified as a structural failure, the abnormal condition may be identified as affecting the left or right wing. Finally, if a propulsion failure was declared, it must now be determined if it affects the left or the right engine.
Figure 9.4 Block Diagram of the Identification Phase of the Proposed HMS Strategy

9.3 - Evaluation

The qualitative evaluation of the failure can be performed by considering the type of the failure as an additional classification target and treating it similarly to an additional identification phase.

The direct evaluation (DE) or failure magnitude assessment can be approached at three levels of accuracy. An approximate magnitude assessment, using larger categories such as “small”, “medium”, and “large”, can be performed by considering an additional classification target within each identified abnormal condition. This process is performed based on the distance of every single activated identifier with respect to the self. Figure 10.5 illustrates a typical failure signature in a 2-dimensional space. The hyper-spheres represent the identifiers generated for that particular failure. Notice that failures of high magnitude correspond to points located at larger distances with respect to the self, while failures of low magnitude correspond to points located closer to the self. In consequence, before a high magnitude identifier is activated, low magnitude identifiers have already been activated, resulting in a certain reduction of the evaluation rate for high magnitude of
failures. Higher accuracy quantitative magnitude assessment with numerical outcomes is possible based on a combination of AIS algorithms and analytical estimates, provided that extensive experimental data are available.

The indirect evaluation (ID) or reduced flight envelope prediction must rely on a combined strategy based on analytical flight envelope reduction assessment and AIS-based approaches for parameter space reduction assessment. In this thesis, a general meaning is attributed to the flight envelope, which includes all dynamic parameters and states that are of interest for performing specific piloting tasks and accomplish specific missions. However, only a few such parameters will be analyzed here. The analytical methods require accurate modeling of the failures and significant on-line computational capabilities. The AIS methods imply that all pertinent parameters to the flight envelope – considering its generalized meaning – are part of the feature sets (Perhinschi, 2009). Let us assume that the self is defined as the set of all n-dimensional hyper-spheres $S_i$ (or Flight Envelope Estimators FEE) characterized by the center $c_i$ and the radius $r_i$:

$$S = \{(c_i, r_i)\} = \{S_i\}, \quad i = 1, 2, \ldots, N_S$$ (9.3.1)

$$c_i = [x_1, \ x_2, \ \ldots, \ x_n]^T_i$$ (9.3.2)
where \( N_s \) is the total number of clusters and \( n \) is the total number of features that define the self. The self can be viewed as a generalized flight envelope. Assume that each failure \( F_k \) is defined by a set of constraints \( C \) on known variables \( x_{Fk} \), with \( k = 1,2,\ldots,N_F \):

\[
C = C(x_{Fk})
\]

(9.3.3)

The variables \( x_{Fk} \) must be part of the feature set:

\[
\{x_{Fk}\}_{k=1,2,\ldots,N_F} \subseteq \{x_{j}\}_{j=1,2,\ldots,n}
\]

(9.3.4)

Then a “new” self can be defined as:

\[
S_{new} = \{(c_i, r_i) \mid (c_i, r_i) \text{ satisfy } F\}
\]

(9.3.5)

The proposed approach for indirect evaluation requires that the direct evaluation is successful and that the effects of the failure can be related to constraints on at least one variable in the feature set. Then the effects on other variables in the feature set can be evaluated. The concept is illustrated in Figure 9.4 for the 2-dimensional case. The novelty in this approach is the use of AIS for online estimation of the achievable flight envelope of the aircraft after occurrence of faults by using an integrated solution and multi-dimensional capabilities. The output information obtained from this phase will be useful to determine the best compensation tasks within the control laws to avoid dangerous flight conditions due to the failures.

In order to reduce the computational requirements and improve efficiency, the system is trained offline with the generation of a bank of flight envelope estimators (FEE) for the known failures at different possible magnitudes. Then, the FEE are used for a flight envelope estimation in real-time by simple interpolation from the offline-generated library.
Finally, notice that completely and accurately defining the self is extremely challenging. It is expected that some areas of the self will not be covered due to lack of test data and that some non-self regions will be included in the self during the detector generation process. It is reasonable to assume that data to define the self and specialized detectors or identifiers and evaluators are likely to be collected over a certain time and that newly acquired information may be useful to improve the FDIE performance. It is useful to develop tools that are capable of tuning the sets of self or non-self detectors based on FDIE results and evaluation of results obtained during on-line operation and/or off-line analysis. This should be regarded as a continuous optimization process. Primarily, tuning of the detector set can be performed through alteration of selected detector size, adding/removing detectors, and/or splitting detectors into several smaller ones, and/or labeling existing detectors as specialized detectors.
Chapter 10 Performance Assessment of the Integrated AIS-Based Scheme

10.1 - HMS Simulink/Matlab Implementation

The purpose of this section is to provide a brief description of the implementation of the developed system using a Matlab/Simulink environment.

As shown in the Figure 10.1, the HMS scheme has been integrated as a new module in the WVU IFCS F-15 research aircraft Simulink model. Thus, the capabilities of the system to detect, identify, and evaluate the failures considered can be studied by an online process.

Figure 10.1 Integrated Artificial Immune System Scheme for the WVU IFCS F-15 Simulink Model

Figure 10.2 shows the top level diagram of the HMS block. At this level, features measured in flight are received from the sensors and provided to the AIS to be monitored by the antibodies. As illustrated in the Figure 10.3, the main block includes the detection, identification and evaluation modules in which the FDIE phases are performed. The output
of every HMS sub-block consists also of a display message system to warn the pilot about the failure.

The detailed HMS detection and identification blocks are shown in the Figures 10.4 and 10.5 respectively. Using Matlab S-functions of level-1, these two blocks contain specific selves where different set of antibodies and identifiers are loaded from a general library. The on-line FDIE process starts when the current input values over moving time windows at a sampling rate of 50Hz are compared against the antibodies. Binary outputs are used to define when an antibody has been activated. The last block in the HMS module corresponds to the evaluation phase which consists of a single S-function where direct and indirect phases are performed.
Finally, Figure 10.6 shows a typical failure warning window which is displayed once the detection of a failure takes place. In addition, information about the type of failure and its magnitude is also displayed. This window warns the pilot about a failure and provides the information needed to augment the response in handling the failure and could help the pilot to perform the best judgment in compensation tasks.
10.2 - HMS Strategy Performance Assessment

As a first step, it is important to define a set of candidate selves and analyze their performance in detecting abnormal conditions for every type of failure. Preliminary results were obtained considering only three points of the flight envelope (points 1-2-3 as defined in the Figure 5.5) and failures of high magnitudes. Then, the best selves were selected to be part of the HMS scheme and their capabilities were tested considering the whole nine points of the flight envelope.

The six sets of features outlined in Table 10.1 were used in the process as candidate sets of hyper-spherical clusters. Corresponding detectors were generated in each case. Using a point to point method, the percentage of test points for which the antibodies are activated is computed as a metric for performance assessment. The detection performance for every self and failure determined in this way is presented in Table 10.2 in terms of detection rate and false alarms. These results show that only the Self#1 achieves an acceptable detection performance for most of the failure cases, but with a high number of false alarms. In contrast, the worst case is represented by Self#5, where only the yaw gyro sensor failure is correctly detected. Note that every self case presents at least one acceptable performance in one of the failures considered. For instance, the Self#3 shows an acceptable detection for the actuator/stabilator and pitch and yaw gyro sensor failures with very low false alarms, but a poor detection for the other failures.
Table 10.1 Feature Configurations for Self Definition

<table>
<thead>
<tr>
<th>Self Number</th>
<th>Features</th>
<th>Solution Space Dimension</th>
</tr>
</thead>
<tbody>
<tr>
<td>Self#1</td>
<td>$\text{NN}_{\text{out}_x}$</td>
<td>3</td>
</tr>
<tr>
<td>Self#2</td>
<td>$\text{NN}_{\text{out}_x}$, and $\text{DQEE}_x$</td>
<td>6</td>
</tr>
<tr>
<td>Self#3</td>
<td>$\text{NN}_{\text{out}<em>x}$ and $\text{x}</em>{\text{TE}}$</td>
<td>6</td>
</tr>
<tr>
<td>Self#4</td>
<td>$\text{NN}_{\text{out}_x}$, $\text{MQEE}$, $\text{OQEE}$, and $\text{DQEE}_x$</td>
<td>8</td>
</tr>
<tr>
<td>Self#5</td>
<td>$\text{DQEE}_x$</td>
<td>3</td>
</tr>
<tr>
<td>Self#6</td>
<td>$\text{MQEE}$, $\text{OQEE}$, $\text{R}<em>{pq}$ and $\text{R}</em>{rr}$</td>
<td>4</td>
</tr>
</tbody>
</table>

Table 10.2 Detection Performance of Different Self Configurations (3 Points of the Flight Envelope)

<table>
<thead>
<tr>
<th>Self Configuration</th>
<th># detectors</th>
<th>Actuator Failure 8deg</th>
<th>Sensor Failure</th>
<th>Structural Failure</th>
<th>Engine Failure</th>
<th>New data</th>
<th>Nominal Test Data</th>
<th>False Alarms (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Detection Rates (%)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Self#1</td>
<td>516</td>
<td>R: 99.97</td>
<td>R: 99.97</td>
<td>R: 98.51</td>
<td>R: 96.39</td>
<td>L: 99.96</td>
<td>R: 34.19</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>L: 99.97</td>
<td>R: 99.75</td>
<td>L: 98.38</td>
<td>R: 98.57</td>
<td>L: 97.40</td>
<td>R: 73.75</td>
<td>0.06</td>
</tr>
<tr>
<td>Self#2</td>
<td>504</td>
<td>L: 99.98</td>
<td>R: 99.75</td>
<td>L: 98.38</td>
<td>R: 98.57</td>
<td>L: 79.56</td>
<td>R: 78.10</td>
<td>0.56</td>
</tr>
<tr>
<td></td>
<td></td>
<td>R: 98.71</td>
<td>R: 68.71</td>
<td>R: 73.72</td>
<td>R: 69.76</td>
<td>L: 44.13</td>
<td>R: 2.86</td>
<td></td>
</tr>
<tr>
<td>Self#3</td>
<td>507</td>
<td>L: 97.97</td>
<td>R: 98.14</td>
<td>L: 70.63</td>
<td>R: 73.04</td>
<td>L: 79.56</td>
<td>R: 78.10</td>
<td>2.5</td>
</tr>
<tr>
<td></td>
<td></td>
<td>R: 99.97</td>
<td>R: 99.97</td>
<td>R: 98.38</td>
<td>R: 98.57</td>
<td>L: 99.5</td>
<td>R: 10.8</td>
<td></td>
</tr>
<tr>
<td>Self#4</td>
<td>504</td>
<td>L: 99.9</td>
<td>R: 99.9</td>
<td>L: 92.3</td>
<td>R: 76.9</td>
<td>L: 99.5</td>
<td>R: 99.9</td>
<td>0.84</td>
</tr>
<tr>
<td></td>
<td></td>
<td>R: 79.8</td>
<td>R: 57.8</td>
<td>R: 92.3</td>
<td>R: 76.9</td>
<td>L: 12.44</td>
<td>R: 8.25</td>
<td>0.78</td>
</tr>
<tr>
<td>Self#5</td>
<td>507</td>
<td>L: 99.9</td>
<td>R: 99.9</td>
<td>L: 0.91</td>
<td>R: 1.60</td>
<td>L: 12.44</td>
<td>R: 8.25</td>
<td>2.5</td>
</tr>
<tr>
<td></td>
<td></td>
<td>R: 6.44</td>
<td>R: 0.23</td>
<td>R: 0.91</td>
<td>R: 1.60</td>
<td>L: 0.84</td>
<td>R: 1.02</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>R: 3.39</td>
<td>R: 9.56</td>
<td>L: 64.28</td>
<td>R: 7.55</td>
<td>R: 98.31</td>
<td>R: 6.03</td>
<td></td>
</tr>
</tbody>
</table>

Based on the previous results, the first four selves presented in the Table 10.1 were selected to be part of the HMS scheme. This selection also considered the need to limit redundancy and the overall number of parameters. As a consequence, self #6, for example, was not used within the HMS scheme. This self included angular rate correlation parameters which are known to capture well the dynamic signature of angular rate sensor failure and actuator failures which produce strong coupling (Napolitano, 2000; Perhinschi, 2007). Although the overall performance of the HMS scheme is excellent as shown later, the alternative consideration of self #6 is expected to have positive effects specifically in the identification of sensor failures.

The detection performance for every self and failure has been determined for the
nine test points of the flight envelope and is presented in Table 10.3. The results represent the averages over different test points around the flight envelope. For all cases, the set of detectors correspond to the maximum number with which no more significant improvement in the detection rate could be achieved. Once again, note that every self presents at least one acceptable performance in one of the failures considered. For instance, the Self#1 shows a poor detection for the sensor, aileron and engine failures, but an acceptable detection of other failures with very low false alarms. The fact that different selves favor the detection of particular types of failures is used to develop an integrated scheme where different self configurations ensure overall high detection rate and low number of false alarms.

Table 10.3 Detection Performance of Different Self Configurations (Extended Flight Envelope)

<table>
<thead>
<tr>
<th>Self Config.</th>
<th># detectors</th>
<th>Failure Test Data</th>
<th>Nominal Test Data</th>
<th>New data</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Detection Rates (%)</td>
<td>False Alarms (%)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Actuator Failure</td>
<td>Sensor Failure</td>
<td>Structural Failure</td>
<td>Engine Failure</td>
</tr>
<tr>
<td></td>
<td>Stabilator</td>
<td>Aileron</td>
<td>Rudder</td>
<td>LSB</td>
</tr>
<tr>
<td>Self#2</td>
<td>1331</td>
<td>L: 99.98 R: 99.98</td>
<td>L: 99.96 R: 100</td>
<td>L: 96.14 R: 88.41</td>
</tr>
<tr>
<td>Self#3</td>
<td>325</td>
<td>L: 98.62 R: 100</td>
<td>L: 99.98 R: 99.97</td>
<td>L: 62.59 R: 43.85</td>
</tr>
<tr>
<td>Self#4</td>
<td>918</td>
<td>L: 92.97 R: 99.30</td>
<td>L: 99.98 R: 100</td>
<td>L: 74.90 R: 53.00</td>
</tr>
</tbody>
</table>

10.2.1 - Detection Performance

After several simulation experiments, and varying the window ω and the threshold parameters, the detection performance of the HMS was analyzed. The results are summarized in the Tables 10.4 for different magnitudes of failures and different points of the flight envelope. The detection outcome is a binary output produced at the sampling rate based on a moving time window of width ω = 10 seconds for each self and a detection
threshold of 25%, which represents the number of consecutive points over the window that trigger detectors, summed over all selves.

As compared to the results presented in Table 10.3 for individual/isolated detector sets, the HMS approach improves significantly the detection performance for all type of failures. For the stabilator, aileron, yaw gyro and structural failures, for example, the percent of detection rate achieves even 100%. Not only the detection rate is improved; the false alarms are also reduced.

Note that the engine failure is the only one that presents a lower detection rate. In fact, none of the four selves outlined in the Table 10.3 achieves a very good detection performance for this type of failure. This is due to the fact that additional features possibly relevant to engine operation such as longitudinal acceleration and information on pilot throttle command have not been considered as part of the self configurations.

Table 10.4 Detection Performance of the HMS Strategy for the Extended F-15 Flight Envelope

<table>
<thead>
<tr>
<th>Failure Test Data Detection Rates</th>
<th>Nominal Test Data False Alarms</th>
</tr>
</thead>
<tbody>
<tr>
<td>(Convention: x-x-x = Magnitude - Flight test points - Subsystem fail)</td>
<td></td>
</tr>
<tr>
<td>Actuator Failure</td>
<td>Sensor Failure</td>
</tr>
<tr>
<td>Stabilator</td>
<td>Aileron</td>
</tr>
<tr>
<td>8-123-R: 100</td>
<td>8-189-R: 100</td>
</tr>
<tr>
<td>1-123-R: 100</td>
<td>1-165-R: 100</td>
</tr>
</tbody>
</table>

As shown in the results, the good detection performance of the HMS is recorded over a wide area of the flight envelope. Note that validation data includes flight tests to the points A, B, C and D as shown in the Figure 5.5. The low number of false alarms presented implies that the nine points selected to define the testing area of the flight envelope are enough and no additional test at intermediate points are necessary to train the AIS. This result implies a certain “robustness” of the general approach of collecting data for self definition. It suggests that a higher resolution in the flight regimes may not be necessary. In other words, the sets of features selected are invariant for self definition purposes over wide areas of the flight envelope and an overall less than expected amount of test data may
be necessary for adequate definition of the self and good FDIE performance using the AIS approach.

In general, these results confirm the fact that better detection performance can be achieved by using an integrated self scheme instead of considering self configurations separately.

10.2.2 - Identification Performance

Once a failure condition is declared by the detection phase scheme, the identification phase starts to perform a pre-classification according to the four categories of sub-systems considered. As mentioned before, the identification phase is based on identifiers instead of detectors. This implies certain flexibility on choosing the best self configuration which can be the same or different than the one used during the detection phase. Thus, after several simulations, the Self#2, Self#3 and Self#4 were selected to be part of the identification scheme since they provided the best identification results. The Tables 10.5 and 10.6 summarize the results for the identification of different type of failures. Note that unknown abnormal conditions have been considered as well. This unknown condition outcome is presented when some antibodies are activated but they do not belong to any of the identifiers sets.

As shown in the results, very good pre-identification and identification rates are achieved using the HMS strategy.

Table 10.5 Pre-identification Performance of the HMS Strategy

<table>
<thead>
<tr>
<th>Failure Type</th>
<th>Failure Category (Identification Rates (%))</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Stablator</td>
</tr>
<tr>
<td>2deg-L: Stablator</td>
<td><strong>98.16</strong></td>
</tr>
<tr>
<td>8deg-L: Stablator</td>
<td><strong>96.58</strong></td>
</tr>
<tr>
<td>2deg-R: Stablator</td>
<td><strong>96.41</strong></td>
</tr>
<tr>
<td>8deg-R: Stablator</td>
<td><strong>97.69</strong></td>
</tr>
<tr>
<td>2.5deg-L: Aileron</td>
<td>0.32</td>
</tr>
<tr>
<td>8deg-L: Aileron</td>
<td>0.83</td>
</tr>
<tr>
<td>2.5deg-R: Aileron</td>
<td>0</td>
</tr>
<tr>
<td>8deg-R: Aileron</td>
<td>0</td>
</tr>
<tr>
<td>4deg-L: Rudder</td>
<td>0</td>
</tr>
<tr>
<td>8deg-L: Rudder</td>
<td>0</td>
</tr>
<tr>
<td>4deg-R: Rudder</td>
<td>0</td>
</tr>
<tr>
<td>8deg-R: Rudder</td>
<td>0</td>
</tr>
<tr>
<td>5p: LSB</td>
<td>0</td>
</tr>
</tbody>
</table>
Table 10.6 Identification Performance of the HMS Strategy

a. Actuator Subcategory Failures

<table>
<thead>
<tr>
<th>Failure Type</th>
<th>L: Stabilator</th>
<th>R: Stabilator</th>
<th>L: Aileron</th>
<th>R: Aileron</th>
<th>L: Rudder</th>
<th>R: Rudder</th>
</tr>
</thead>
<tbody>
<tr>
<td>2deg-L: Stabilator</td>
<td>100</td>
<td>0</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
</tr>
<tr>
<td>8deg-L: Stabilator</td>
<td>100</td>
<td>0</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
</tr>
<tr>
<td>2deg-R: Stabilator</td>
<td>0</td>
<td>100</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
</tr>
<tr>
<td>8deg-R: Stabilator</td>
<td>0</td>
<td>100</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
</tr>
<tr>
<td>2.5deg-L: Aileron</td>
<td>--</td>
<td>--</td>
<td>100</td>
<td>0</td>
<td>--</td>
<td>--</td>
</tr>
<tr>
<td>8deg-L: Aileron</td>
<td>--</td>
<td>--</td>
<td>100</td>
<td>0</td>
<td>--</td>
<td>--</td>
</tr>
<tr>
<td>2.5deg-R: Aileron</td>
<td>--</td>
<td>--</td>
<td>0</td>
<td>100</td>
<td>--</td>
<td>--</td>
</tr>
<tr>
<td>8deg-R: Aileron</td>
<td>--</td>
<td>--</td>
<td>0</td>
<td>100</td>
<td>--</td>
<td>--</td>
</tr>
</tbody>
</table>

b. Sensor Subcategory Failures

<table>
<thead>
<tr>
<th>Failure Type</th>
<th>p: LSB</th>
<th>q: LSB</th>
<th>r: LSB</th>
</tr>
</thead>
<tbody>
<tr>
<td>5deg/sec-p: LSB</td>
<td>100</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>10deg/sec-p: LSB</td>
<td>100</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>5deg/sec-q: LSB</td>
<td>2.03</td>
<td>97.97</td>
<td>0</td>
</tr>
<tr>
<td>10deg/sec-q: LSB</td>
<td>1.6</td>
<td>98.40</td>
<td>0</td>
</tr>
<tr>
<td>1deg/sec -r: LSB</td>
<td>0</td>
<td>0</td>
<td>100</td>
</tr>
<tr>
<td>3deg/sec -r: LSB</td>
<td>0</td>
<td>0</td>
<td>100</td>
</tr>
</tbody>
</table>

c. Structural Subcategory Failures

<table>
<thead>
<tr>
<th>Failure Type</th>
<th>L: Structural</th>
<th>R: Structural</th>
</tr>
</thead>
<tbody>
<tr>
<td>6%-L: Structural</td>
<td>100</td>
<td>0</td>
</tr>
<tr>
<td>15%-L: Structural</td>
<td>100</td>
<td>0</td>
</tr>
<tr>
<td>6%-R: Structural</td>
<td>0</td>
<td>100</td>
</tr>
<tr>
<td>15%-R: Structural</td>
<td>0</td>
<td>100</td>
</tr>
</tbody>
</table>

d. Engine Subcategory Failures

<table>
<thead>
<tr>
<th>Failure Type</th>
<th>L: Engine</th>
<th>R: Engine</th>
</tr>
</thead>
<tbody>
<tr>
<td>60%-L: Engine</td>
<td>100</td>
<td>0</td>
</tr>
<tr>
<td>100%-L: Engine</td>
<td>98.75</td>
<td>1.25</td>
</tr>
<tr>
<td>60%-R: Engine</td>
<td>0</td>
<td>100</td>
</tr>
<tr>
<td>100%-R: Engine</td>
<td>0</td>
<td>100</td>
</tr>
</tbody>
</table>

It is important to notice that the rudder failure was considered only during the pre-
identification process. This is because the effect of this type of failure on the dynamic response of the aircraft and in particular on the set of features selected is independent of the side (left or right) failed. Therefore, the accommodation tasks related to this failure could be considered the same for any of the two sides.

10.2.3 - Direct Evaluation Performance

To perform the direct evaluation phase using the HMS scheme, several failure magnitudes have been considered. Using the concept of identifiers, an additional classification based on wide categories such as “small”, “medium”, and “large”, was performed. The same self configuration used for the identification phase has been used for the direct evaluation phase. Table 10.7 presents the results of the direct evaluation performance of the HMS for the four categories of failures considered.

<table>
<thead>
<tr>
<th>Failure Type</th>
<th>Failure Category (Evaluation Rates (%))</th>
<th>Low 2±1.5deg</th>
<th>Medium 5±1.5deg</th>
<th>High 8±1.5deg</th>
</tr>
</thead>
<tbody>
<tr>
<td>2deg-L: Stabilator</td>
<td>99.15</td>
<td>0.84</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>8deg-L: Stabilator</td>
<td>2.00</td>
<td>16.85</td>
<td>81.14</td>
<td></td>
</tr>
<tr>
<td>2deg-R: Stabilator</td>
<td>95.10</td>
<td>4.89</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>8deg-R: Stabilator</td>
<td>1.20</td>
<td>11.54</td>
<td>87.24</td>
<td></td>
</tr>
<tr>
<td>2.5deg-L: Aileron</td>
<td>100</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>8deg-L: Aileron</td>
<td>4.08</td>
<td>11.27</td>
<td>84.63</td>
<td></td>
</tr>
<tr>
<td>2.5deg-R: Aileron</td>
<td>98.57</td>
<td>1.43</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>8deg-R: Aileron</td>
<td>4.36</td>
<td>9.51</td>
<td>86.11</td>
<td></td>
</tr>
<tr>
<td>4deg-L: Rudder</td>
<td>2.38</td>
<td>97.61</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>8deg-L: Rudder</td>
<td>0</td>
<td>0</td>
<td>100</td>
<td></td>
</tr>
<tr>
<td>4deg-R: Rudder</td>
<td>0</td>
<td>100</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>8deg-R: Rudder</td>
<td>0</td>
<td>0</td>
<td>100</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Failure Type</th>
<th>Failure Category (Evaluation Rates (%))</th>
<th>Low 2±1deg/sec</th>
<th>Medium 5±2deg/sec</th>
<th>High 9±2deg/sec</th>
</tr>
</thead>
<tbody>
<tr>
<td>5deg/sec-p: LSB</td>
<td>0</td>
<td>100</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>10deg/sec-p: LSB</td>
<td>0</td>
<td>0.35</td>
<td>99.64</td>
<td></td>
</tr>
<tr>
<td>5deg/sec-q: LSB</td>
<td>0</td>
<td>100</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>10deg/sec-q: LSB</td>
<td>0</td>
<td>0</td>
<td>100</td>
<td></td>
</tr>
<tr>
<td>1deg/sec-r: LSB</td>
<td>100</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>3deg/sec-r: LSB</td>
<td>0.29</td>
<td>0.38</td>
<td>99.31</td>
<td></td>
</tr>
</tbody>
</table>
### c. Structural Failure

<table>
<thead>
<tr>
<th>Failure Type</th>
<th>Failure Category (Evaluation Rates (%))</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Low 6±2.5%</td>
</tr>
<tr>
<td>6%-L: Structural</td>
<td>100</td>
</tr>
<tr>
<td>15%-L: Structural</td>
<td>0.05</td>
</tr>
<tr>
<td>6%-R: Structural</td>
<td>100</td>
</tr>
<tr>
<td>15%-R: Structural</td>
<td>0.74</td>
</tr>
</tbody>
</table>

### d. Engine Failure

<table>
<thead>
<tr>
<th>Failure Type</th>
<th>Failure Category (Evaluation Rates (%))</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Low &lt;=40%</td>
</tr>
<tr>
<td>60%-L: Engine</td>
<td>0</td>
</tr>
<tr>
<td>100%-L: Engine</td>
<td>0</td>
</tr>
<tr>
<td>60%-R: Engine</td>
<td>0</td>
</tr>
<tr>
<td>100%-R: Engine</td>
<td>0</td>
</tr>
</tbody>
</table>

The results show that the HMS strategy achieves a high percentage of evaluation rate for different magnitudes of failure in all categories. As explained in the Section 9.3 for direct evaluation, the apparent better performance obtained for low magnitude cases is due to the natural migration of the data points away from the self, passing first through regions of the hyper-space that correspond to low magnitude failures. The rate at which this migration takes place could be a good indicator for evaluation and this hypothesis should be investigated in the future.

**10.2.4 - Indirect Evaluation Performance**

According to the criteria described in Section 9.3, the indirect evaluation process has been implemented inside the HMS scheme. Once a failure is detected, identified and its magnitude correctly determined, this phase is intended to estimate the achievable operational limits of the aircraft based on all or some dynamic parameters and states that are important to accomplish specific pilot tasks or specific missions.

For every category of failure, a set of relevant parameters defining the performance limits of the aircraft have been selected. Then, hyper-sphere evaluators are generated and their capabilities to predict the reduction of the flight envelope analyzed.
10.2.4.1 - Surface Control Failure

For surface control failures, the angular rates in all three channels have been considered to be part of the feature sets that define the self. Corresponding surface control deflections under abnormal conditions will produce constraints and alteration/reduction of the possible range of angular rates. Thus, the operational ranges of the aircraft in terms of angular rates can be determined based on the achievable limits of these variables in the presence of control surface abnormal operation.

Because a certain delay exists between the angular rates and the corresponding control deflections, preprocessing in the data needs to be performed in order to reduce this effect and obtain a more directly relationship between this two variables.

Using healthy flight test data from the nine points of the flight envelope described in the Section 5.3, the angular rates and corresponding deflection variables were plotted and the limits in nominal flight condition obtained. Figures 10.7 through 10.9 show a 15% normalized version plot of every pair of variables to consider future and additional flight test data. The operational ranges have been calculated and are summarized in the Table 10.8.

![Figure 10.7 Defined Self for Envelope Reduction Due to Aileron Failure](image)
Figure 10.8 Defined Self for Envelope Reduction Due to Stabilator Failure

Figure 10.9 Defined Self for Envelope Reduction Due to Rudder Failure
### Table 10.8 Nominal Operational Ranges

<table>
<thead>
<tr>
<th>Self Variable</th>
<th>Minimum Nominal Limit</th>
<th>Maximum Nominal Limit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Roll rate</td>
<td>-1.19 rad/sec</td>
<td>1.19 rad/sec</td>
</tr>
<tr>
<td>Differential aileron deflection</td>
<td>-11 deg</td>
<td>11 deg</td>
</tr>
<tr>
<td>Pitch rate</td>
<td>-0.25 rad/sec</td>
<td>0.25 rad/sec</td>
</tr>
<tr>
<td>Differential stabilator deflection</td>
<td>-8 deg</td>
<td>8 deg</td>
</tr>
<tr>
<td>Yaw rate</td>
<td>-0.08 rad/sec</td>
<td>0.08 rad/sec</td>
</tr>
<tr>
<td>Differential rudder deflection</td>
<td>-24 deg</td>
<td>24 deg</td>
</tr>
</tbody>
</table>

It is important to notice that the y-axis in the Figures 10.7 to 10.9 represents the total differential value between left and right deflections. For example, assuming the left aileron surface control gets stuck at 8deg, the constraint values for the differential deflection can be determined as follows:

\[
\delta_{a_{\text{max}}} = \frac{\delta_{a_{/\text{Left}}} - \delta_{a_{/\text{Right}}}}{2} = \frac{8\text{deg}+11\text{deg}}{2} = 9.5\text{deg} \tag{10.2.1}
\]

\[
\delta_{a_{\text{min}}} = \frac{\delta_{a_{/\text{Left}}} - \delta_{a_{/\text{Right}}}}{2} = \frac{8\text{deg}-11\text{deg}}{2} = -1.5\text{deg} \tag{10.2.2}
\]

\[\delta_{a_{/\text{Left}}} \text{ and } \delta_{a_{/\text{Right}}} \]

represent the maximum deflection for left and right ailerons respectively, and \[\delta_{a_{/\text{Left}_{\text{min}}} \text{ and } \delta_{a_{/\text{Right}_{\text{min}}}} \]

to the minimum ones. The normalized values correspond to: \[\delta_{a_{\text{max}}} = 0.93 \text{ and } \delta_{a_{\text{min}}} = 0.43\]

Considering a right stabilator control surface control locked at 5deg, the constraint values for the differential deflection are determined as:

\[
\delta_{e_{\text{max}}} = \frac{\delta_{e_{/\text{Left}}} + \delta_{e_{/\text{Right}}}}{2} = \frac{8\text{deg}+5\text{deg}}{2} = 6.5\text{deg} \tag{10.2.3}
\]

\[
\delta_{e_{\text{min}}} = \frac{\delta_{e_{/\text{Left}}} + \delta_{e_{/\text{Right}}}}{2} = \frac{-8\text{deg}+5\text{deg}}{2} = -1.5\text{deg} \tag{10.2.4}
\]

In the case of left rudder surface control stuck at 4deg, the constraint values are calculated as:

\[
\delta_{r_{\text{max}}} = \frac{\delta_{r_{/\text{Left}}} + \delta_{r_{/\text{Right}}}}{2} = \frac{4\text{deg}+24\text{deg}}{2} = 14\text{deg} \tag{10.2.5}
\]

\[
\delta_{r_{\text{min}}} = \frac{\delta_{r_{/\text{Left}}} + \delta_{r_{/\text{Right}}}}{2} = \frac{4\text{deg}-24\text{deg}}{2} = -10\text{deg} \tag{10.2.6}
\]
After nominal operational limits have been determined, 86 hyper-sphere evaluators were generated for every self. The Figures 10.10 to 10.12 show the distribution of such evaluators filling the defined self regions. As illustration, these Figures also present the evaluators that are activated when the considered examples for control surface failure are present. The set of parameters $p_{f_{\text{MAX}}}$, $p_{f_{\text{MIN}}}$, $q_{f_{\text{MAX}}}$, $q_{f_{\text{MIN}}}$, $r_{f_{\text{MAX}}}$ and $r_{f_{\text{MIN}}}$ represent the new maximum and minimum values of the angular rates and define the achievable limit range under failure condition. The Tables 10.9 to 10.11 summarize more results for different magnitudes of control surface failure.

*Figure 10.10 Evaluator Distribution and Activation when an Aileron Failure is Declared (Left Aileron Surface Control Locked at 8deg)*
Figure 10.11  Evaluator Distribution and Activation when a Stabilator Failure is Declared
(Right Stabilator Surface Control Locked at 5deg)

Figure 10.12  Evaluator Distribution and Activation when an Rudder Failure is declared
(Left Rudder Surface Control Locked at 4deg)
Table 10.9 Roll Rate Failure Operational Ranges

<table>
<thead>
<tr>
<th>Failure type</th>
<th>Minimum roll rate (rad/sec)</th>
<th>Maximum roll rate (rad/sec)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>( p_{f_{\text{MIN}}} )</td>
<td>( p_{f_{\text{MAX}}} )</td>
</tr>
<tr>
<td>Left Aileron Locked at 8deg</td>
<td>-0.71</td>
<td>1.13</td>
</tr>
<tr>
<td>Right Aileron Locked at 5deg</td>
<td>-1.05</td>
<td>0.79</td>
</tr>
<tr>
<td>Left Aileron Locked at 2.5deg</td>
<td>-0.79</td>
<td>0.96</td>
</tr>
<tr>
<td>Right Aileron Locked at 0deg</td>
<td>-0.88</td>
<td>0.88</td>
</tr>
</tbody>
</table>

Table 10.10 Pitch Rate Failure Operational Ranges

<table>
<thead>
<tr>
<th>Failure type</th>
<th>Minimum pitch rate (rad/sec)</th>
<th>Maximum pitch rate (rad/sec)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>( q_{f_{\text{MIN}}} )</td>
<td>( q_{f_{\text{MAX}}} )</td>
</tr>
<tr>
<td>Left Stabilator Locked at 8deg</td>
<td>-0.22</td>
<td>0.12</td>
</tr>
<tr>
<td>Right Stabilator Locked at 5deg</td>
<td>-0.19</td>
<td>0.15</td>
</tr>
<tr>
<td>Left Stabilator Locked at 2.5deg</td>
<td>-0.18</td>
<td>0.15</td>
</tr>
<tr>
<td>Right Stabilator Locked at 0deg</td>
<td>-0.16</td>
<td>0.17</td>
</tr>
</tbody>
</table>

Table 10.11 Yaw Rate Failure Operational Ranges

<table>
<thead>
<tr>
<th>Failure type</th>
<th>Minimum yaw rate (rad/sec)</th>
<th>Maximum yaw rate (rad/sec)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>( r_{f_{\text{MIN}}} )</td>
<td>( r_{f_{\text{MAX}}} )</td>
</tr>
<tr>
<td>Right Rudder Locked at 16 deg</td>
<td>-0.07</td>
<td>0.04</td>
</tr>
<tr>
<td>Right Rudder Locked at 8 deg</td>
<td>-0.06</td>
<td>0.04</td>
</tr>
<tr>
<td>Right Rudder Locked at 4 deg</td>
<td>-0.06</td>
<td>0.05</td>
</tr>
<tr>
<td>Left Rudder Locked at 0 deg</td>
<td>-0.05</td>
<td>0.06</td>
</tr>
</tbody>
</table>

As mentioned before, the results of the Tables 10.9 to 10.11 are part of a complete bank of estimators for known failures and can be used for flight envelope estimation in real time.

Finally, to illustrate the performance of the approach, the Figures 10.13 and 10.14 present the signature of the dynamic behavior of the aircraft in terms of angular rates for surface control failure condition. It is clear that the maximum and minimum values for roll, pitch, and yaw rates are inside of the predicted range values determined by the evaluators. This implies that the capabilities of the aircraft performing certain types of maneuvers will be limited by the estimated operational ranges in the presence of failures.
Figure 10.13 Dynamic Behavior for Left Aileron Failure (Locked at 8deg)

Figure 10.14 Dynamic Behavior for Left Stabilator Failure (Locked at 8deg)
10.2.4.2 - Sensor Failure

In contrast to the surface control failures, for the sensor failure case, the angular rates in all three sensor channels have been considered as a set of constraints that produce maneuvering limits. Since the sensor channels are used within the control laws of the aircraft for automatic compensation, a faulty sensor output will induce unnecessary – even damaging - compensation tasks and may determine the deflections of the aerodynamic control surfaces to get closer to the saturation limit, thus limiting the stick/pedal authority of the pilot. In consequence, the stick-command inputs must represent the set of features that define the self, in such a way that when one of the three sensors fails, the estimation of the new maneuverable ranges can be performed. However, the process can also be inverted in order to estimate the achievable limits on the angular rates due to a limitation on the stick-command inputs.

The limits in nominal flight condition for the angular rates and the stick inputs are summarized in the Table 10.12. They correspond to the maximum and minimum values of these variables that were achieved during flight test in healthy condition performed for the flight envelope described in the Section 5.3.
Table 10.12 Nominal Operational Ranges

<table>
<thead>
<tr>
<th>Self Variable</th>
<th>Minimum Nominal Limit</th>
<th>Maximum Nominal Limit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Roll rate</td>
<td>-1.19 rad/sec</td>
<td>1.19 rad/sec</td>
</tr>
<tr>
<td>Lateral Stick Input (\delta_{lat})</td>
<td>-0.0670</td>
<td>0.0670</td>
</tr>
<tr>
<td>Pitch rate</td>
<td>-0.25 rad/sec</td>
<td>0.25 rad/sec</td>
</tr>
<tr>
<td>Longitudinal Stick Input (\delta_{lg})</td>
<td>-0.0824</td>
<td>0.0824</td>
</tr>
<tr>
<td>Yaw rate</td>
<td>-0.08 rad/sec</td>
<td>0.08 rad/sec</td>
</tr>
<tr>
<td>Directional Stick Input (\delta_{dir})</td>
<td>-0.0657</td>
<td>0.0657</td>
</tr>
</tbody>
</table>

The Figures 10.16 through 10.18 illustrate a normalized version of the three selves. These figures show the distribution of different sets of evaluators that have been generated for every self. Figure 10.16 presents the activation of an evaluator when a large step bias of magnitude 10deg/sec has been identified in the roll rate sensor. Similar examples for other magnitudes of failure are presented in Figure 10.17 and 10.18 for pitch and yaw sensors, respectively. Note that the sensor failure induces a shift in the ‘cero’ position of the stick input at each direction (‘cero’ position refers to the position of the stick command at which the steady state level of the aircraft can be maintained). For example, as shown in the Figure 10.17, a 5deg/sec step bias in the pitch gyro sensor reduces by around 25% the capability of the pilot to produce a positive pitch maneuver. In Tables 10.13 through 10.15 more results for different magnitudes of sensor failure have been summarized.

Figure 10.16 Evaluator Distribution and Activation when a Sensor Failure is declared (Large Step Bias in the Roll Rate Sensor)
Figure 10.17 Evaluator Distribution and Activation when a Sensor Failure is declared (Medium Step Bias in the Pitch Rate Sensor)

Figure 10.18 Evaluator Distribution and Activation when a Sensor Failure is declared (Small Step Bias in the Yaw Rate Sensor Failure)

Table 10.13 Roll Rate Sensor Failure Operational Ranges

<table>
<thead>
<tr>
<th>Failure type</th>
<th>Shift Value $\delta_{\text{lat}}$</th>
<th>Reduction Percentage of capability to generate a positive roll rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>Large Step Bias (10 deg/sec)</td>
<td>$0.0094 \pm 0.0087$</td>
<td>14%</td>
</tr>
<tr>
<td>Medium Step Bias (5 deg/sec)</td>
<td>$0.0067 \pm 0.0087$</td>
<td>10%</td>
</tr>
<tr>
<td>Small Step Bias (2 deg/sec)</td>
<td>$0.0040 \pm 0.0087$</td>
<td>6%</td>
</tr>
</tbody>
</table>
Table 10.14 Pitch Rate Sensor Failure Operational Ranges

<table>
<thead>
<tr>
<th>Failure type</th>
<th>Shift Value δ_{lg}</th>
<th>Reduction Percentage of capability to generate a positive pitch rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>Large Step Bias (10 deg/sec)</td>
<td>0.060 ± 0.0043</td>
<td>69.4%</td>
</tr>
<tr>
<td>Medium Step Bias (5 deg/sec)</td>
<td>0.022 ± 0.0043</td>
<td>25%</td>
</tr>
<tr>
<td>Small Step Bias (2 deg/sec)</td>
<td>0.010 ± 0.0086</td>
<td>12%</td>
</tr>
</tbody>
</table>

Table 10.15 Yaw Rate Sensor Failure Operational Ranges

<table>
<thead>
<tr>
<th>Failure type</th>
<th>Shift Value δ_{dir}</th>
<th>Reduction Percentage of capability to generate a positive yaw rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>Large Step Bias (3 deg/sec)</td>
<td>0.052 ± 0.0033</td>
<td>80%</td>
</tr>
<tr>
<td>Medium Step Bias (2 deg/sec)</td>
<td>0.046 ± 0.0033</td>
<td>70%</td>
</tr>
<tr>
<td>Small Step Bias (1 deg/sec)</td>
<td>0.020 ± 0.0066</td>
<td>30.4%</td>
</tr>
</tbody>
</table>

Once again, these results represent a preliminary database that have been generated offline and the can be used for online estimation to reduce the computing time. However, the prediction of flight envelope reduction when new magnitudes of failure are detected can be performed by interpolation from the offline-generated evaluators.

10.2.4.3 - Structural Failure

To determine the effects of wing damage on aircraft performance, the aerodynamic coefficients must be considered to be part of the self. The magnitudes of the failure, expressed as the percentage of the wing tip removed, represent the set of constraints that affect the performance by reducing the lift capability. Thus, the determination of the achievable operational ranges of the damaged asymmetric aircraft is based on its current aerodynamic and structural configuration.

Note that only the aerodynamic effects associate with loss of the lift have been considered in this section. Changes in the pitching and yawing moment coefficients, center of gravity shift, and aircraft mass properties may be included following the same methodology.

A simple approach based on the aerodynamic model for different wing damage cases has been implemented. This model is used to build an offline library, and then by simple interpolation, the estimation of the lift margins is performed online. As illustrated in Figure 10.19, finite element structural models for the damage aircraft or wind tunnel tests
can be also developed to account for the damage effects and provide useful and more accurate information to the flight envelope database.

Since the current research is focused only on single prediction of loss of lift capability, the evaluators must be generated in such a way that they represent a defined relationship between percent of the loss wing and lift coefficient, and hence, their activation depends on the magnitude of the damage identified by the AIS-based FDIE. Figure 10.20 shows the reduction of lift due to wing loss as used for the structural damage model. The lift coefficient can be reduced by as much as 18% for loss of 15% of the wingtip. Figure 10.21 illustrates the distribution of the evaluators generated for structural failure and the activation of one of them when a 15% of loss of the wing is present and has been identified previously by the AIS scheme. More activation results are summarized in Table 10.16.

Figure 10.19  Indirect Evaluation Scheme for Structural Failure
Figure 10.20 Defined Self for Envelope Reduction Due to Structural Failure

Figure 10.21 Evaluator Distribution and Activation when a Structural Failure is Declared
Table 10.16 Operational Ranges Under Structural Damage

<table>
<thead>
<tr>
<th>Failure Type</th>
<th>Lift Reduction Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>5% of Loss of the Right Wing</td>
<td>6%</td>
</tr>
<tr>
<td>15% of Loss of the Left Wing</td>
<td>18%</td>
</tr>
<tr>
<td>25% of Loss of the Right Wing</td>
<td>30%</td>
</tr>
<tr>
<td>30% of Loss of the Left Wing</td>
<td>38%</td>
</tr>
</tbody>
</table>

Finally, it is important to notice that for the F15 wing geometry and configuration, a wing loss beyond 6% span will result in the loss of one of the ailerons. As a consequence, the ailerons are deflected asymmetrically, producing some pitch-roll coupling and a change in the lift coefficient. All of these effects have been considered in the structural damage model for both left and right wings. Thus, the evaluation approach to estimate the reduction on the lift capabilities described above, can be applied for both wings without any difference.

10.2.4.4 - Engine Failure

In analyzing the effects of the propulsion system abnormal operation, the conventional flight envelope of the F-15 ACTIVE vehicle defined in terms of airspeed and altitude (Timothy, 1998) has been considered to be part of the feature sets that define the self. The differential power setting, calculated as the average of the power setting between the two engines, represents a set of constraints that reduce the operational ranges of the aircraft based on the achievable limits of Mach number and altitude.

As illustrated in Figure 10.22, the flight envelope areas for several differential power settings have been estimated. Although they represent a flight envelope database created offline, they can be used for online estimation via interpolation using hyper-sphere evaluators. The Figure 10.23 represents the defined 3-dimensional self to be used for envelope reduction prediction in the presence of propulsion system failures.
Using positive selection, 268 hyper-sphere evaluators were generated. Figure 10.24 shows the distribution of such evaluators around the defined self regions. In this case, the set of constrains, given by the maximum and minimum values of the differential power, needs to be represented as hyper-planes intersecting the surface of the self. Then the
activated evaluators will correspond to achievable limit ranges in terms of Mach and altitude for such failure condition.

Assuming, for example, that the right engine throttle is stuck such that the engine is producing 60% of its total power, the constrained values for the total differential power can be calculated as follows:

\[
\delta_{T_{\text{max}}} = \frac{\delta_{T/\text{Left}}_{\text{max}} + \delta_{T/\text{Right}}_{\text{max}}}{2} = \frac{91\% + 60\%}{2} = 75.5\% \quad (10.2.6)
\]

\[
\delta_{T_{\text{min}}} = \frac{\delta_{T/\text{Left}}_{\text{min}} + \delta_{T/\text{Right}}_{\text{min}}}{2} = \frac{0.14\% + 60\%}{2} = 30.07\% \quad (10.2.7)
\]

\(\delta_{T/\text{Left max}}\), \(\delta_{T/\text{Right max}}\), \(\delta_{T/\text{Left min}}\) and \(\delta_{T/\text{Right min}}\) represent the maximum and minimum power settings for left and right engines. The nominal range values for these parameters were found to be between 0.14% and 91%.

A better visualization of the reduced flight envelope can be obtained from Figure 10.25. As combination of altitude and Mach number, the dark region corresponds to the set of operational limits or achievable ranges within which the aircraft can be flown safely. In Tables 10.17 through 10.19, more results for different magnitudes of failure have been summarized. It is important to notice that the best way in which the information regarding the flight envelope reduction can be provided to the pilot is to display the limits on the Mach number at the current altitude, in addition to the total range of achievable altitudes in the presence of a particular propulsion failure.
Figure 10.24 Distribution of Evaluators for Estimation of Flight Envelope Reduction Due to Propulsion System Failure

Figure 10.25 Propulsion Failure Impact on Flight Envelope Reduction Engine Stuck at 60% of Its Total Power
Table 10.17 Flight Envelope Reduction Due to Propulsion Failures at 90000ft

<table>
<thead>
<tr>
<th>Failure Type</th>
<th>Altitude Range (ft)</th>
<th>Mach Number Limits At current Altitude</th>
</tr>
</thead>
<tbody>
<tr>
<td>Loss of 98% of the left power</td>
<td>0 to 35160</td>
<td>0.18 to 0.55</td>
</tr>
<tr>
<td>Right engine power stuck at 40%</td>
<td>0 to 45729</td>
<td>0.40 to 0.79</td>
</tr>
<tr>
<td>Left engine power stuck at 60%</td>
<td>0 to 47209</td>
<td>0.47 to 0.90</td>
</tr>
</tbody>
</table>

Table 10.18 Flight Envelope Reduction due to Propulsion Failures at 20000ft

<table>
<thead>
<tr>
<th>Failure Type</th>
<th>Altitude Range (ft)</th>
<th>Mach Number Limits At current Altitude</th>
</tr>
</thead>
<tbody>
<tr>
<td>Loss of 98% of the left power</td>
<td>0 to 35160</td>
<td>0.23 to 0.73</td>
</tr>
<tr>
<td>Right engine power stuck at 40%</td>
<td>0 to 45729</td>
<td>0.60 to 0.99</td>
</tr>
<tr>
<td>Left engine power stuck at 60%</td>
<td>0 to 47209</td>
<td>0.62 to 1.17</td>
</tr>
</tbody>
</table>

Table 10.19 Flight Envelope Reduction due to Propulsion Failures at 31000ft

<table>
<thead>
<tr>
<th>Failure Type</th>
<th>Altitude Range (ft)</th>
<th>Mach Number Limits At current Altitude</th>
</tr>
</thead>
<tbody>
<tr>
<td>Loss of 98% of the left power</td>
<td>0 to 35160</td>
<td>0.33 to 1.04</td>
</tr>
<tr>
<td>Right engine power stuck at 40%</td>
<td>0 to 45729</td>
<td>0.30 to 1.43</td>
</tr>
<tr>
<td>Left engine power stuck at 60%</td>
<td>0 to 47209</td>
<td>0.29 to 1.79</td>
</tr>
</tbody>
</table>

In this chapter, the detection, identification and evaluation capabilities have been demonstrated in terms of low false alarm and high detection rates for different categories and magnitudes of failures. The results confirm the fact that using an integrated multiple self approach instead of considering self configurations separately can improve significantly the detection performance with reduced computational requirements.

Finally, it is important to notice that the performance results of the HMS have been obtained using a hyper-sphere self/non-self representation; however, the approach can be expanded to other hyper-shapes as hyper-cubes, hyper-ellipse or even the combination of them. Thus, the HMS can produce a flexible scheme and extract the best characteristics of different feature definitions (see Section 8.3) and integrate them to improve detection performance characteristics.

10.3 - Robustness Evaluation of the Integrated AIS

This section is intended to provide a general and preliminary analysis of the degree to which the integrated AIS operates correctly in the presence of environmental perturbations. Different test cases are selected in such a way that the robustness of the system, under atmospheric turbulence, can be assessed. However, other conditions, such as
aircraft icing, weak vortex encounters, and adverse headwinds could be considered and addressed in a similar manner.

The WVU IFCS F-15 simulation environment, in addition to the non-linear dynamic aircraft and the failure models, also incorporates an atmospheric disturbance model based on the NASA Dryden wind gust and turbulence model. Inside the scheme, the standard deviations of the wind velocity components, which are a measure of the turbulence intensity, can be modified. In the present study, only turbulence without wind effects has been addressed and the following are the three different levels considered for this purpose:

- Light turbulence: $\sigma = 1.5$ m/sec
- Moderate turbulence: $\sigma = 3$ m/sec
- Severe turbulence: $\sigma = 4.5$ m/sec

Several flight tests, lasting between 1 and 3 minutes each, were performed around Mach number 0.75 and altitude of 20000ft. They included moderate maneuvers such as doublets, coordinated turns and progressive bank angles. The atmospheric turbulence is injected at the beginning of every flight test. Table 10.20 summarizes the nine cases studied. Here, nominal condition refers to normal operation of the aircraft without considering external disturbances. Aileron subsystem failure stuck at 2.5 deg has been also considered to be part of the analysis.

<table>
<thead>
<tr>
<th></th>
<th>Nominal</th>
<th>Maneuvers</th>
<th>Turbulence Level</th>
<th>Left Aileron Stuck at 2.5deg</th>
</tr>
</thead>
<tbody>
<tr>
<td>Case 1</td>
<td>X</td>
<td>--</td>
<td>Light</td>
<td>--</td>
</tr>
<tr>
<td>Case 2</td>
<td>X</td>
<td>--</td>
<td>Moderate</td>
<td>--</td>
</tr>
<tr>
<td>Case 3</td>
<td>X</td>
<td>--</td>
<td>Severe</td>
<td>--</td>
</tr>
<tr>
<td>Case 4</td>
<td>X</td>
<td>X</td>
<td>Light</td>
<td>--</td>
</tr>
<tr>
<td>Case 5</td>
<td>X</td>
<td>X</td>
<td>Moderate</td>
<td>--</td>
</tr>
<tr>
<td>Case 6</td>
<td>X</td>
<td>X</td>
<td>Severe</td>
<td>--</td>
</tr>
<tr>
<td>Case 7</td>
<td>--</td>
<td>X</td>
<td>Light</td>
<td>X</td>
</tr>
<tr>
<td>Case 8</td>
<td>--</td>
<td>X</td>
<td>Moderate</td>
<td>X</td>
</tr>
<tr>
<td>Case 9</td>
<td>--</td>
<td>X</td>
<td>Severe</td>
<td>X</td>
</tr>
</tbody>
</table>

The simulation results are shown in Table 10.21. The detection and identification performances of the AIS under atmospheric turbulence have been analyzed in terms of detection/identification rate and false alarms.
Different aspects could be noticed from the results of the Table 10.21, and are discussed as following:

Considering that severe turbulence can be classified as a new abnormal condition, all the cases show an acceptable AIS-detection performance. Notice that for the cases 3 and 6, the heavy turbulence increases significantly the number of false alarms. This implies that at this level, some antibodies start to detect anomalies in the system and the nominal condition is no longer recognized as such.

For case 6, the number of false alarms decreases as compared to case 3. The difference between these two cases is the set of maneuvers performed. Thus, the reduction in the FA can be attributed to the compensation effect of the pilot inputs on the aircraft response. Because the pilot is considered part of the whole system, and the AIS antibodies were trained with such condition, it is expected that the AIS system would work better in this situation.

Regarding the identification performance, the cases 1 to 6 show that when the level of the turbulence is large enough to be detected as abnormal condition, such anomaly is classified as sensor failure with around 60% of identification rate. This implies that more identifiers with better resolution must be generated in order to achieve the correct identification of these two conditions.

In the presence of a sub-system failure (cases 7 through 9) the results show an excellent detection performance. Of course, the presence of two abnormal conditions at the same time will increase the number of activated antibodies. However, the same cases also show that AIS-identification performance decreases with the level of turbulence. This issue

<table>
<thead>
<tr>
<th>Case</th>
<th>DR</th>
<th>FA</th>
<th>Stab.</th>
<th>Aileron</th>
<th>Rudder</th>
<th>Sensor</th>
<th>Struct.</th>
<th>Engine</th>
<th>Unknown</th>
</tr>
</thead>
<tbody>
<tr>
<td>Case 1</td>
<td>--</td>
<td>0</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
</tr>
<tr>
<td>Case 2</td>
<td>--</td>
<td>14.85</td>
<td>0</td>
<td>29.9</td>
<td>0</td>
<td>54.3</td>
<td>14.9</td>
<td>0</td>
<td>0.78</td>
</tr>
<tr>
<td>Case 3</td>
<td>--</td>
<td>94.15</td>
<td>2.43</td>
<td>15.6</td>
<td>0.30</td>
<td>60.03</td>
<td>6.07</td>
<td>0</td>
<td>15.50</td>
</tr>
<tr>
<td>Case 4</td>
<td>--</td>
<td>0.25</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
</tr>
<tr>
<td>Case 5</td>
<td>--</td>
<td>15.11</td>
<td>0</td>
<td>18.24</td>
<td>0</td>
<td>75.18</td>
<td>5.83</td>
<td>0.72</td>
<td>0</td>
</tr>
<tr>
<td>Case 6</td>
<td>--</td>
<td>82.44</td>
<td>1.98</td>
<td>15.89</td>
<td>0.39</td>
<td>57.74</td>
<td>7.01</td>
<td>0</td>
<td>16.95</td>
</tr>
<tr>
<td>Case 7</td>
<td>97.31</td>
<td>--</td>
<td>2.88</td>
<td>87.86</td>
<td>0</td>
<td>2.36</td>
<td>4.26</td>
<td>2.16</td>
<td>0.10</td>
</tr>
<tr>
<td>Case 8</td>
<td>98.28</td>
<td>--</td>
<td>13.33</td>
<td>60.15</td>
<td>0</td>
<td>11.74</td>
<td>10.63</td>
<td>0.47</td>
<td>3.65</td>
</tr>
<tr>
<td>Case 9</td>
<td>99.36</td>
<td>--</td>
<td>14.37</td>
<td>35.14</td>
<td>0</td>
<td>19.48</td>
<td>12.46</td>
<td>0</td>
<td>18.53</td>
</tr>
</tbody>
</table>
is explained by the fact that the identifiers have not been generated using the combined signature of these two types of abnormal condition.

Finally, it is important to mention that the complexity to perform flight tests increases significantly under sub-system failures and severe turbulence. This results in a significant reduction in the amount of flight data available to test the system. Thus, the conclusions about the results of the case 9 must be done carefully, since the size of data of this case is not representative as compared with the others used so far and presented in this thesis. With these limitations, it can be concluded that the AIS performance is robust in the presence of low level of turbulence disturbance even if the system has not be trained for such conditions. If the level of turbulence exceeds certain levels, the AIS-based FDIE scheme is capable to detect the flight condition as abnormal.
Chapter 11 Conclusions and Future Work

An integrated AIS based scheme has been designed, developed, and implemented using a novel Hierarchical Multi-Self strategy. Testing on the WVU flight simulator for an extended region of the flight envelope showed the effectiveness of the HMS for detection, identification, and evaluation of different aircraft subsystem failures/damages. Based on the observation that every self favors the detection of certain types of failures, the HMS demonstrates to be a flexible scheme that can be updated whenever new types of failures need to be considered and/or more features are available in the system to improve the definition of the existing selves or the generation of new ones.

Different solutions and alternatives in the design of the AIS scheme have been explored and the importance of selected feature type and high dimensionality limitations on the FDIE performance have been analyzed. They showed to have an important effect on detection performance and are a critical aspect when designing the best configuration for the AIS. A careful selection of the self space configuration must be made in order to properly balance numerous contradictory effects and increase the capabilities of the AIS for FDIE. The results demonstrate that the AIS paradigm addresses directly the complexity and multi-dimensionality of aircraft dynamic response in the context of abnormal conditions and provides the tools necessary for a comprehensive/integrated solution to the FDIE problem.

The capabilities of the AIS-based scheme to predict or estimate reduction of the flight envelope have been addressed in a general manner. Relevant parameters were selected for every category of failure and their effects on the performance limits of the aircraft have been studied. The achievable states were determined using hyper-sphere evaluators and offline databases of estimators for known failures were created in order to improve the computational efficiency for online estimation process. However, other limitations on performance and handling qualities when new magnitudes of failures are detected can be performed by interpolation from the offline-generated database.
A preliminary robustness analysis was performed in order to determine the level at which the AIS-based scheme can operate correctly. Different flight tests with atmospheric turbulence at different magnitudes showed that the AIS is robust enough to detect the new condition as abnormal. However, more training may be necessary to achieve an excellent FDIE performance under high intensity of turbulence disturbance.

In general, the results in this thesis show that the developed integrated AIS based on the HMS strategy, addresses the issues related to the FDIE problem:

- Multi-dimensionality problems have been avoided by considering smaller hyper-space dimensions within an integrated scheme.
- Multiple types of failures at different magnitude levels have been considered for the FDIE process.
- The excellent identification results show that the developed scheme is able to distinguish more than ten different types of failures.
- Extended area, instead of only one point of the flight envelope, has been considered for training and testing purposes.
- Failure evaluation phase has been addressed by using the AIS paradigm.

The implementation of the proposed AIS-based scheme can potentially have a significant impact on the safety of aircraft operation. The output information obtained from the scheme will be useful to increase pilot situational awareness and determine automated compensation.

In considering future studies for the improvement and further extension of the AIS-based scheme, some recommendations are formulated as following:

- Regarding the self space representation, hyper-shapes other than spheres should be explored and implemented. The ellipsoids for example, as a generalization of spheres, can improve the coverage of the non-self and reduce the number of antibodies significantly. Also, these two types of hyper-shapes together with hyper-rectangles, can be integrated in a hybrid scheme where the most important characteristics of each one are captured and combined to improve the FDIE performance.
- It is important to notice that genetic algorithm tools have been developed to improve the generation of antibodies for the detection phase. However, the same
tools could be used to improve the identification and evaluation phases by reducing the number of generated identifiers. In consequence, this additional process would improve considerably the computing time for online testing of these two phases.

- The process to test the AIS-based scheme for online incoming data consists of comparing every current point against each set of antibodies. If the test point falls inside of the detection radius of an antibody, this detector is called activated. In consequence, the logic output of this process is "1" for activation and "0" for non-activation. More general criteria may be implemented in order to consider the values between "0" and "1". Fuzzy logic becomes a valid candidate to improve the FDIE performance by assigning a testing radius of every incoming data point. In this way, different levels of warnings can be triggered before a failure is declared definitively.

- To improve the indirect evaluation phase, it would be interesting to increase the dimensionality space of the self definitions. Note that only low dimensions were considered in this study, limiting the estimation of the flight envelope reduction to one or two variables. If the effect of the defined constraints on other sets of variables are studied, a more compact and integrated scheme could be developed with the addition of these new variables to the self definition and providing thus, the estimation of operational limits on different states at the same time under abnormal conditions.

- In the present study, a general F15 engine model was used to analyze the FDIE performance of engine type of failures. However, with this model, not enough parameters relevant for the detection of propulsion systems failures were available. It is recommended that a more sophisticated engine model be implemented with which more propulsion failures types can be modeled and more variables can be considered to create different sets of selves. Since the HMS strategy provides a flexible environment; these selves can be easily added as new modules to the AIS-based scheme without affecting the functionality of the already created selves.

- In this thesis, all the variables considered to create the selves are defined in time domain. However, it must be important to explore other space alternatives such as frequency domains where Fourier transforms or even wavelets can be used. For
example, these spaces could capture changes in the parameters due to failures that are not easily accessible in the time space. The HMS would permit the development of a hybrid space within the same scheme.

- Since all systems change constantly or they experiment new environmental conditions during their operational life (e.g. environmental disturbances, aircraft icing, structural improvements or even geometric modifications), it would be interesting and very useful to develop tools that are capable of tuning the sets of self or non-self detectors and evaluators adapting to these new changes. For instance, in the case of turbulence disturbances, even if the system was not trained for such condition, the antibodies would have the capability to adapt to this new condition, moving and exploring new non-self spaces, changing their detection radius, cloning, removing, and/or generating new specialized ones. This adaptive AIS-based scheme would reduce significantly the offline design process by generating and optimizing the antibodies during an online manner and improving considerably the FDIE performance.

- With the developed AIS-based scheme, which ensures high FDIE performance, it would be interesting to perform a forward step towards the extension for adaptive control system purposes. In addition to the specific characteristics of every antibody (e.g. detection radius and center, for hyper-sphere case), other information such as compensation tasks commands can be assigned to each detector. When a detector is being activated, different compensatory actions, previously assigned to this detector, would be triggered to maintain stability and control of the aircraft.

- The analysis presented in this thesis corresponds to data generated from flight simulation tests; however, continuation of this research effort can also include implementation of the scheme using real flight data. Results from this thesis have been used by Sanchez et al., (2009) showing preliminary and promising performance on the application of the AIS paradigm for FDI purposes on an Unmanned Air Vehicle tested in flight. The on-line operation of such schemes has been determined not to cause real-time computation problems. Based on this, the HMS could be implemented within an on-board computer for real-world applications such as supporting adaptive or scheduled fault tolerant flight control
laws, increasing the situational awareness of manned aircraft pilot or UAV operator, and health monitoring of space exploration systems.
References


