11-3-2008

Fault-Tolerant Adaptive Model Predictive Control Using Joint Kalman Filter for Small-Scale Helicopter

Carlos L. Castillo
University of South Florida

Fault-Tolerant Adaptive Model Predictive Control Using Joint Kalman Filter for Small-Scale Helicopter

by

Carlos L. Castillo

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
Department of Electrical Engineering
College of Engineering
University of South Florida

Co-Major Professor: Wilfrido A. Moreno, Ph.D.
Co-Major Professor: Kimon P. Valavanis, Ph.D.
James T. Leffew, Ph.D.
Paris Wiley, Ph.D.
Fernando Falquez, Ph.D.

Date of Approval:
November 3, 2008

Keywords: UAVs, Nonlinear Estimation, Receding Horizon Control, VTOL, Parameter Estimation

Dedication

I dedicate this dissertation to my beloved family:

My wife Arley, my son Christopher and my daughter Stephanie

To my mother and siblings

Acknowledgements

I thank my two co-major professors, Dr. Wilfrido A. Moreno and Dr. Kimon P. Valavanis for their direction, support and the opportunity to collaborate in their research activities. The academic and research environments of the Linear Control Lab and the Unmanned System Lab provided me the opportunity to grow both my academic and research capabilities.

I thank all the members of my committee, which include Dr. James Leffew, Dr. Paris Wiley and Dr. Fernando Falquez, for their time in reviewing both my research and documentation. Their timely feedback provided welcome guidance from the time of the dissertation proposal to the completion of the writing of the dissertation.

I thank all my friends, with whom I have had the pleasure to interact during the journey of earning my Ph.D. degree. I enjoyed the time, which we spent together while working in the Linear Control lab, working on papers, projects and/or as classmates.

This work was also partially supported by grant ARO W911NF-06-1-0069 and grant SPAWAR N00039-06-C-0062.

Table of Contents

List of Tables vi

List of Figures vii

Abstract xiii

Chapter 1 Introduction 1

1.1. Background on Relevant Issues Encountered in the Implementation of Control Systems 7

1.1.1. Uncertainty 7

1.1.2. Robust Stability and Robust Performance 8

1.1.3. Nonlinearities 8

1.1.4. Physical Limitations on Sensors/Actuators 9

1.1.5. Fault Tolerance 9

1.1.6. Adaptability 10

1.2. Research Objectives 10

1.3. Research Methodology 10

1.4. Summary of Contributions 11

1.5. Outline of this Dissertation 11

Chapter 2 UAV Low-level Control Literature Review and Background 13

2.1. Literature Review of the Main UAV Research Groups 13

2.1.1. Carnegie Mellon University 13

2.1.1.1. Classical Control 14

2.1.1.2. Robust Control 14

2.1.2. Massachusetts Institute of Technology 15

2.1.2.1. Classical Control 15

2.1.2.2. Hybrid Control 16

2.1.3. Georgia Institute of Technology 16

2.1.3.1. Neural Networks Control 17

2.1.3.2. Fault-Tolerant Control 18

2.1.3.3. Fuzzy Logic and Neuro-Fuzzy Control 18

2.1.4. University of California, at Berkeley 19

2.1.4.1. Classical Control 19

2.1.4.2. Nonlinear Control 19

2.1.4.3. Model Predictive Control 20

2.1.5. University of Southern California 20

2.1.6. Software Enabled Control, (SEC) 21

2.1.7. University of South Florida 22

2.1.7.1. Classical Control 22

2.1.7.2. Fuzzy Logic Control 22

2.1.7.3. Model Predictive Control 22

2.1.7.4. Robust Control 22

2.2. Literature Review Summary 23

2.3. Background on Adaptive Control 25

2.3.1. Gain Scheduling 27

2.3.2. Model-Reference Adaptive Control 28

2.3.3. Self-Tuning Regulators 29

2.3.4. Adaptive Dual Control 30

2.4. Model Predictive Control 31

2.5. Fault-Tolerant Control 36

2.5.1. Types and Modeling of Faults and Failures 39

2.5.2. Fault Detection Methods 42

2.6. Summary 43

Chapter 3 Estimation 44

3.1. Estimation Theory 44

3.2. Standard Kalman Filter 44

3.3. Extended Kalman Filter 48

3.4. Unscented Kalman Filter 51

3.4.1. The Unscented Transformation 52

3.5. Dual Estimation 55

3.6. Literature Review about Unscented Kalman Filter 58

3.7. Comparison of the Effect of the Sampling Time on the Performance of the EKF and the UKF 62

3.7.1. Simulation Example 1: Vertically Falling Body 62

3.8. Comparison of the Performance of the EKF and the UKF for Parameter Estimation 80

3.8.1. Noise Sensitivity 94

3.9. Discussion 96

Chapter 4 Model Predictive Control Literature Review 98

4.1. Literature Review about Adaptive Model Predictive Control 98

4.2. Literature Review of Fault-Tolerant Model Predictive Control 102

4.3. Summary 104

Chapter 5 Fault-Tolerant Adaptive Model Predictive Control for Flight Systems 105

5.1. Flight Control Systems 105

Chapter 6 Results 110

6.1. Performance Comparison 110

6.2. Stability Test 115

6.3. Passive Fault Tolerance, (Robustness) 118

6.4. Fault-Tolerant Model Predictive Control 123

6.4.1. Fault Case 1 123

6.4.2. Fault Case 2 131

6.4.3. Fault Case 3: Bell Mixer 139

6.4.4. Fault Case 4: Loss of Effectiveness 146

Chapter 7 Conclusions and Future Work 158

7.1. Conclusions 158

7.2. Future Work 160

References 162

About the Author End Page

List of Tables

Table 1:	Appraisal of Capabilities to Handle Some of the Control Issues	23
Table 2:	Comparison of Reconfigurable Control Methods	25
Table 3:	Kalman Filter Algorithm	48
Table 4:	Simulation Time of Call for a Measurement Frequency of 1 Hz and a Simulation Steps Size of 10 ms	70
Table 5:	Equations for the Augmented Mettler's Model for the Estimation of the Stability Derivative, X_u	81
Table 6:	RMSE and RMAE for the Tracking of the Parameter X_u when the Initial Value was -0.061	85
Table 7:	RMSE and RMAE for the Tracking of the Parameter X_u when the Initial Value was -0.183	86
Table 8:	RMSE and RMAE for the Tracking of the Parameter X_u when its Real Value Changed from -0.122 to 0	87
Table 9:	RMSE and RMAE for the Tracking of the Parameter X_u when its Real Value Changed from -0.122 to -0.244	88
Table 10:	RMSE and RMAE for the Tracking of the Parameter X_u when its Real Value Changed from -0.122 to 1	90
Table 11:	RMSE and RMAE for the Tracking of the Parameter X_u when its Real Value Changed from -0.122 to 2	91
Table 12:	RMSE and RMAE for the Tracking of the Parameter X_u when its Real Value Changed from -0.122 to 3	92

List of Figures

Figure 1: Unmanned Aircraft Systems Roadmap 2005 - 2030 3

Figure 2: Classification of UAV Users 5

Figure 3: Yamaha RMAX Commercial UAV 6

Figure 4: A Simplified Block Diagram of a Gain Scheduling Controller 28

Figure 5: A Simplified Block Diagram of a Model Reference Adaptive System 29

Figure 6: A Simplified Block Diagram of a Self-Tuning Regulator System 30

Figure 7: A Simplified Block Diagram of an Adaptive Dual Control System 31

Figure 8: Basic Structure for Model Predictive Control 34

Figure 9: Model Predictive Signals 34

Figure 10: Basic Block Diagram of a Fault-Tolerant Control System 38

Figure 11: Classification of Fault-Tolerant Control Systems 39

Figure 12: Types of Faults and Failures 39

Figure 13: Principle of the Unscented Transformation [92] 52

Figure 14: 2D Example of the Sigma-Point or Unscented Approach [97] 55

Figure 15: Block Diagram of a Dual Kalman Filter 57

Figure 16: Geometry for the Example of a Vertically Falling Body 63

Figure 17: Comparison of the Position Estimation Error of the EKF and the UKF: T_s = 1 Hz, T_{sim} = 10 ms; Fourth-order Runge-Kutta Method 66

Figure 18: Comparison of the Velocity Estimation Error of the EKF and the UKF: T_s = 1 Hz, T_{sim} = 10 ms; Fourth-order Runge-Kutta Method 67

Figure 19: Comparison of the Ballistic Coefficient Estimation Error of the EKF and the UKF: T_s = 1 Hz, T_{sim} = 10 ms; Fourth-order Runge-Kutta Method 67

Figure 20: Comparison of the Position Estimation Error of the EKF and the UKF: T_s = 1 Hz, T_{sim} = 0.1 ms; Fourth-order Runge-Kutta Method 68

Figure 21: Comparison of the Velocity Estimation Error of the EKF and the UKF: T_s = 1 Hz, T_{sim} = 0.1 ms; Fourth-order Runge-Kutta Method 69

Figure 22: Comparison of the Ballistic Coefficient Estimation Error of the EKF and the UKF: T_s = 1 Hz, T_{sim} = 0.1 ms; Fourth-order Runge-Kutta Method 69

Figure 23: Comparison of the Position Estimation Error of the EKF and the UKF: T_s = 100 Hz, T_{sim} = 1 ms; Fourth-order Runge-Kutta Method 71

Figure 24: Comparison of the Velocity Estimation Error of the EKF and the UKF: T_s = 100 Hz, T_{sim} = 1 ms; Fourth-order Runge-Kutta Method 72

Figure 25: Comparison of the Ballistic Coefficient Estimation Error of the EKF and the UKF: T_s = 100 Hz, T_{sim} = 1 ms; Fourth-order Runge-Kutta Method 73

Figure 26: Comparison of the Position Estimation Error of the EKF and the UKF: T_s = 1 Hz, T_{sim} = 10 ms; Euler's Method 74

Figure 27: Comparison of the Velocity Estimation Error of the EKF and the UKF: T_s = 1 Hz, T_{sim} = 10 ms; Euler's Method 74

Figure 28: Comparison of the Ballistic Coefficient Estimation Error of the EKF and the UKF: T_s = 1 Hz, T_{sim} = 10 ms; Euler's Method 75

Figure 29: Comparison of the Position Estimation Error of the EKF and the UKF: T_s = 1 Hz, T_{sim} = 0.1 ms; Euler's Method 76

Figure 30: Comparison of the Velocity Estimation Error of the EKF and the UKF: T_s = 1 Hz, T_{sim} = 0.1 ms; Euler's Method 76

Figure 31: Comparison of the Ballistic Coefficient Estimation Error of the EKF and the UKF: T_s = 1 Hz, T_{sim} = 0.1 ms; Euler's Method 77

Figure 32: Comparison of the Position Estimation Error of the EKF and the UKF: T_s = 100 Hz, T_{sim} = 1 ms; Euler's Method 78

Figure 33: Comparison of the Velocity Estimation Error of the EKF and the UKF: T_s = 100 Hz, T_{sim} = 1 ms; Euler's Method 78

Figure 34: Comparison of the Ballistic Coefficient Estimation Error of the EKF and the UKF: T_s = 100 Hz, T_{sim} = 1 ms; Euler's Method 79

Figure 35: Noisy Translational Velocities, *u*, *v*, and *w* 82

Figure 36: Noisy Rotational Rates, *p*, *q*, and *r* 83

Figure 37: Tracking of the Parameter X_u from an Incorrect Value of -0.062. The Real Value was -0.122 84

Figure 38: Tracking of the Parameter X_u from an Incorrect Initial Value of -0.183. The Real value of X_u was -0.122 85

Figure 39: Tracking of the Parameter X_u when its Real Value Changed from -0.122 to Zero. A Correct Initial Value was used in the Simulation 86

Figure 40: Tracking of the Parameter X_u when its Real Value Changed from -0.122 to -0.244. A Correct Initial Value was Used in the Simulation 88

Figure 41: Tracking of the Parameter X_u when its Real Value Changed from -0.122 to 1. A Correct Initial Value was Used in the Simulation 89

Figure 42: Tracking of the Parameter X_u when its Real Value Changed from -0.122 to 2. A Correct Initial Value was Used in the Simulation 90

Figure 43: Tracking of the Parameter X_u when its Real Value Changed from -0.122 to 3. A Correct Initial Value was Used in the Simulation 91

Figure 44: Tracking of the Parameter X_u when its Real Value Changed from -0.122 to -3 92

Figure 45: Responses of the Filters when the X_u Parameter Changed its Value from -0.122 to -3. (a) Estimates of the State *u*, (b) Estimates of the State θ 93

Figure 46: Response of the Filters to a Noiseless System: Factor = 10^{-10} 94

Figure 47: Response of the Filters to a Moderately Noisy System: Factor = 10^{-2} 95

Figure 48: Response of the Filters to the Original Noisy System: Factor = 1 95

Figure 49: Typical Low-level Flight Control System Architecture 105

Figure 50: Generic Block Diagram of the Fault-Tolerant Adaptive Model Predictive Controller 108

Figure 51: FTA-MPC Flight Control System 109

Figure 52: *u* and *v* Response of the System in the Nominal Case, No Fault 111

Figure 53: *w* and *r* Response of the System in the Nominal Case, (No Fault) 112

Figure 54: *x* and *y* Response of the System in the Nominal Case, (No Fault) 113

Figure 55: *z* and Ψ Responses of the System in the Nominal Case, (No Fault) 114

Figure 56: 3D Plot of the Response of the System to the Double Circle with Varying Altitude Trajectory in the Nominal Case, (No Fault) 115

Figure 57: Stability Test of the System under Nominal Conditions with the Initial States/Outputs Given by y0 = [6,-1,-1,-1,-1,-1,-1] 116

Figure 58: Stability Test of the System under Nominal Conditions with Initial Values Given by y0 = [6,-1,-1,-1,-1,-1] 117

Figure 59: Stability Test of the System under Nominal Conditions with Initial Values Given by y0 = [6,-6, 5,-5, 4,-4] 118

Figure 60: *u* and *v* Responses of the System When a Fault Occurs, (X_u = 0.3) 119

Figure 61: *w* and *r* Responses of the System When a Fault Occurs, (X_u = 0.3) 120

Figure 62: *x* and *y* Responses of the System When a Fault Occurs, (X_u = 0.3) 121

Figure 63: *z* and Ψ Responses of the System When a Fault Occurs, (X_u = 0.3) 122

Figure 64: 3D Plot of the Response of the System to the Double Circle with Varying Altitude Trajectory when the Parameter Xu was Equal to 0.3, (Fault) 123

Figure 65: Response of the Estimated X_u Parameter: Fault Case 1 124

Figure 66: X_u Covariance: Fault Case 1 125

Figure 67: *u* Response of the Control System: Fault Case 1 126

Figure 68: *x* Response of the Control System: Fault Case 1 127

Figure 69: *y* Response of the Control System: Fault Case 1 128

Figure 70: *z* Response of the Control System: Fault Case 1 129

Figure 71: Ψ Response of the Control System: Fault Case 1 130

Figure 72: 3D Response of the Control System: Fault Case 1 131

Figure 73: Response of the Estimated X_u Parameter: Fault Case 2 132

Figure 74: X_u Covariance: Fault Case 2 133

Figure 75: *u* Response of the Control System: Fault Case 2 134

Figure 76: *x* Response of the Control System: Fault Case 2 135

Figure 77: *y* Response of the Control System: Fault Case 2 136

Figure 78: *z* Response of the Control System: Fault Case 2 137

Figure 79: Ψ Response of the Control System: Fault Case 2 138

Figure 80: 3D Response of the Control System: Fault Case 2 139

Figure 81: Response of the Estimated A_c Parameter: Bell Mixer Fault 140

Figure 82: A_c Covariance: Bell Mixer Fault 141

Figure 83: *u* Response of the Control System: Bell Mixer Fault 142

Figure 84: *x* Response of the Control System: Bell Mixer Fault 143

Figure 85: *y* Response of the Control System: Bell Mixer Fault 144

Figure 86: *z* Response of the Control System: Bell Mixer Fault 145

Figure 87: *Ψ* Response of the Control System: Bell Mixer Fault 146

Figure 88: Response of the Estimated Z_{col} Parameter: LOE Fault 147

Figure 89: Z_{col} Covariance: LOE Fault 148

Figure 90: Response of the Estimated N_{col} Parameter: LOE Fault 149

Figure 91: N_{col} Covariance: LOE Fault 150

Figure 92: *v* Response of the Control System: LOE Fault 151

Figure 93: *w* Response of the Control System: LOE Fault 152

Figure 94: r Response of the Control System: LOE Fault 153

Figure 95: x Response of the Control System: LOE Fault 154

Figure 96: y Response of the Control System: LOE Fault 155

Figure 97: z Response of the Control System: LOE Fault 156

Figure 98: Ψ Response of the Control System: LOE Fault 157

Fault-Tolerant Adaptive Model Predictive Control Using Joint Kalman Filter for Small-Scale Helicopters

Carlos L. Castillo

ABSTRACT

A novel application is presented for a fault-tolerant adaptive model predictive control system for small-scale helicopters. The use of a joint Extended Kalman Filter, (EKF), for the estimation of the states and parameters of the UAV, provided the advantage of implementation simplicity and accuracy. A linear model of a small-scale helicopter was utilized for testing the proposed control system. The results, obtained through the simulation of different fault scenarios, demonstrated that the proposed scheme was able to handle different types of actuator and system faults effectively. The types of faults considered were represented in the parameters of the mathematical representation of the helicopter.

Benefits provided by the proposed fault-tolerant adaptive model predictive control systems include:

- The use of the joint Kalman filter provided a straightforward approach to detect and handle different types of actuator and system faults, which were represented as changes of the system and input matrices.

- The built-in adaptability provided for the handling of slow time-varying faults, which are difficult to detect using the standard residual approach.
- The successful inclusion of fault tolerance yielded a significant increase in the reliability of the UAV under study.

A byproduct of this research is an original comparison between the EKF and the Unscented Kalman Filter, (UKF). This comparison attempted to settle the conflicting claims found in the research literature concerning the performance improvements provided by the UKF. The results of the comparison indicated that the performance of the filters depends on the approximation used for the nonlinear model of the system. Noise sensitivity was found to be higher for the UKF, than the EKF. An advantage of the UKF appears to be a slightly faster convergence.

Chapter 1

Introduction

Since their inception, control systems have been an enabling technology, [2]. Control systems were introduced during the industrial revolution with devices like the James Watt flyball governor, [1], [2]. Over the past 40 years, the developments in analog and digital electronics have resulted in dramatic increases in the computational power of microcomputers and microcontrollers. These developments provided for the implementation of advanced control techniques. These advanced control techniques enabled the successful development of high performance applications such as:

- Guidance and control systems for aerospace vehicles such as commercial aircraft, guided missiles, advanced fighter aircraft, launch vehicles and satellites. These control systems provide stability and tracking in the presence of large environmental and system uncertainties, [2].
- Control systems in the manufacturing industries from automotive to integrated circuits, which are associated with computer-controlled machines, provide the precise positioning and assembly required for high-quality, high-yield fabrication of components and products, [2].
- Industrial process control systems, particularly in the hydrocarbon and chemical processing industries, maintain high product quality. Product

quality is maintained by monitoring thousands of sensors signals and making corresponding adjustments to hundred of valves, heaters, pumps and other actuators, [2].

- Control of communication systems such as the telephone system, cell phones, and the Internet are especially pervasive. These control systems regulate the signal power levels in transmitters and repeaters, manage packet buffers in network routing equipment and provide adaptive noise cancelation to respond to varying transmission line characteristic, [2].

Control systems have reached a high level of theoretical development and there exists a myriad of applications. However, the development of new sensors and actuators for old and new applications continues. Therefore, the demand for new theoretical concepts and approaches, to handle increasingly complex applications remains high.

The development of flight control systems for UAVs is a relative new application of advanced control techniques. Due to the successful use of unmanned aircrafts, (UAs), in the Global War on Terrorism, (GWOT), an enormous interest has developed for increasing their contributions in sorties, hours and expanded roles. As of September 2004, some twenty types of coalition unmanned aerial vehicles, (UAVs), large and small, have flown over 100,000 flight hours in support of Operation ENDURING FREEDOM, (OEF), and Operation IRAQI FREEDOM, (OIF), [3]. Previously, the only application for UAV's was as reconnaissance vehicles. However, current applications include strike, force protection and signals collection, which have helped to reduce the complexity and time lag in the sensor-to-shooter chain for acting on "actionable intelligence". UA

systems, (UAS), continue to expand and encompass a broad range of mission capabilities. Figure 1 presents the expected evolution or trend for UAV systems.

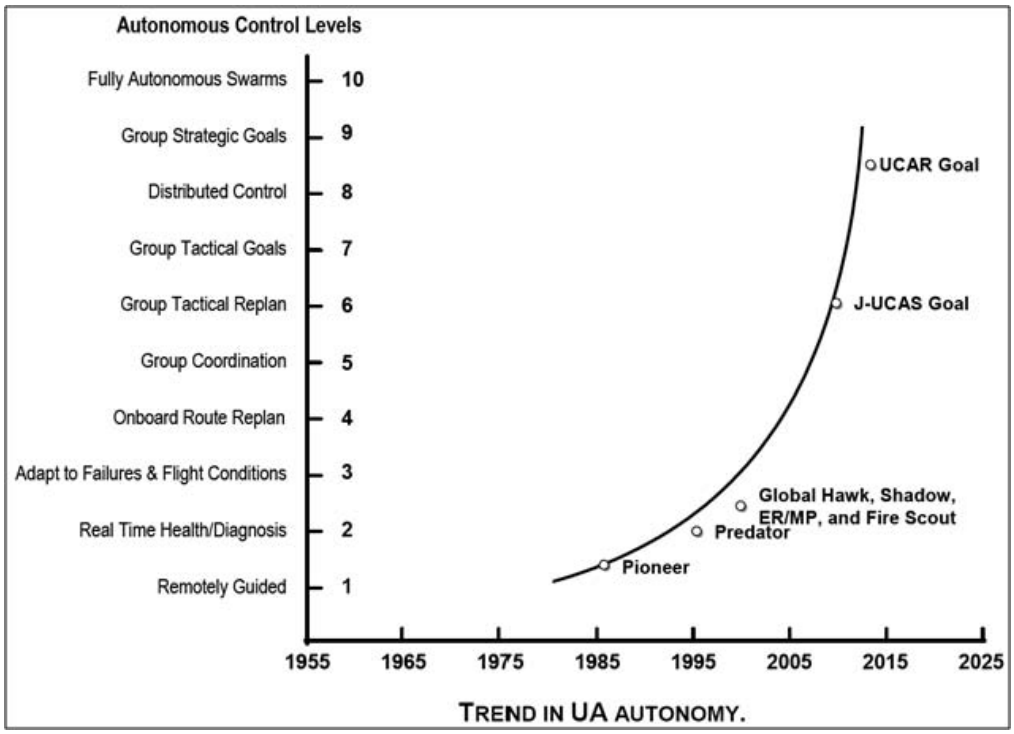


Figure 1: Unmanned Aircraft Systems Roadmap 2005 - 2030

The trend associated with increases in the capabilities and complexity of UAVs is expected to grow enormously. The latest successes of UAVs applications have been impressive. However, several crashes have raised concerns about their reliability. Consequently, a need to improve UAV's reliability has become a very important subject. The Office of the Secretary of Defense has acknowledged the significance of UA reliability by stating that "Improving UA reliability is the single most immediate and long -reaching need to ensure their success", [3]. Fault-tolerance and adaptability to

unpredictable flight conditions will be fundamental for increasing the reliability of UAVs.

Currently, most of the UAVs, which are associated with military applications, are fixed wind airplanes. However, as part of the Future Combat Systems, (FCS), initiative, it was recommended that several types of Vertical Take-Off and Landing, (VTOL), UAs be developed. VTOL UA vehicles will provide reconnaissance, surveillance and target acquisition assistance for ground troops. VTOL UA vehicles will offer major advantages over fixed-wing UAs. The Future Combat Systems initiative was formerly known as the Future Ground Combat Systems program

In addition, to military applications for UAVs there are civil and commercial applications. These applications include search and rescue, traffic monitoring, demining, forest fire detection, border patrol, filming industry and dam inspections. Carrier companies such as FedEx and UPS have expressed interest in unmanned vehicles for long-haul cargo duty, [4]. A NASA Civil UAV Capability Assessment indicating the diverse user spectrum for UAVs is presented in Figure 2.

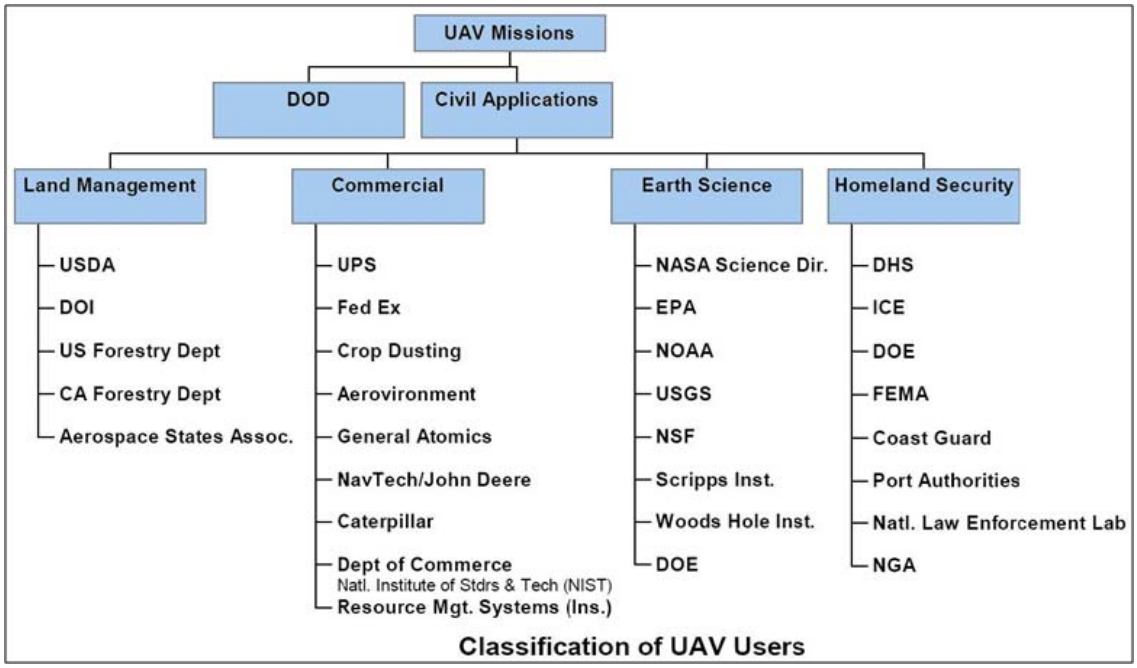


Figure 2: Classification of UAV Users

Many universities around the world have advocated the development of research platforms for the development of small-scale rotorcraft as UAV prototypes. These platforms have the goal of allowing the proof of concepts of new algorithms to tackle some of the challenging problems associated with the development of an autonomous vehicle. Fault detection and identification, (FDI), fault-tolerant flight control systems, path planning, obstacle avoidance and cooperative control are some of the many problems, which have to be resolved. Figure 3 provides a picture of a popular commercial UAV, which is used at several university research laboratories.

Figure 3: Yamaha RMAX Commercial UAV
(http://uav.ae.gatech.edu/pics/gtmax/)

The development of UAV flight control systems, which are capable of obtaining the autonomous control level indicated in Figure 1, is a challenging task. In order to achieve high levels of autonomous control, it is necessary to address some of the typical issues encountered in the implementation of advanced control systems. These issues will be reviewed briefly.

1.1. Background on Relevant Issues Encountered in the Implementation of Control Systems

The diversity of application areas where control systems are used or will be used makes the characterization of all the possible controllable systems an almost unreachable task. Therefore, this research focused on some of the most relevant issues encountered when addressing the challenge of implementing advanced control techniques.

1.1.1. Uncertainty

When a control engineer must obtain a desired behavior from plants, the main reason that forces the use of closed-loop control systems is uncertainty. The absence of uncertainty would allow the implementation of control systems without the use of feedback. Feedback introduces cost, complexity and possibly instability. Uncertainty is one of the major issues to be dealt with for the practical implementation of control systems. Uncertainty can be classified either as disturbance signals or as dynamic perturbations. The former includes input and output disturbances such as a gust on an aircraft, sensor noise and actuator noise. The later represents the discrepancy between the mathematical model and the actual dynamics of the system in operation, [5]. Most of the relevant control techniques, developed through decades by the control research community, are model-based techniques. The use of a *mathematical* model of the system has been fundamental for the enormous development obtained in control theory. However, it is considered that models will never provide exact representations of the true system, [6]. The development of a model inherently produces uncertainties due to unmodeled dynamics, neglected nonlinearities, system-parameter variation due to

environmental changes and torn-and-worn factors. These and other factors render modeling the exact behavior of physical systems impossible. Therefore, there exists the need of representing and taking into account these uncertainties in the control design.

1.1.2. Robust Stability and Robust Performance

In order to be useful or even practical, a closed-loop control system has to be stable under certain specified levels of uncertainty. This is the concept of robust stability. The nominal stability is obtained when the closed-loop is stable assuming zero uncertainty. Following the same idea, the concepts of nominal and robust performance can be developed. The performance can be specified in the time domain, in the frequency domain or, as is typical, in both domains. Given its importance, a considerable effort is normally dedicated to guarantee robust stability. Robust control methods are one of the standard ways to deal with uncertainty in dynamical systems.

1.1.3. Nonlinearities

Every physical system has nonlinearities to some extent. However, the use of linear models to represent the local behavior of nonlinear systems has been used for decades with great success in a vast number of applications in many different fields. The linear approach to control of the nonlinear plant is theoretically based in the so-called first theorem of Lyapunov.

1.1.4. Physical Limitations on Sensors/Actuators

Any kind of electronic or mechanic devices such as sensors or actuators will have some kind of maximums and/or minimums limits, in their specifications. Valves, a common actuator used in the process industry are limited by maximum flow rates, which they can provide.

1.1.5. Fault Tolerance

Stringent requirements for safety, reliability and profitability are demanded for the chemical and manufacturing industries. These requirements have generated the necessity of designing control systems with the ability of handling defects/malfunctions in process equipment, communication networks, sensors and actuators, [8].
Issues related to faults may include physical damage to the process equipment, misuse of raw material and energy resources, increase in the downtime for process operation resulting in significant production losses and jeopardizing personnel and environmental safety [7]. Management of abnormal situations is a challenge in the chemical industry since abnormal situations account annually for 10 billion in lost revenue in the U.S. alone, [8]. Aside from the economical implications, which failures in technological systems imply, the loss of life is also a fundamental reason for designing control systems capable of handling systems' components faults or failures. Reliability and operational safety is one of the main research focus areas in the design of current and future control systems of UAVs.

1.1.6. Adaptability

System dynamics change considerably when their operating conditions change. Aircraft and helicopters are typical examples of these types of systems. The controllers of these systems need to possess mechanisms to account for varying system characteristics. A common way to deal with this issue is to use adaptive based control systems. Adaptive control methods are also considered as an approach to handle the uncertainty of dynamical systems.

1.2. Research Objectives

The main objective of this research was the study of the use of the model predictive control (MPC) technique, as the primary approach to be employed for a novel development of fault-tolerant and adaptive flight control systems for small-scale helicopters.

1.3. Research Methodology

An extensive literature review of low-level control of UAVs was performed as the starting point of this research. Based on the literature review, the frameworks for fault-tolerant control, adaptive control and model predictive control were selected. The framework developed can be described as a hybrid approach to be applied to small-scale helicopters. Additional literature reviews were carried out for adaptive model predictive control and fault-tolerant MPC.

An outcome of this research, which was motivated by the literature review, was a performance comparison study between the Extended Kalman Filter and the Unscented

Kalman Filter. This comparison study attempted to provide insight into the reasons for the conflicting results found during the literature review.

The proposed Fault-Tolerant Adaptive Model Predictive controller was tested in simulations, for several fault case studies.

1.4. Summary of Contributions

This research provided the following contributions:

- A novel application of a fault-tolerant adaptive MPC to a small-scale helicopter was developed and validated using computer simulations.
- The Joint Extended Kalman filter was employed for parameter estimation of the helicopter's aerodynamic coefficients. This approach provided an accurate and simple approach for implementing the adaptive mechanism of the controller and an implicit implementation of the FID function.
- A novel comparison of the Extended Kalman Filter and the Unscented Kalman filter was developed. The comparison provided insights into the different claims related to the improved performance of the Unscented Kalman filter.

1.5. Outline of this Dissertation

Chapter 2 presents an UAV low-level control literature review and a brief background related to the control concepts and techniques, which were used during this research.

Chapter 3 presents a brief description of Estimation theory, which provides the theoretical background for the standard Kalman Filter, the Extended Kalman Filter, the Unscented Kalman Filter, (UKF), and their use for parameter estimation. A literature review of the Unscented Kalman filter is presented. In addition, a novel comparison of the EKF and the UKF is presented.

Chapter 4 presents a literature review associated with adaptive and fault- tolerant model predictive control.

Chapter 5 presents the control architecture proposed and implemented during this research.

Chapter 6 presents the results obtained for the UAV fault-tolerant adaptive model predictive control.

Chapter 7 presents the conclusions derived from this research. In addition, foreseeable future work, which is envisioned from the details and particularities encountered during the implementation of the proposed control architecture, is outlined.

Chapter 2

UAV Low-level Control Literature Review and Background

This chapter presents a review of the literature associated with UAV low-level control. Additionally, a background review of the basic ideas associated with adaptive control, model predictive control and fault-tolerant control is presented.

2.1. Literature Review of the Main UAV Research Groups

The last decade has seen a strong interest in the development of Unmanned UAVs. Many universities, [9], [10], [11], research institutes, [12], and companies, [13], [14], [15], [16] have dedicated enormous efforts to building UAVs prototype. Some aspects considered in the implementation of UAVs are the type of aeronautic platform, the computational platform, the operating system, the path planning algorithms, low-level control techniques and sensors. The intent of the review was to find the latest contributions from the main research groups in the area of low-level control techniques.

2.1.1. Carnegie Mellon University

The Carnegie Mellon University Robotics Institute, (CMURI), is arguably the first research group that implemented vision-based techniques for navigation. Since 1991, researchers at CMURI have been working in Vision-based control of small-scale

helicopters. In addition, development of several helicopter research platforms was undertaken, [17]. The primary focus of their research from 1991-1997 was the development of vision based navigation and sensor fusion. In 1997, the CMURI won the International Aerial Robotics Competition, (IARC), which was held at Disney World's Epcot Center, [18], [19].

2.1.1.1. Classical Control

In 1999, Bernard Mettler extended the application of the Comprehensive Identification from FrEquency Responses, (CIFER), a integrated software packages of system identification tools for full size helicopters, to the Yamaha R50, which is a fully instrumented small-scale helicopter, [20]. An accurate, high-bandwidth, linear state-space model was derived for both the hover and the cruise flight conditions. The model structure included the explicit representation of coupled rotor-flap dynamics and rigid-body fuselage dynamics, and the yaw damper dynamics. In 2000, Mettler's continuation of this research presented a new 13^{th} order linear state-space helicopter model, which explicitly accounted for the coupled rotor/stabilizer/fuselage, (r/s/f), dynamics in the hover and cruise modes, [21]. Optimization based tuning was performed utilizing the CONtrol Designer's Unified InTerface, (CONDUIT), computational facility and the developed model in order to implement classical control techniques.

2.1.1.2. Robust Control

In 2001, a control design technique based on Reinforcement Learning Policy Search Methods was presented, [22]. The control problems within the robotics field are

treated as a Partially Observed Markovian Decision Problem, which is a type of optimal control formalism. The idea is to "learn" the control law from the data obtained from experiments, which produce a minimum value for certain performance criteria.

In 2002, Marco La Civita presented a novel modeling technique called MOdeling for Flight Simulation and Control Analysis, (MOSCA), [23]. In 2003, Marco La Civita implemented a gain-scheduled H_∞ loop-shaping controller for the Yamaha R-50 helicopter, [24], [25].

2.1.2. Massachusetts Institute of Technology

The Massachusetts Institute of Technology, Boston University and Draper Laboratory developed an autonomous helicopter, which won the 1996 International Aerial Robotics Competition, (IARC), [10]. The control system was implemented utilizing four control loops. In addition, this group has provided significant contributions to the development of small-scale helicopter nonlinear models, [26]. More recently, the MIT aerial robotics group has been dedicated to providing research in the areas of hybrid control architecture, [27], [28], and path planning for Multiple UAVs, [29], [30].

2.1.2.1. Classical Control

During the 1998 IARC, MIT's Aerial Robotics Club presented the “chopter'98”. The vehicle was based on a “Bergen Industrial Hel”, (BIH), and included a series of modified off-the-shelf products. Some improvements were made in the software implementation. However, there was very little significant change in the control system,

[9]. Classical controllers were used for the roll, pitch, yaw and collective/throttle with feed forward gain implementation on some of the variables.

2.1.2.2. Hybrid Control

In 1999, a hybrid controller based on an automaton whose states represent feasible trajectory primitives was developed. A control system for aggressive maneuver of an autonomous helicopter is presented in, [31], [32]. The main idea was based on incorporating a *maneuver automaton* for selecting optimally different control laws according to the motion primitives, which required executed. The maneuver automaton concept was developed further and tested in simulation within the framework of the Software Enabled Control program, (SEC), [33]. Nonlinear control techniques such as the Back Stepping Algorithm, [34], and Linear Quadratic Control techniques, [35], have also been researched.

2.1.3. Georgia Institute of Technology

The Georgia Institute of Technology research group is arguably the one group that has contributed the most to the UAV field. This group has collected the most IARC competition prizes. In addition, they have played the crucial role in developing and implementing the DARPA SEC program.

The Georgia Institute of Technology research group has provided major contributions related to the control of unmanned helicopters. A prototype implementation of OCP, [36], [33], was developed in the form of a fully rigged autonomous helicopter, which incorporated a fault detection and identification module to

compensate for collective actuator failures. A control design methodology for accommodating different flight modes and limit avoidance through mode transition controllers was presented, [37], [38]. An experimental platform called GTMax, which included a Yamaha RMAX helicopter with full avionic instrumentation, a simulation model of the helicopter, a ground control station and all baseline on-board routines, was developed. The first two components run on Windows platforms and the onboard routines run under QNX, [39], [40], [41].

2.1.3.1. Neural Networks Control

In 1994, a direct adaptive tracking control architecture using neural networks, (NN), and a nonlinear controller based on feedback linearization was studied, [42]. In 1999, an adaptive nonlinear controller using a combination of feedback linearization and a neural network for on-line adaptation was presented, [43].

Johnson et al, [44], [45], [46], developed an adaptive control scheme based on NN and a method termed Pseudo-Control Hedging, (PCH), was presented. The purpose of PCH was to prevent the adaptive element of an adaptive control system from adapting to selected plant input characteristics.

Calise and Rysdyk, [47], presented a robust nonlinear adaptive flight control system, which utilized model inversion control with an adaptive neural network. This flight control system was oriented to provide consistent handling qualities for piloted unconventional modern aircraft like a tiltrotor.

2.1.3.2. Fault-Tolerant Control

Idan et al, [48], presented a fault-tolerant flight control system, which blended aerodynamic and propulsion actuation for safe flight operation in the presence of actuator failures. Fault tolerance was obtained using the nonlinear adaptive control scheme and the previously developed PCH.

Drozeski et al, [49], [50], presented a fault-tolerant control architecture, which coupled techniques for fault detection and identification with reconfigurable flight control to augment the reliability and autonomy of a UAV. An adaptive, neural network, feedback linearization technique was employed to stabilize the vehicle after the detection of a fault.

2.1.3.3. Fuzzy Logic and Neuro-Fuzzy Control

In 1997, Fuzzy Logic was used to implement critical vehicle modules as the route planner, the fuzzy navigator, the fault-tolerant tools and the flight controller, [51]. An adaptive mode transition control technique was presented, [37], [38], which cited additional references. The control technique consisted of an on-line adaptation of the parameter of mode transition controllers designed off-line via the method of blending local mode controllers, (BLMC). The adaptation scheme was composed of a desired transition mode to be adapted. The desired transition model, the active plant model and the blending gains portion of the active controller model were represented via a fuzzy neural network. Valenti et al, [33], handled the control problems of limit detection and avoidance by constantly redefining artificial limits on the actuators.

2.1.4. University of California, at Berkeley

The BErkeley AeRobot, (BEAR), research team at UC Berkeley has consistently contributed to the field of VTOL type UAVs since 1996, [52]. Recent research published by the UC deals with control of multiple UAVs, [53], and the incorporation of obstacle avoidance strategies for navigation in urban environments, [54].

2.1.4.1. Classical Control

Kim et al, designed a multi-loop PD controller, [55], and compared it with a nonlinear model predictive controller.

2.1.4.2. Nonlinear Control

In 1996, [56], a nonlinear controller was presented to deal with tracking in non-minimum phase nonlinear systems with inputs. The method was applied to simplified planar dynamics of VTOL and CTOL aircraft. In 1998, [57], the output tracking control design of a helicopter model based on approximate input-output linearization was compared with the exact linearization. Depending of the selection of output variables, exact linearization can produce unstable zero dynamics. It was shown that by neglecting the coupling between the forces and the moments, the approximate system with dynamics decoupling is full state without zero dynamics by choosing positions and heading as outputs.

2.1.4.3. Model Predictive Control

Kim et al, [55], [58], presented a nonlinear model predictive controller. The on-line optimization was implemented using the gradient-descent method. The computational load of this nonlinear model predictive tracking control was claimed to be low enough for real-time control of rotorcraft unmanned aerial vehicles. Shim et al, [53], [54], [59], treated the vehicle control, with state constraints and input saturation, as an optimization of a model predictive control framework. The optimization considered cost functions including penalties for obstacle avoidance or symmetric pursuit-evasion games, [60], [61].

2.1.5. University of Southern California

Research at the University of Southern California, (USC), started in 1991 with the first version of an Autonomous Flying Vehicle, (AFV). The AFV won the IARC competition in 1994 with the first generation of Autonomous Vehicle Aerial Tracking and Retrieval, (AVATAR), helicopters, [11]. The AVATAR software and control architecture was further explained in, [62], along with other research efforts in autonomous landing and vision-based state estimation. The AVATAR main feature was its hierarchical behavior-based control architecture with all behaviors acting in parallel at different levels. An autonomous landing approach on a moving target and visual surveying in urban areas are the topics discussed in, [63], and, [64]. Behavior-based architectures for helicopter control have also been reported, [11], [65].

2.1.6. Software Enabled Control, (SEC)

The Software Enable Control, (SEC), program began in late fiscal year 1999 under Defense Advanced Research Projects Agency, (DARPA), funding and sponsorship to address, among other issues, the search for solutions, which would lead to greater levels of autonomy in man-made systems. Realization of complex controls for such systems involves major computational complexity concerns and requires computationally efficient techniques, which can be implemented in real-time. Therefore, computing plays a prominent role when dealing with such complex controls and systems.

The primary focus of the SEC program was to advance control technologies, which improve UAV performance, reliability and autonomy. One of the main results was derivation and implementation of an Open Control Platform, (OCP), which enabled development and deployment of control functions in terms of objects. In OCP, object-oriented control components are distributed across embedded platforms and enable coordination and cooperation among UAVs, [33]. A component-based design environment called Ptolemy was developed and integrated with OCP. Ptolemy provided for model-based control design of heterogeneous systems. Ptolemy also accounted for the hybrid nature of most technical systems and different models of computation. The SEC program provided major contributions of in the field of low level VTOL control. The program accounted for several model predictive controls, (MPC), strategies and the so-called mode transition controller, which blended different linear controllers according to the corresponding appropriate flight mode. To date, the SEC program has been the most comprehensive effort involving major companies and Universities across the US.

2.1.7. University of South Florida

2.1.7.1. Classical Control

A decentralized control system, based on multi-loop PID controllers, was implemented, [66], [103]. The control system designed focused on non-aggressive flights. Tuning of the PIDs was obtained using optimization methods. A decentralized control system based on multi-loop two-degrees of freedom PIDs was designed following the "one loop at the time" approach to guarantee good phase and gain margins

2.1.7.2. Fuzzy Logic Control

A decentralized control system, based on multi-loop PID-like Fuzzy controllers, was implemented, [103]. The control system designed focused on non-aggressive flights. Tuning of the Fuzzy Logic controllers was obtained using optimization methods.

2.1.7.3. Model Predictive Control

A Model Predictive Control Based Trajectory Tracking, (MPCTT), system for UAVs was presented in, [67]. Simulation results demonstrated the superiority of the proposed MPCTT approach. MPCTT required substantially less control effort in order to track waypoint trajectories.

2.1.7.4. Robust Control

A practical and simple approach to the design of UAVs was based on a standard, easily tunable, PID control as the starting step of the design, [68]. Then, robust loop-

shaping techniques were applied to derive a controller with optimal properties with respect to robustness, noise sensitivity and bandwidth.

2.2. Literature Review Summary

The previous low-level control literature review reveals that almost any existent control method has been used. However, only a few cases of fault-tolerant control for application to small-scale helicopter or unmanned aerial vehicles were presented to the literature.

In order to confront the most important issues of flight control systems for UAVs, it is the author's conviction that a convergence of several advanced control techniques is required to accomplish the challenging task of developing a safe and reliable flight control system. Table 1 presents a personal appraisal of the capabilities of some of the most successful advanced control techniques available.

Table 1: Appraisal of Capabilities to Handle Some of the Control Issues

	Nonlinearity	MIMO	States/Inputs Constraints	Robustness to Uncertainty	Strong Coupling	Fault-Tolerant
Classical	low/medium	low/medium	low	low	low	low
LQR/LQG	low	high	low/high	low	high	low
Adaptive	high	high	low	low	high	medium/high
Robust	medium/high	high	low	high	high	medium/high
Nonlinear	High	high	medium/high	medium/high	high	medium/high
Predictive	medium/high	high	high	medium/high	high	medium/high
Neural	high	high	medium/high	low/medium	high	medium/high
Fuzzy	high	low	low	low	medium/high	medium/high
Hybrid	medium/high	medium/high	low	low	medium/high	medium/high

The adaptive control approach is the only control technique, which has been used to improve the reliability of UAVs, [48], [49], [50]. This approach was realized within the framework of fault-tolerant control systems.

Colin N. Jones, [69],presented a recent report on Reconfigurable Flight Control. That report suggests that model predictive control presents intrinsic properties, which allow it to handle easily some typical actuator faults. The report presents the application of fault-tolerant model predictive control to the case of the Flight EL AL 1862, [109]. Table 2 , which was extracted from the report, indicates that, to date, only Model Predictive Control has the potential for solving the general reconfigurable control problem [69]. Filled circles mean that the method has the property, while empty circles imply that an author has suggested that the approach could be modified to incorporate the property.

Table 2: Comparison of Reconfigurable Control Methods

Method	Failures		Robust	Adaptive	Fault Model			Model Type	
	Actuator	Structural			FDI	Assumed	Constraints	Linear	Nonlinear
Multiple Model Switching and Tuning (MMST)		•		•	•			•	
Interacting Multiple Model (IMM)		•		•	•		o	•	
Propulsion Controlled Aircraft (PCA)	•		o			•		•	•
Control Allocation (CA)	•					•	o	•	
Feedback linearization	•	•		•	•				•
Sliding Mode Control (SMC)	o[1]	•	•[2]				•		•
Eigenstructure Assignment (EA)		•				•		•	
Pseudo Inverse Method (PIM)		•				•		•	
Model Reference Adaptive Control (MRAC)		•		•	•			•	o
Model Predictive Control (MPC)	•	•	o	o	•	•	•	•	•

Based on the UAV low-level control literature review, the appraisal presented in Table 1 and the comparison of reconfigurable control methods presented in Table 2, this research focused on application of the fault-tolerant control framework to small-scale helicopters using an adaptive model predictive control approach.

2.3. Background on Adaptive Control

The origins of Adaptive Control can be traced back to the early 1950s, [70], [71], when an extensive effort in the design of autopilots for high-performance aircraft, like the X-15 experimental aircraft, was begun. Since such aircraft operated over a wide range of speeds and altitudes, [70], aerodynamic characteristics changed considerably. Therefore,

---

[1] SMC can handle partial loss of effectiveness of actuators, but not complete loss.
[2] SMC assumes robust control can handle all forms of structural failures.

it was necessary to use more sophisticated control techniques than the simple linear time-invariant constant-gain feedback controllers. These controllers were not able to operate throughout the complete flight envelope. A heuristic approach called Gain scheduling was determined to be a suitable technique for flight control systems.

A universally accepted definition of adaptive control systems does not exist. Åström and Wittenmark, [70], proposed "A*n adaptive controller is a controller with adjustable parameters and a mechanism for adjusting the parameters*". More recently, Filatov and Unbehauen, [73], proposed the definition, "*A control system operating under conditions of uncertainty of the controller that provides the desired system performance by changing its parameters and/or structure in order to reduce the uncertainty and to improve the approximation of the desired system is an adaptive control system*."

During the decades after the beginnings of adaptive control, researchers in the adaptive control field devised the following main types of adaptive systems, [70]:

- Gain Scheduling,
- Model-reference adaptive control,
- Self-tuning regulators,
- Dual control.

The first three methodologies are based in the *certainty equivalence*, (CE), principle or approach. The *certainty equivalence* principle consist of assuming that the parameters estimates are the true parameters values of the model, ignoring the uncertainty of the estimation, and these estimates are used for the controller design. Most of the current adaptive approaches are based on the CE principle, [70], [71], [73].

2.3.1. Gain Scheduling

The basic idea associated with *gain scheduling* is to change the parameters of the controller based on the changes of certain variables, called *scheduling variables*. Parameter changes are well correlated with the changes on the dynamics of the process, [70]. In general terms, the design of a gain scheduling controller consists of the following steps, [72]:

- Linear Parameter-Varying Model Generation: The most common approach is based on Jacobian linearization of the nonlinear plant about a family of equilibrium points. This yields a parameterized family of linearized plants and forms the basis for what is termed *linearization scheduling*.
- Design of the linear controller set: A linear controller is designed for each linearized plant model, which constitutes the linear parameter-varying model. This step results in a family of linear controllers
- Implementation of the Gain Scheduling logistic block: Since only a selected number of equilibrium points are linearized, it is necessary to establish a procedure to change the controller's parameters when the scheduling variables change. The use of thresholds represents the most basic approach. However, thresholds could produce "jumps" in some variables of interest. In order to avoid jumps typical approaches incorporate interpolation of controller's parameters and blending.

- Performance Assessment: Typically, the local stability and the performance properties of the gain scheduled controller are subject to analytical investigation, while nonlocal performance evaluation is based on simulation studies, [72].

The main disadvantage of gain scheduling is that possible future changes in the system's parameters are not taken into account. A simplified block diagram of a gain scheduling controller is presented in Figure 4.

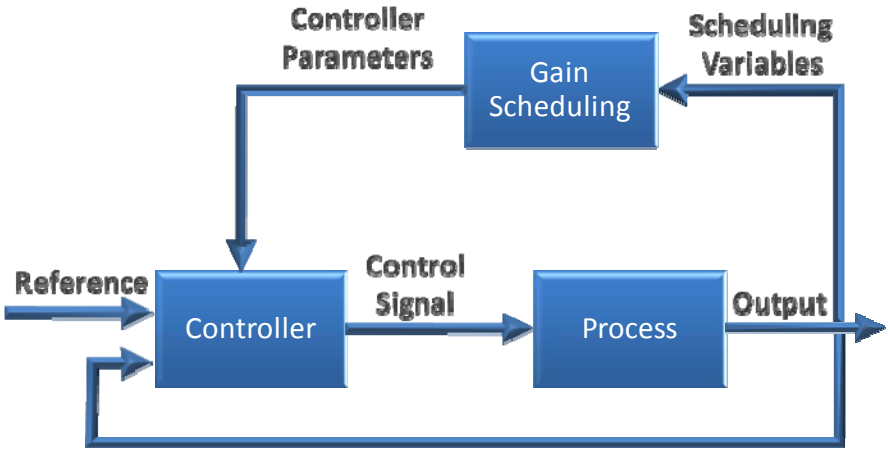


Figure 4: A Simplified Block Diagram of a Gain Scheduling Controller

2.3.2. Model-Reference Adaptive Control

In this type of adaptive controller, a model is used to generate a reference signal. This signal shows the adaptive controller how the closed-loop system should respond to input commands, [70]. The controller parameters are adjusted in such a way that the difference between the process output and the reference signal is kept small. In Model-

Reference Adaptive Systems, (MRAS), the main issue is to determine the adjustment mechanism so that a stable system, which brings the error to zero, is obtained [70].

A simplified block diagram of a Model Reference Adaptive System is presented in Figure 5.

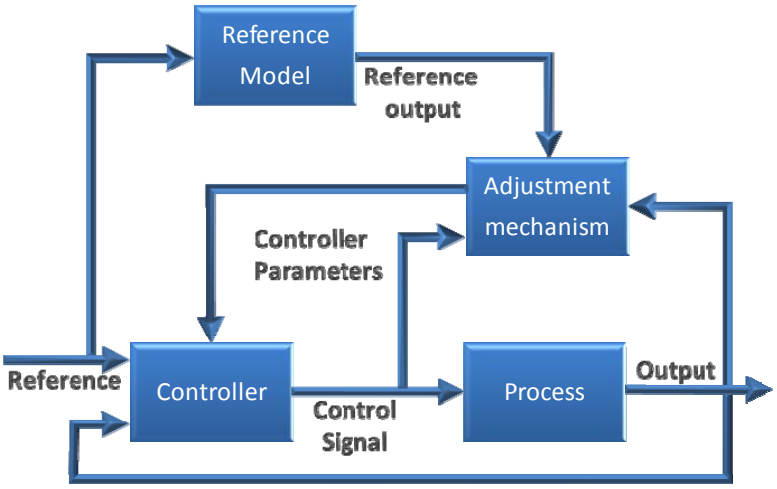


Figure 5: A Simplified Block Diagram of a Model Reference Adaptive System

2.3.3. Self-Tuning Regulators

In self-tuning regulators, (STR), estimates of the process parameters are obtained and then used to obtain the controller parameters using a controller design method based on the updated/estimated parameters, [70], [71]. A simplified block diagram of a Self-Tuning Regulator is presented in Figure 6.

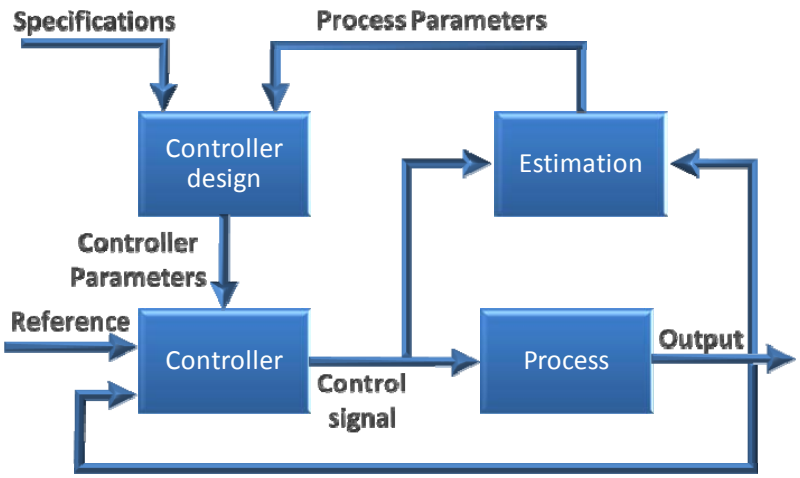


Figure 6: A Simplified Block Diagram of a Self-Tuning Regulator System

2.3.4. Adaptive Dual Control

The previous adaptive control approaches were based on reasonable heuristics. They were based on separation of the parameter estimation and controller designs, [70]. Adaptive Dual Control is an approach, which was derived from an abstract problem formulation, used in optimization theory. The method was originally proposed by A. Feldbaum, (1960-61, 1965), [73]. In his early work, Feldbaum indicated that systems based on the *certainty equivalence* (CE) principle are not always optimal. In fact, CE based systems can be far from optimal, [73]. Feldbaum postulated two main properties, which the control signal of an optimal adaptive control system should possess. It should ensure that the system output cautiously tracks the desired reference value and it should excite the plant sufficiently for accelerating the parameter estimation process so that the control quality improves in future time intervals.

The formal solution of the original approach of adaptive dual control, as proposed by A. Feldbaum, can be realized using dynamic programming. However, the equation is considered to be practically unsolvable and only a few very simple control problems have been solved, [73], [70]. A simplified block diagram of an Adaptive Dual Control System is presented in Figure 7.

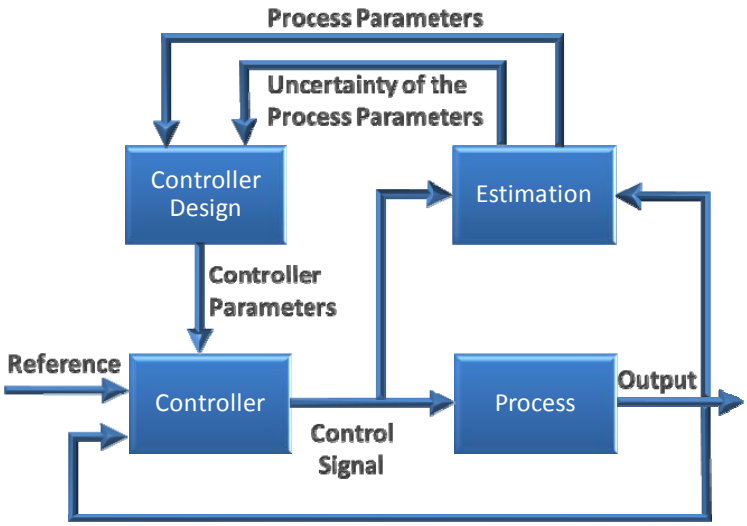


Figure 7: A Simplified Block Diagram of an Adaptive Dual Control System

2.4. Model Predictive Control

Model Predictive Control, (MPC), Model-Based Predictive Control, (MBPC), or simply Predictive Control, (PC), was developed and used in the industry for nearly twenty years before attracting very much serious attention from the academic control community, [74].

Maciejowski stated that predictive control was proposed or devised independently for several people more or less simultaneously, [74]. This technique was used for years in the industry before it was presented or published in papers. Therefore, it is difficult to determine who was first to propose the original predictive control approach. Richalet et al, [75],in 1978 at ADERSA, published their Model Predictive Heuristic Control, (MPHC), which was later known as Model Algorithmic Control, (MAC), [78]. MPHC software is termed Identification and Command, (IDCOM). Cutler and Ramaker, [76], published their predicted control called Dynamic Matrix Control, (DMC), in 1980. Interestingly, Juan Martin Sanchez, [77], holds the earliest patent for a control technique with the characteristics of the current standard predictive control. The US patent is titled Adaptive-Predictive Control System.

MPC refers to a set of control strategies based on the same basic ideas or concepts, [8], which are:

- The explicit use of a plant model to predict the behavior, in terms of states and outputs, of the plant at future time instants.
- The computation of a control sequence for minimizing a cost or objective function, which takes into account the output/states errors and control effort.
- The receding horizon strategy where the predicted behavior at each instant is displaced towards the future and only the first value of the calculated control sequence at each instant is applied.

MPC possesses the ability to naturally handle many situations, which other control techniques are not able to handle. Therefore, MPC may be considered as the most general approach capable of addressing a control problem in the time domain. Some of the main advantages for using/implementing MPC, also called Receding Horizon Predictive Control, (RHPC), [78], [74], are:

- Ability to support constraints of variables associated with the control problem under study such as input, output or states variables,
- Its basic formulation may be extended to multivariable plants with almost no modification,
- Intrinsic compensation for dead time and no minimum phase dynamics,
- Deals with zone objectives,
- Deals naturally with non-square plants,
- Possesses the ability to use future values of references when they are available. This capability allows MPC to improve performance in navigation such as waypoint trajectory tracking.

The basic ideas upon which MPC is based are quite general. They can, in principle, be applied to any plant for which it is possible to develop a model. In addition, MPC provides for simulation of the model at a speed faster than real-time and minimization of the cost function at speed faster than real-time. The basic structure of Model Predictive Control, [79], is presented in Figure 8.

Figure 9 displays signals involved in Model Predictive Control for a Single-Input Single-Output system assuming a discrete-time control approach.

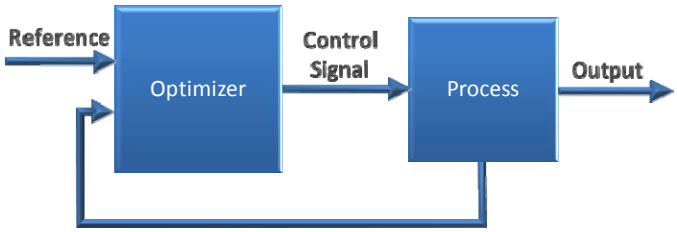


Figure 8: Basic Structure for Model Predictive Control

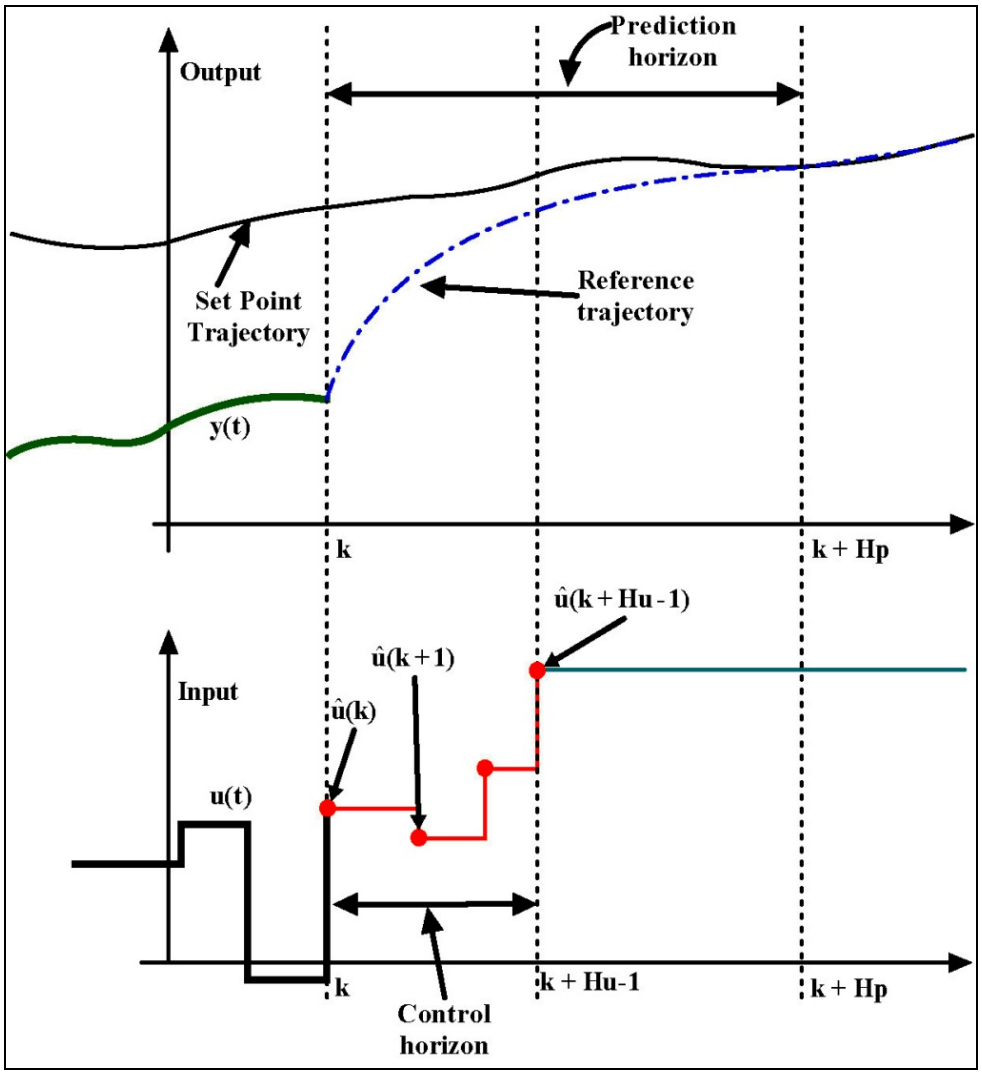


Figure 9: Model Predictive Signals

The set point trajectory, s(t), is the trajectory that the output, y(t), should follow. The reference trajectory, r(t), is the trajectory that starts at the current output, y(k), and defines an ideal trajectory along which the output should return, after a possible disturbance, to the set point trajectory, [74]. This reference trajectory is normally an exponential function. However, it could be any other function or it could be the same set point trajectory.

The Prediction horizon, H_p, is the number of sampling intervals, which the internal model will be simulated to predict the behavior of the plant. The internal model will be simulated from the initial time

$$t_{initial} = k * T_{sampling}$$

to the final time

$$t_{final} = (k + H_p) * T_{sampling}$$

It is important to observe that the simulation of the internal model will depend on the assumed input trajectory. The assumed input trajectory,

$$\{\hat{u}(k), \hat{u}(k+1), \ldots, \hat{u}(k+H_p-1)\}$$

is the trajectory, which the controller should attain through optimization of the cost function.

The Control horizon, H_u, is the number of control signal values,

$$\{\hat{u}(k), \hat{u}(k+1), \ldots, \hat{u}(k+H_u-1)\},$$

of the input trajectory, which will be considered as variables and will be obtained from the optimization step. Considering

$$H_u \leq H_p,$$

for

$$k > H_u - 1$$

the control signal values will be

$$\hat{u}(k + H_u - 1) = \hat{u}(k + H_u + 1) = \ldots = \hat{u}(k + H_p - 2) = \hat{u}(k + H_p - 1).$$

It is strange to assume $H_u > H_p$. However, it can be reasonable under certain conditions. For instance, it is reasonable under the condition that the control signal values for $k > H_p - 1$ are all equals to $\hat{u}(k + H_p - 1)$ [74].

2.5. Fault-Tolerant Control

Safety and reliability are very important aspects of current complex technological systems. Control systems used to improve the overall performance of commercial, industrial and military processes are composed of sophisticated digital system design techniques and complex hardware such as input-output sensors, actuators, components and processing units, [80].

Specific terminology is needed to understand the concepts and ideas related with Fault-Tolerant Control Systems. Some terms are presented based on the information obtained from the SAFEPROCESS Technical Committee. They are considered, "on-going", in the sense that new definitions and updates are being formulated, [80]:

- Fault is "an unpermitted deviation of at least one characteristic property or parameter of the system from the acceptable, usual or standard condition", [80].

- Failure is “a permanent interruption of a system’s ability to perform a required function under specified operating conditions”, [80].
- Malfunction is “an intermittent irregularity in the fulfillment of a system’s desired function”, [80].
- Fault detection is the “determination of faults present in a system and the time of detection”, [80].
- Fault isolation is the “determination of the kind, location and time of detection of a fault. Follows fault detection”, [80].
- Fault identification is the “determination of the size and time-variant behavior of a fault. Follows fault isolation”, [80].
- Fault diagnosis is “the kind, size, location and time of detection of a fault. Follows fault detection. Includes fault detection and identification”, [80].

The Architecture of Fault-Tolerant Control Systems is presented in Figure 10 [81]. The architecture is composed of the fault diagnosis block and the control re-design block. The fault diagnosis block uses the measured input and output and tests their consistency with the plant model. The control re-design block uses the fault information and adjusts the controller to the faulty situation.

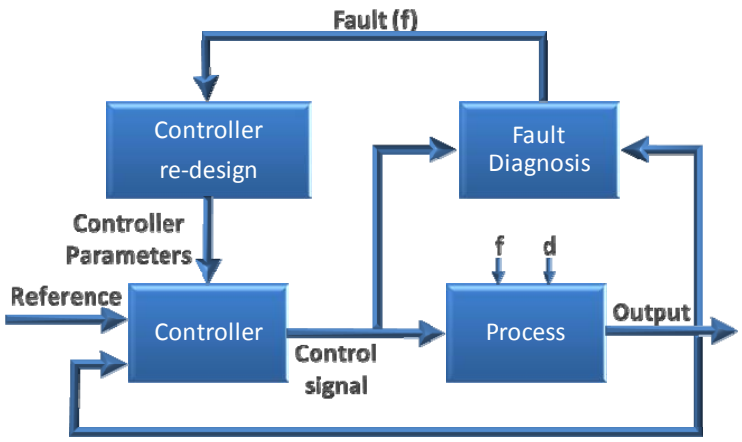


Figure 10: Basic Block Diagram of a Fault-Tolerant Control System

Patton, [82], and Zhang & Jiang, [83], classify Fault-Tolerant Control System into two major groups. The groups are the passive fault-tolerant control systems, (PFTCS), and active fault-tolerant control systems, (AFTCS). Figure 11 presents a diagram, which represents these classifications.

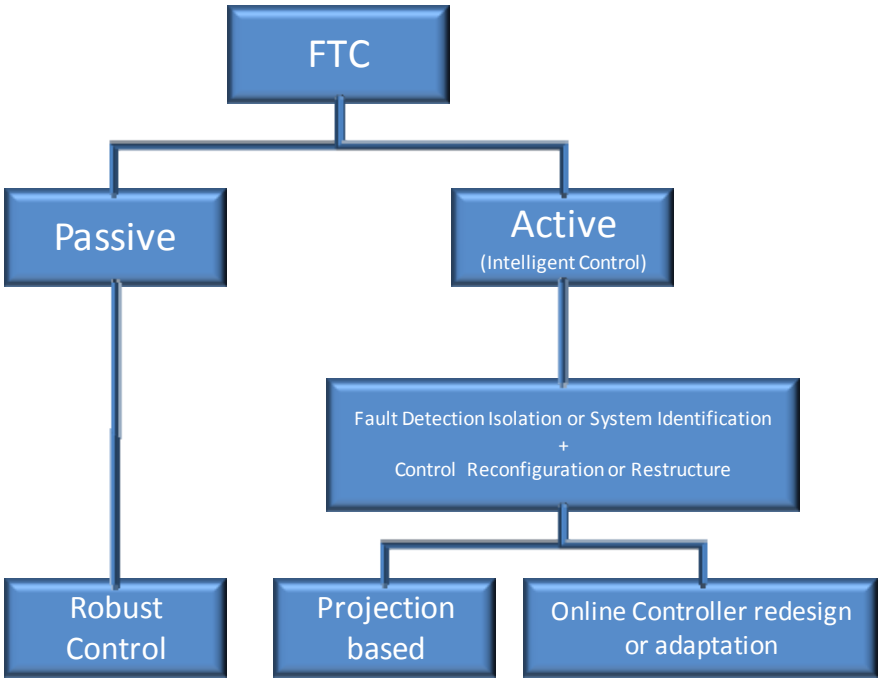


Figure 11: Classification of Fault-Tolerant Control Systems

2.5.1. Types and Modeling of Faults and Failures

In general, faults can be classified as actuator faults, sensor faults and system faults. Figure 12 presents a diagram of these fault classifications.

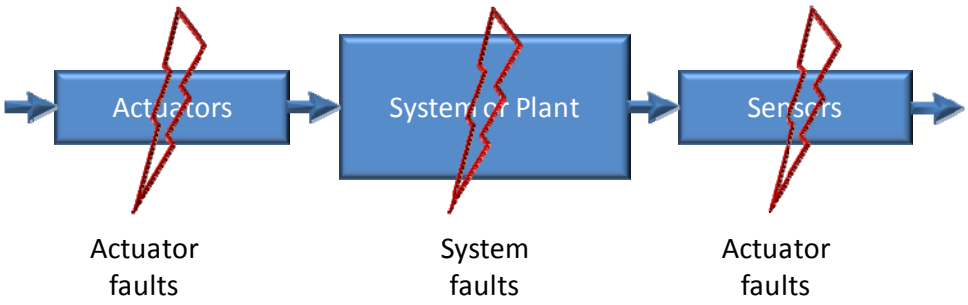


Figure 12: Types of Faults and Failures

"Sensor faults break the information link between the plant and the controller. These faults can render the plant partially unobservable. New measurements may have to be selected and used in order to solve the control task. Actuator faults disturb the possibilities to influence the plant. These faults can make the plant partially uncontrollable. New actuators may have to be used. Plant faults change the dynamic behavior of the process. Since any control law cannot tolerate severe changes, a redesign of the controller is necessary, [84].

Assuming that the whole system can be modeled as a typical state space linear system, it can be represented by:

$$\begin{aligned} \dot{\mathbf{x}}(t) &= \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) \\ \mathbf{y}(t) &= \mathbf{C}x(t) \end{aligned}, \tag{1}$$

with $\mathbf{x}(t) \in \mathbb{R}^n$, $\mathbf{u}(t) \in \mathbb{R}^m$, $\mathbf{y}(t) \in \mathbb{R}^l$, $\mathbf{A} \in \mathbb{R}^{n \times n}$, $\mathbf{B} \in \mathbb{R}^{n \times m}$, and $\mathbf{C} \in \mathbb{R}^{l \times n}$. The parameter *n* is the number of states, *m* is the number of inputs and *l* is the number of outputs.

An actuator fault is normally represented in the literature as a decrease in the actuator's effectiveness, which is represented by:

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) - \mathbf{B}\mathbf{K}\mathbf{u}(t), \tag{2}$$

with

$$\mathbf{K} = diag(k_1, \ldots k_m), \tag{2a}$$

where the k_i are scalars satisfying $0 \le k_i \le 1$, [85]. The k_i scalars model a reduction in the effectiveness, (gain), of the *i*th actuators. If $k_i = 0$, then the *i*th actuator functions

normally. The ith actuator presents a fault if $k_i > 0$. If $k_i = 1$, the ith actuator presents a failure.

Another type of fault, which occurs in aircraft, is structural damage. Structural damage may change the operating conditions of the aircraft from its nominal conditions due to changes in the aerodynamic coefficients of the aircraft or a change in the center of gravity. Therefore, in terms of the linear model, the **A** matrix will also be perturbed. Mathematically, this can be represented by, [85]:

$$\dot{\mathbf{x}}(t) = (\mathbf{A} + \Delta\mathbf{A})\dot{\mathbf{x}}(t) + (\mathbf{B} + \Delta\mathbf{B})\mathbf{u}(t) + \xi(\mathbf{x}, \mathbf{u}, t), \tag{3}$$

where $\Delta\mathbf{A}$ and $\Delta\mathbf{B}$ represent the changes in the **A** and **B** matrices and $\xi(\mathbf{x}, \mathbf{u}, t) \in \mathbb{R}^n$ represents additional changes ,which are not included in $\Delta\mathbf{A}$ and $\Delta\mathbf{B}$, [85].

Boskovic and Mehra, [86], describe some typical actuator failures:

- Lock-In-Place, (LIP),
- Hard-Over Failure, (HOF),
- Float,
- Loss of Effectiveness, (LOE).

LIP is a failure condition, which occurs when the actuator becomes stuck and immovable. The actuator moving to the upper or lower position limit at its maximum rate limit, without responding to commands, characterizes HOF. LOE is a decrease of the actuator gain.

Typical sensor failures are, [86]:

- Bias is a constant offset/error between the actual and measured signals;
- Drift occurs when the measurement errors increase over time;

- Performance degradation, (loss of accuracy), occurs when the measurements never indicate the true values of the signals;
- Freezing occurs when a sensor provide a constant value instead of the true value;
- Calibration error,(loss of effectiveness), is a gain error of the sensor.

2.5.2. Fault Detection Methods

Typical methods used for the detection of sensors, plant and actuators faults or failures are, [80]:

- Observer,
- Parity Space,
- Parameter Estimation,
- Frequency spectral analysis,
- Neural networks.

Based on some statistic provided by, [80], it can be stated that parameter estimation and observer-based methods are the most frequently applied techniques for fault detection. In addition, it is mentioned that more than 50% of sensor faults are detected using observer-based methods while the other methods play a less important role. For the detection of actuator faults, observer-based methods are mostly used, followed by parameter estimation and neural networks. The detection of process faults is performed mostly by parameter estimation methods.

2.6. Summary

It was observed, from the UAV low-level control literature review, that adaptive control approaches have been used to increase the reliability of the UAV, which "is the single most immediate and long -reaching need to ensure their success", [3]. Table 1 and Table 2 data, force the conclusion that MPC has a high potential for use in the development of fault-tolerant control systems.

The importance of the observer-based and the parameter estimation methods for the detection of sensors, process and actuators faults should be clear. It is important to consider that most practical processes need the use of an observer to estimate the states signals.

The following chapters will cover the issues mentioned in this summary. Specifically, chapter 3 will cover states observers or estimators and parameter estimation. Chapter 4 will cover adaptive and fault-tolerant predictive control.

Chapter 3

Estimation

3.1. Estimation Theory

The need to extract or estimate useful information, from noisy signals or from partial information sources, is almost pervasive in most of the real-world signal processing and control systems. Estimating the values of signals or parameters is a fundamental part of many signal-processing systems. In the particular case of control systems, the requirement is pervasive to use an algorithm to obtain measured outputs and the estimated values of the state variables of the process from noise. The Kalman filters are the most commonly used algorithm for the purposes of extracting information from noise. A brief background of the most common types of Kalman filter will be presented.

3.2. Standard Kalman Filter

The Kalman filter is an estimator for what is called the linear-quadratic problem. This is the problem of estimating the instantaneous "state" of a linear dynamical system, which has been perturbed by additive white Gaussian noise with normal distribution, using measurements linearly related to the state and perturbed by additive white Gaussian noise with normal distribution.

In his original formulation, [87], Kalman addressed the general problem of estimating the state, $\mathbf{x} \in \mathcal{R}^n$, of a discrete-time process whose dynamics are described by the linear stochastic difference equation, which is given by:

$$\mathbf{x}_{k+1} = \mathbf{F}\mathbf{x}_k + \mathbf{G}\mathbf{u}_k + \mathbf{w}_k \tag{4}$$

$$\mathbf{z}_k = \mathbf{H}\mathbf{x}_k + \mathbf{v}_k \tag{5}$$

where **F** is the state transition matrix, also termed the system or dynamic matrix, **G** is the input matrix, $\mathbf{u}_k$ is the input vector, $\mathbf{w}_k$ is the process noise vector and $\mathbf{v}_k$ is the observation or measurement noise vector. The process noise vector is white Gaussian with zero mean and covariance matrix given by:

$$E[\mathbf{w}_k \mathbf{w}_k^T] = \begin{cases} \mathbf{Q}_k & n = k \\ \mathbf{0} & n \neq k \end{cases} \tag{6}$$

The observation or measurement noise is white Gaussian with zero mean and covariance matrix given by:

$$E[\mathbf{v}_k \mathbf{v}_k^T] = \begin{cases} \mathbf{R}_k & n = k \\ \mathbf{0} & n \neq k \end{cases} \tag{7}$$

In real situations, the process noise covariance and measurement noise covariance matrices might change with each time step or measurement. However, it is assumed that they are constant. The process noise and the measurement noise are uncorrelated, which requires:

$$E[\mathbf{w}_k \mathbf{v}_k^T] = 0 \tag{8}$$

To present the equations, which allow the implementation of the Kalman filter, some definition and nomenclature must be defined. Let $\hat{\mathbf{x}}_k^- \in R^n$ represent the *a priori*

state estimate at step k given knowledge of the process prior to the step k. Let $\hat{\mathbf{x}}_k \in R^n$ represent the *a posteriori* state estimate at step k given the measurement $\mathbf{z}_k$. Given these representations, prior and posteriori estimates can be defined as:

$$\mathbf{e}_k^- \equiv \mathbf{x}_k - \hat{\mathbf{x}}_k^- \tag{9}$$

$$\mathbf{e}_k \equiv \mathbf{x}_k - \hat{\mathbf{x}}_k \tag{10}$$

The *a priori* estimate error covariance is given by:

$$\mathbf{P}_k^- = E[\mathbf{e}_k^- \mathbf{e}_k^{-T}] \tag{11}$$

and the *a posteriori* estimate error covariance is given by

$$\mathbf{P}_k = E[\mathbf{e}_k \mathbf{e}_k^{-T}] \tag{12}$$

The Kalman filter obtains the *a posteriori* state estimate, $\hat{\mathbf{x}}_k$ as a linear combination of the *a priori* estimate $\hat{\mathbf{x}}_k^-$ and a weighted difference between the measurement $\mathbf{z}_k$ and the measurement prediction $\mathbf{H}\hat{\mathbf{x}}_k^-$ at step k. This *a posterior* state estimate is given by:

$$\hat{\mathbf{x}} = \hat{\mathbf{x}}_k^- + \mathbf{K}_k(\mathbf{z}_k - \mathbf{H}\mathbf{x}_k^-) \tag{13}$$

The difference, $(\mathbf{z}_k - \mathbf{H}\mathbf{x}_k^-)$ in equation (13) , is called the measurement innovation or the residual.

The $n \times m$ matrix $\mathbf{K}$,in equation (13), is termed the Kalman gain or blending factor, [88]. It is chosen to minimize the *a posteriori* error covariance, which is presented in equation (12). One form that minimizes the error covariance is given by:

$$\begin{aligned}\mathbf{K}_k &= \mathbf{P}_k^- \mathbf{H}^T (\mathbf{H}\mathbf{P}_k^- \mathbf{H}^T + \mathbf{R})^{-1} \\ &= \frac{\mathbf{P}_k^- \mathbf{H}^T}{\mathbf{H}\mathbf{P}_k^- \mathbf{H}^T + \mathbf{R}}\end{aligned} \tag{14}$$

The Kalman filter is a recursive algorithm whose equations can be separated into groups concerned with time update or prediction equations and measurement update or correction equations. The time update or predictor equations are used for projecting the current state and error covariance estimates forward to obtain the *a priori* estimates for the next time step. The measurement update or corrector equations are responsible for incorporating a new measurement into the *a priori* estimate to obtain an improved *a posteriori* estimate, [88].

Table 3 presents the Kalman filter equations in a sequential approach, which indicates the calculations required for the appropriate implementation of the filter.

Table 3: Kalman Filter Algorithm

t	Operation
k-1	Obtain the measurements at $t = k$ -1 , z_{k-1}
	$\hat{\mathbf{x}}_{k-1} = \hat{\mathbf{x}}_{k-1}^{-} + \mathbf{K}_{k-1}(z_{k-1} - \mathbf{H}_{k-1}\hat{\mathbf{x}}_{k-1}^{1})$
	Calculations of $\mathbf{u}_{k-1}$
	$\hat{\mathbf{x}}_{k}^{-} = \mathbf{F}\hat{\mathbf{x}}_{k-1} + \mathbf{B}\mathbf{u}_{k-1} + \mathbf{w}_{k-1}$
	$\mathbf{P}_{k}^{-} = \mathbf{F}\mathbf{P}_{k-1}^{+}\mathbf{F}^{T} + \mathbf{Q}_{k}$
	$\mathbf{K}_{k} = \mathbf{P}_{k}^{-}\mathbf{H}^{T}(\mathbf{H}\mathbf{P}_{k}^{-}\mathbf{H}^{T} + \mathbf{R}_{k})^{-1}$
k	Obtain the measurements at $t = k$, z_{k}
	$\hat{\mathbf{x}}_{k} = \hat{\mathbf{x}}_{k}^{-} + \mathbf{K}_{k}(\mathbf{z}_{k} - \mathbf{H}_{k}\hat{\mathbf{x}}_{l}^{-})$
	$\mathbf{P}_{k}^{+} = (\mathbf{I} - \mathbf{K}_{k}\mathbf{H})\mathbf{P}_{k}^{-}(\mathbf{I} - K_{k}\mathbf{H})^{T} + \mathbf{K}_{k}\mathbf{R}_{k}\mathbf{K}_{k}^{T}$
	Calculation of $\mathbf{u}_{k}$
	⋮
k+1	Obtain the measurements at $t = k$ +1, z_{k+1}

Having presented the fundamentals concepts and the equations needed for the implementation of the Kalman Filter, the next sections present some of the extensions that were of interest for this research.

3.3. Extended Kalman Filter

The Extended Kalman Filter, (EKF), was the first extension and at the same time, the first application of the Kalman Filter. The EKF is probably the most widely used

estimator for nonlinear systems. However, the practical use of the EKF has two well-known drawbacks, [89]:

- Linearization can produce a highly unstable filter if the assumptions of local linearity are violated;
- The derivation of the Jacobian matrices is nontrivial in most applications and often lead to significant difficulties.

To understand the causes of the problems obtained with the application of the EKF to nonlinear dynamical systems, some concepts need to be investigated.

Consider the equations of a stochastic time-invariant or autonomous nonlinear dynamical system, which are given by:

$$\mathbf{x}_{k+1} = F(\mathbf{x}_k, \mathbf{u}_k, \mathbf{w}_k) \tag{15}$$

and

$$\mathbf{z}_k = H(\mathbf{x}_k, \mathbf{v}_k) \tag{16}$$

where $\mathbf{x}_k$ is the state vector, $\mathbf{u}_k$ is the input vector, $\mathbf{w}_k$ is the process noise and $\mathbf{v}_k$ is the measurement noise. The process noise and the measurement noise do not need to be considered as additive. The nonlinearity presented in the system results due to the presence of F, which is a nonlinear function, the presence of H, which is a nonlinear function or the presence of both functions, which are nonlinear functions.

Given the noisy measurements $\mathbf{z}_k$, the recursive estimation of $\hat{\mathbf{x}}_k$ can be obtained using the equation (13). If the *a priori* estimate $\hat{\mathbf{x}}_k^-$ and the current measurement or observation are Gaussian, [90], then the recursion provides the optimal Minimum Mean-

Square Error estimate of **x**. The optimal values terms of the recursive equation (11) are given by, [90] [91]

$$\hat{\mathbf{x}}_k^- = E[F(\hat{\mathbf{x}}_{k-1}, \mathbf{u}_{k-1}, \mathbf{w}_{k-1})] \tag{17}$$

$$\hat{\mathbf{z}}_k^- = E[H(\hat{\mathbf{x}}_k^-, \mathbf{v}_k)] \tag{18}$$

and

$$\mathbf{K}_k = \mathbf{P}_{\mathbf{x}_k \mathbf{z}_k} \mathbf{P}_{\tilde{\mathbf{z}}_k \tilde{\mathbf{z}}_k}^{-1} = E[(\mathbf{x}_k - \hat{\mathbf{x}}_k^-)(\mathbf{z}_k - \hat{\mathbf{z}}_k^-)^T] \times E[(\mathbf{z}_k - \hat{\mathbf{z}}_k^-)(\mathbf{z}_k - \hat{\mathbf{z}}_k^-)^T]^{-1} \tag{19}$$

where the optimal prediction $\hat{\mathbf{x}}_k^-$ is the expectation of a nonlinear function of $\hat{\mathbf{x}}_{k-1}$, $\mathbf{u}_{k-1}$ and $\mathbf{w}_{k-1}$, which are random variables. The same applies to the optimal prediction of $\hat{\mathbf{z}}_k^-$. The Kalman gain is expressed as a function of posterior covariance matrices in which

$$\tilde{\mathbf{z}}_k = \tilde{\mathbf{z}}_k - \tilde{\mathbf{z}}_k^-$$

In all these terms, it is necessary to calculate expectations of nonlinear functions in order to obtain the optimal values. It is well-known that the optimal solution of the nonlinear filtering problem requires that a complete description of the conditional probability density be maintained. Unfortunately, the exact description requires a potentially unbounded number of parameters. Therefore, a number of suboptimal approximations have been proposed, [89]. The EKF is a suboptimal approximation that obtains the terms of equation (13) using the following simplifications:

$$\hat{\mathbf{x}}_k^- \approx F(\hat{\mathbf{x}}_{k-1}, \mathbf{u}_{k-1}, \bar{\mathbf{w}}) \tag{20}$$

$$\mathbf{K}_k \approx \hat{\mathbf{P}}_{\mathbf{x}_k \mathbf{z}_k} \hat{\mathbf{P}}_{\tilde{\mathbf{z}}_k \tilde{\mathbf{z}}_k}^{-1} \tag{21}$$

and

$$\hat{\mathbf{z}}_k^- \approx H(\hat{\mathbf{x}}_k^-, \bar{\mathbf{v}}) \tag{22}$$

where the prediction of $\hat{\mathbf{x}}_k^-$ and $\hat{\mathbf{z}}_k^-$ are approximated by directly evaluating the nonlinear function *F* and *H* with the prior mean values. The covariances are determined by linearizing the dynamical nonlinear equations of the system and then analytically determining the posterior covariance matrices for the linear system as in the case of the standard Kalman filter. Having these values the *a posteriori* estimate of $\hat{\mathbf{x}}_k$ can be obtained from equation (13).

The values obtained with these approximated equations can be considered as "first order" approximations of the optimal values. When the nonlinear system is not well represented by the linearization, the calculated *a posteriori* estimates of the mean and covariance matrix will have large errors. These errors could produce severe suboptimal performance and possibly lead to divergence of the filter.

All these issues have led researchers, to seek more accurate methods for the solution of the problem of filtering nonlinear dynamical systems. Even though, there are EKF variants, which are more accurate, they are more complex and computationally demanding.

3.4. Unscented Kalman Filter

The Unscented or Sigma-Point Kalman Filter developed by Julier and Uhlmann, [89], was introduced as a solution to the problems of the EKF. The propagation of a Gaussian random variable, (GRV), through the system dynamic is a central and vital operation upon which all Kalman filters are based. The approach presented by Julier and

Uhlmann is a new method of calculating the first and second order statistics of a random variable, which undergoes a nonlinear transformation. The Unscented or sigma-Point Kalman filter is a direct application of the transformation with the similar name Unscented Transformation.

3.4.1. The Unscented Transformation

The Unscented Transformation, (UT), is a new, novel method for calculating the statistics of a random variable, which undergoes a nonlinear transformation, [89]. Julier and Uhlmann founded their work with the intuition that "with a fixed number of parameters it should be easier to approximate a Gaussian distribution than it is to approximate an arbitrary nonlinear function/transformation", [92]. Figure 13 depicts that sigma points capturing the mean and covariance of the distribution are transformed independently. The mean and covariance of the transformed sigma points define the statistics of the transformed random variable

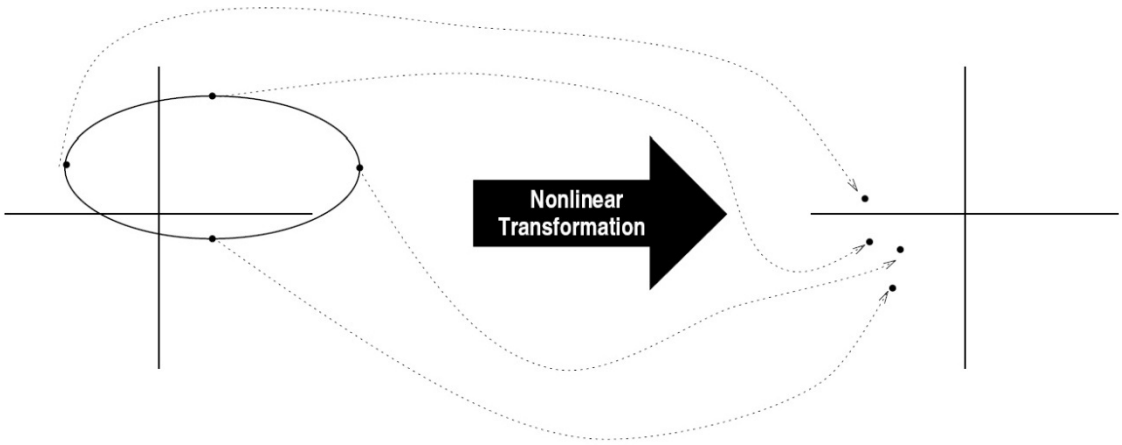


Figure 13: Principle of the Unscented Transformation [92]

The general problem of calculating the statistics of a random variable, which undergoes a nonlinear transformation, is governed by a relatively complex algorithm.

Given an n-dimensional vector random variable x with mean $\bar{}$x and covariance $\mathbf{P}_{xx}$, obtain the mean $\bar{\mathbf{y}}$ and the covariance $\mathbf{P}_{yy}$ of the vector random variable **y,** which is related to **x** by the nonlinear transformation

$$\mathbf{y} = \mathbf{g}[\mathbf{x}] \tag{23}$$

The Sigma-Point method follows, [92]. Compute the set **σ** of 2*n* points from the rows or columns of the matrices $\pm\sqrt{n\mathbf{P}}$. This set is zero mean with covariance **P**. Compute a set of points with the same covariance but with mean $\bar{\mathbf{x}}$, by translating each point as:

$$\boldsymbol{\chi} = \boldsymbol{\sigma} + \bar{\mathbf{x}}$$

where

$$\sigma \leftarrow 2n \text{ rows or columns from } \pm\sqrt{(n+k)\mathbf{P}_{xx}}$$

k will be defined later and

$$\boldsymbol{\chi}_0 = \hat{\mathbf{x}}$$
$$\boldsymbol{\chi}_i = \sigma_i + \hat{\mathbf{x}}$$

which assures that

$$\mathbf{P}_{xx} = \frac{1}{2(n+k)} \sum_{i=1}^{2n} [\boldsymbol{\chi}_i - \hat{\mathbf{x}}][\boldsymbol{\chi}_i - \hat{\mathbf{x}}]^T$$

Transform the set of sigma point by

$$\boldsymbol{\gamma}_i = \mathbf{g}[\boldsymbol{\chi}_i] \tag{24}$$

The approximated mean is computer by:

$$\hat{\mathbf{y}} = \frac{1}{n+k}\left\{k\boldsymbol{\gamma}_0 + \frac{1}{2}\sum_{i=1}^{2n}\boldsymbol{\chi}_i\right\} \tag{25}$$

and the approximated covariance is computed by:

$$\mathbf{P}_{yy} = \frac{1}{n+k}\left\{k[\boldsymbol{\gamma}_0 - \hat{\mathbf{y}}][\boldsymbol{\gamma}_0 - \hat{\mathbf{y}}]^T + \frac{1}{2}\sum_{i=1}^{2n}[\boldsymbol{\gamma}_i - \hat{\mathbf{y}}][\boldsymbol{\gamma}_i - \hat{\mathbf{y}}]^T\right\} \tag{26}$$

The properties of this algorithm were summarized by Julier and Uhlmann, [89]. Since the mean and covariance of x are captured precisely up to the second order, the calculated values of the mean and covariance of y are also correct to the second order. The sigma points capture the same mean and covariance irrespective of the choice of matrix square root, which is used. Numerically efficient and stable methods such as the Cholesky decomposition can be used. The mean and covariance are calculated using standard vector and matrix operations. This means that the algorithm is suitable for any choice of process model and implementation is extremely rapid since it is not necessary to evaluate the Jacobian, which is required by an EKF.

The parameter k provides an extra degree of freedom to "fine tune" the higher order moments of the approximations, and can be used to reduce the overall prediction errors. When $\mathbf{x}_k$ is assumed to be Gaussian, a useful heuristic is to select $n+k = 3$. If a different distribution is assumed for $\mathbf{x}_k$ then a different choice of k might be more appropriate. Although k can be positive or negative, a negative choice of k can lead to a non-positive semi-definite estimate of $\mathbf{P}_{yy}$. Figure 14 presents the different cases of propagating the statistics of a 2D random variable through a nonlinear transformation.

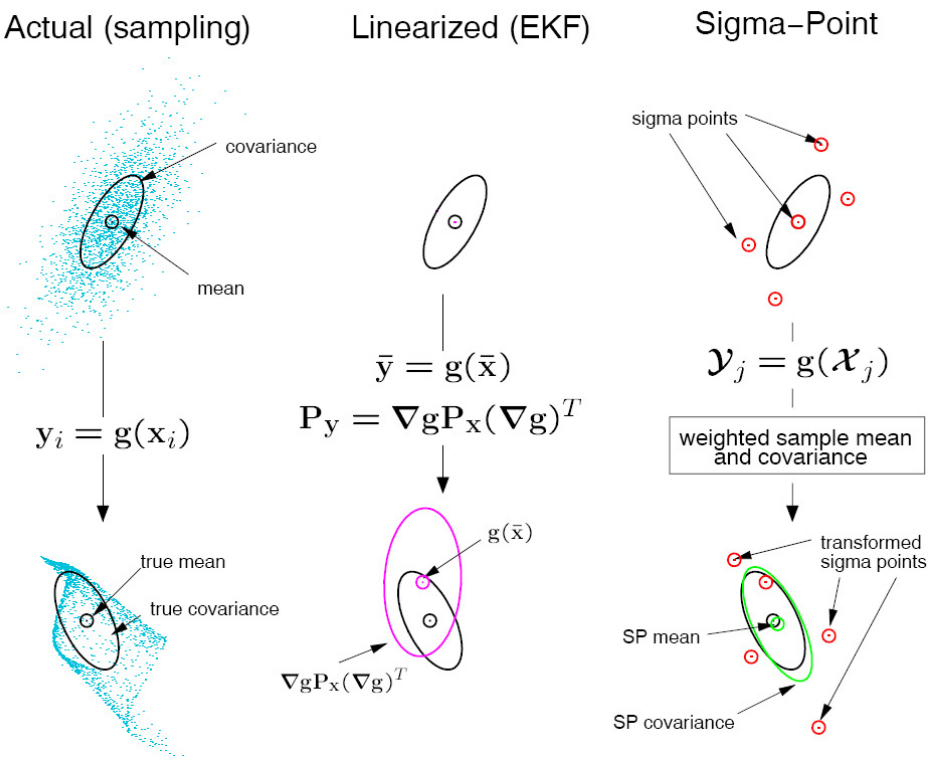


Figure 14: 2D Example of the Sigma-Point or Unscented Approach [97]

The Unscented Filter is a straightforward extension of the UT to the recursive estimation where $\boldsymbol{\sigma} = \mathbf{x}_k$ and the corresponding matrix is represented as $\boldsymbol{\chi}(k|k)$. It is interesting to note that no explicit calculation of Jacobians or Hessians is necessary to implement this algorithm.

3.5. Dual Estimation

The problem of Dual Estimation consists on the simultaneous estimation of the states and the parameters' model of the dynamical system from which the measurements or observations are taken. Considering that the dynamical system is expressed by:

$$\mathbf{x}_{k+1} = F(\mathbf{x}_k, \mathbf{u}_k, \mathbf{w}_k, \boldsymbol{\theta}) \tag{27}$$

and

$$\mathbf{z}_k = H(\mathbf{x}_k, \mathbf{v}_k, \boldsymbol{\theta}) \tag{28}$$

where $\mathbf{x}_k$ is the state vector, $\mathbf{u}_k$ is the input vector, $\boldsymbol{\theta}$ is the parameter's model vector, $\mathbf{w}_k$ is the process noise and $\mathbf{v}_k$ is the measurement noise. The process noise and the measurement noise do not need to be considered as additive. The system will be linear or nonlinear in the states as a function of the linearity or nonlinearity of the system function F and measurement function H. If either of these functions is nonlinear, the estimation will become a nonlinear estimation problem. However, even if the functions F and H are linear or affine in the states and inputs, the estimation problem becomes nonlinear when the simultaneous estimation of the parameters is considered.

Most common algorithms used to solve the dual estimation problems are:

- Expectation Maximization, (EM),
- Dual Kalman Filter,
- Joint Kalman Filter.

The EM algorithm uses an extended Kalman smoother for the E-step where forward and backward passes are made through the data to estimate the signal. The model is updated during a separate M-step, [90].

The Dual Kalman Filter algorithm uses two separate Kalman filters. One filter is used for estimating the states given the current parameters and one filter is used for parameters' model estimation given the current states. To estimate the parameter's model vector using the dual Kalman filter or the joint Kalman filter it is necessary to

represent them as a stationary process, with an identity state-transition matrix, which is driven by process noise $\mathbf{r}_k$:

$$\boldsymbol{\theta}_k = \boldsymbol{\theta}_{k-1} + \mathbf{r}_k \tag{29}$$

and

$$\mathbf{z}_k = f(\mathbf{x}_{k-1}, \boldsymbol{\theta}_k, \mathbf{w}_k, \mathbf{v}_k) \tag{30}$$

A simplified block diagram of a Dual Kalman filter is presented in Figure 15.

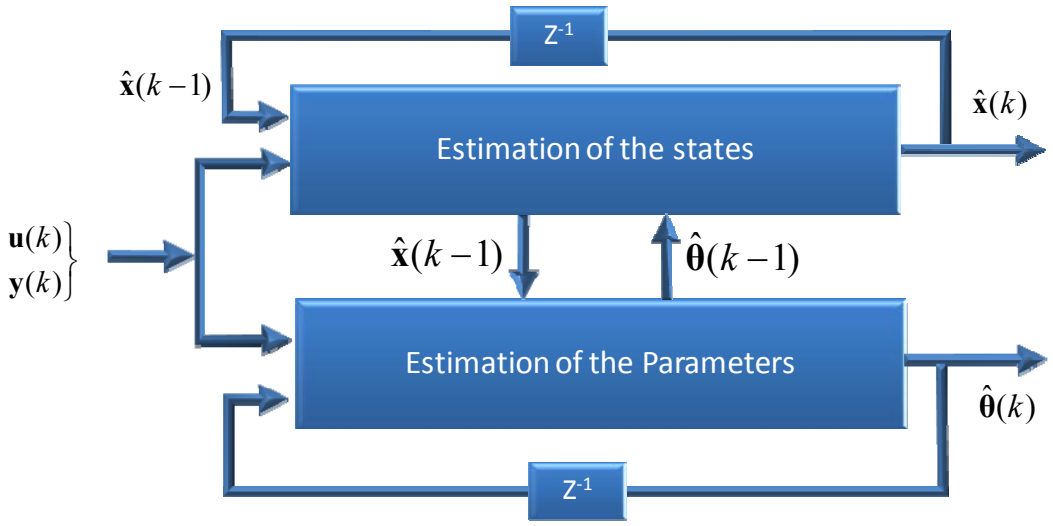


Figure 15: Block Diagram of a Dual Kalman Filter

The Joint Kalman Filter uses a combined state vector, which is formed by the state variables vector of the system and the model parameters. Only one Kalman filter is required and both states and parameters are estimated simultaneously based on the current estimates of the states and the parameters. The augmented state vector is simply formed and it is given in by:

$$\mathbf{x}_{aug}(k) = \begin{bmatrix} \mathbf{x}_k \\ \boldsymbol{\theta}_k \end{bmatrix} \tag{31}$$

The joint dynamical system can be represented simply by:

$$\hat{\mathbf{x}}_{aug}(k+1) = \begin{bmatrix} \mathbf{x}_{k+1} \\ \hat{\boldsymbol{\theta}}_{k+1} \end{bmatrix} = \begin{bmatrix} F(\mathbf{x}_k, \mathbf{u}_k, \mathbf{w}_k, \boldsymbol{\theta}) \\ \boldsymbol{\theta}_k \end{bmatrix} + \begin{bmatrix} \mathbf{0} \\ \mathbf{r}_k \end{bmatrix} \tag{32}$$

Since the joint filter concatenates the state and parameter variables into a single state, it effectively models the cross-covariance between the states and the parameter estimates, which should theoretically provide better estimates, [97]. The coupled covariance matrix, $\mathbf{P}_{aug}$, would provide for treatment of the uncertainty of the states and parameter estimates. In addition, it also models the interaction between the model and parameters, which is given by:

$$\mathbf{P}_{aug} = \begin{bmatrix} \mathbf{P}_{\mathbf{x}_k} & \mathbf{P}_{\mathbf{x}_k\boldsymbol{\theta}_k} \\ \mathbf{P}_{\boldsymbol{\theta}_k\mathbf{x}_k} & \mathbf{P}_{\boldsymbol{\theta}_k} \end{bmatrix} \tag{33}$$

In this research, the dual estimation was handled using joint Kalman filters due to its potential for better performance and implementation simplicity.

3.6. Literature Review about Unscented Kalman Filter

Several research papers have presented different comparisons between the Extended Kalman Filter and the relatively new Unscented or Sigma Points Kalman Filter. Both filters represented different approaches to the problem of recursive states/parameters estimation of nonlinear systems disturbed by process and measurements noise. Every practical industrial process contains some sort of nonlinearities, [93]. Some researchers have asserted that linear systems do not really exist, [94]. Independently of

the absolute non-existence of linear systems, decades of incessant development of the theory of control of linear systems have allowed amazing advanced application of control in areas such as the aerospace, manufacturing and chemical industries.

Undoubtedly, the most widely used nonlinear state estimation technique that has been applied since the sixties is the Extended Kalman Filter, (EKF), [94]. Stanley Schmidt originally proposed its use for solving nonlinear spacecraft navigation problems.

Simon Julier, Jeffrey Uhlmann and Hugh Durrant-Whyte presented the original version of the Unscented Kalman Filter in 1995, [95]. Julier, [96], presented the Scaled Unscented Transformation, which introduces an additional degree of freedom to control the scaling of the sigma points. This avoids the possibility that the resulting covariance can become non-possible semidefinite. This scaled version of the Unscented Transformation seems to have become the standard version. because the scaled version presents the same second order accuracy of the normal or original UT and allow a controllable scaling of the high order errors, [97].

Several research papers have been published. Some of them compare the EKF and the UKF. Others focus on the application of the UKF to specific fields of study. Rudolf van der Merwe, (2004), presented an extensive work in his Ph.D. dissertation, [97]. In his dissertation, van der Merwe studied the performance and divergence properties of the EKF, UKF and the Central Difference Kalman Filter, (CDKF). The CDKF filter is based on Sterling's polynomial interpolation formula. He developed the Square-Root Unscented Kalman Filter, (SR-UKF), the Square-Root Central Difference Kalman Filter, (SR-CDKF). These filters were used to obtain state and parameter

estimations. He also presented a method for the use of the UKF to improve the Sequential Monte Carlo method, which is also known as the Particle Filter. Several application examples were implemented for states, parameter and joint estimation. In his conclusion, it was claimed that there are large performance benefits to be gained by applying Sigma-Point Kalman filters to areas were EKFs have been used as the standard as well to areas where use of the EKF was impossible, [97].

Girish Chowdhary and Ravindra Jategaonkar, (2006), reported their comparison of the EKF, the simplified version of the UKF with additive noise and the augmented UKF for aerodynamic parameter estimation of two aircrafts from real flight data, [98]. For the first study case, which involved the HFB-320 fixed wing research aircraft, a nonlinear model was used for the experiments. The results obtained indicated a very good comparable performance using the three estimation techniques. The excellent performance and close agreement of the three methods was attributed to the use of an accurate mathematical model. For the second study case, which involved a miniature rotary aircraft, a linear model in the hover domain was used for the experiments. The results indicated similar steady state performance between the EKF and the simplified UKF. The augmented UKF displayed marginally better performance. It was concluded that the three estimation methods present comparable performances. The augmented UKF demonstrated a faster convergence than the EKF and the simplified UKF. However, the computational cost of the simplified UKF was three times more than the EKF. The computational cost of the augmented UKF was six times more than the EKF.

In all cases, the continuous-discrete or hybrid versions of the three methods was implemented, [94].

Rambabu Kandepu, Bjarne Foss, and Lars Imsland, (2008), discussed the difference between the EKF and the augmented or general UKF and compared their performance when the filters were applied to four different simulation cases, [93]. A simple approach to handling states constraints was also proposed by the authors. The examples considered were:

- The Van der Pol oscillator,
- An induction machine,
- A gas-phase reversible reaction,
- A solid oxide fuel cell, (SOFC), stack integrated in a gas turbine,(GT), cycle.

The characteristics compared were the robustness of the estimators due to model errors and initial states errors. The authors found that the augmented or general UKF demonstrated consistently improved performance compared to the EKF. The proposed constraints handling method was found to be promising. However, only one example was presented.

Dan Simon, (2008), compared the Linearized Kalman Filter, (LKF), the Extended Kalman Filter and the Unscented Kalman Filter for the study of aircraft turbofan engine health parameter estimation,[99]. The authors concluded that both the EKF and UKF outperformed the LKF. The computational cost of the EKF is one order of magnitude higher than the LKF and the UKF is another order of magnitude higher than the EKF.

Most of the computational cost for the LKF and the EKF was associated with the numerical calculations of the Jacobians. In the UKF case, most of the computational cost was associated with the simulation of the nonlinear system.

3.7. Comparison of the Effect of the Sampling Time on the Performance of the EKF and the UKF

There are some research papers, which draw comparisons between these two variants of the Kalman filter. However, there is not yet a general accepted opinion about their performance. Some researchers, [101], [90], claim that the improvements in accuracy obtained for the UKF are considerable and others, [98], indicate that the accuracy is comparable. There are also disparate results related to the computational cost of the two filters. Some researchers, [99], claim that the UKF has computational costs, which are an order of magnitude higher than the computational cost of the EKF. Still others indicate that the computational costs of the two filters are similar, [93].

Several simulation examples are presented in order to study the issues related to accuracy, computational cost and noisy sensitivity of the EKF and the UKF.

3.7.1. Simulation Example 1: Vertically Falling Body

This particular example has been analyzed previously in the literature, Athans et al, [100]. Julier, Uhlmann and Durrant-Whyte, (2000), used this problem to show the improved accuracy of the new filter presented in their paper, which is now termed the UKF, [101]. Welch and Bishop studied the same problem, [94]. This problem is considered to contain significant nonlinearities in both the states and output equations.

This case consists of the estimation of the altitude, $x_1(t)$, velocity, $x_2(t)$ and the constant ballistic coefficient, $x_3(t)$, of a vertically falling body as it reenters the atmosphere at a very high altitude and at a very high velocity. The measurements are taken at discrete instants of time by radar, which measures range in the presence of discrete white Gaussian noise. The radar was at an altitude, H, of 100,000 ft and the horizontal distance, M, between the vertical trajectory of the body and the radar was 100,000 ft. It is assumed that the effect of gravity is negligible, [100], [101]. Figure 16 sketches the geometry for this example.

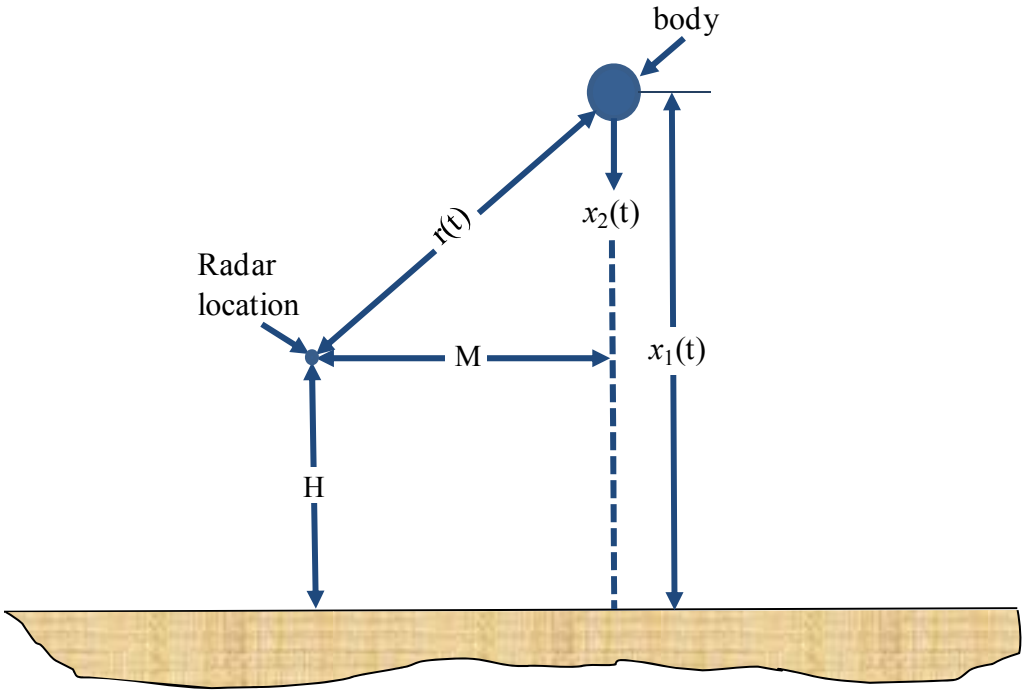


Figure 16: Geometry for the Example of a Vertically Falling Body

The equations of motions for the vertically falling body are given by, [100]:

$$\begin{aligned} \dot{x}_1(t) &= -x_2(t) \\ \dot{x}_2(t) &= -\frac{C_D A \rho}{2m} x_2{}^2(t) \end{aligned} \tag{34}$$

where the air density, ρ, is approximated by the exponential function given by, [100]:

$$\rho = \rho_0 e^{-\gamma x_1(t)} \tag{35}$$

and γ is a constant, (5 x 10^{-5}), which relates the air density with the altitude.

Defining

$$x_3 \equiv C_D A \rho_0 / 2m \tag{35a}$$

a constant, as the ballistic parameter, the continuous-time state equations of the system are given by:

$$\begin{aligned} \dot{x}_1(t) &= -x_2(t) + w_1(t) \\ \dot{x}_2(t) &= -e^{-\gamma x_1(t)} x_2(t)^2 x_3(t) + w_2(t) \\ \dot{x}_3(t) &= w_3(t) \end{aligned} \tag{36}$$

where w_1(t), w_2(t) and w_3(t) are zero-mean uncorrelated noises with covariances given by the process covariance matrix, **Q**(t). The output r(t) is given by:

$$r(t) = \sqrt{M^2 + \left[x_1(t) - H\right]^2} \tag{37}$$

The range was observed at discrete instants of time. Therefore, the observed sequence is given by, [100]:

$$z(k) = \sqrt{M^2 + \left[x_1(k) - H\right]^2} + v(k) \tag{38}$$

where $v(k)$ is the discrete observation white Gaussian noise with zero-mean and constant covariance R(t), which equaled 10^4 ft. The process matrix covariance, **Q**(t), was set to

zero for both filters since the process noise can be used to mask the linearization errors, [101].

The initial true state of the system is given by:

$$\mathbf{x}(0) = \begin{bmatrix} 3\times10^{5} \\ 2\times10^{4} \\ 10^{-3} \end{bmatrix}$$

The initial estimates and covariance of the states are given by:

$$\hat{\mathbf{x}}(0\,|\,0) = \begin{bmatrix} 3\times10^{5} \\ 2\times10^{4} \\ 3\times10^{-5} \end{bmatrix}$$

$$\mathbf{P}(0\,|\,0) = \begin{bmatrix} 10^{6} & 0 & 0 \\ 0 & 4\times10^{6} & 0 \\ 0 & 0 & 10^{-4} \end{bmatrix}$$

A hybrid EKF and a hybrid UKF, as presented in, [94], were implemented using MATLAB scripts. Two numerical integration methods were used to simulate the nonlinear system given by equation (36). The continuous-time part, (time update), of the hybrid EKF and the propagation from $(k-1)^{+}$ to k^{-} of the sigma points time (time update) of the hybrid UKF were simulated. The methods used to represent the simulations were the fourth-order Runge-Kutta method and the Euler's method, which involves rectangular integration.

In order to verify the effect of the measurement frequency and the simulation step size on the accuracy of the filter, several Monte Carlo simulations, which consisted of 50 runs each, were implemented for different values of the measurement frequency and the simulation step size.

The results obtained for a measurement frequency, T_s, equal to 1 Hz, [100], [101], and a simulation step size, T_{sim}, equal to 10 ms are presented in Figure 17, Figure 18 and Figure 19

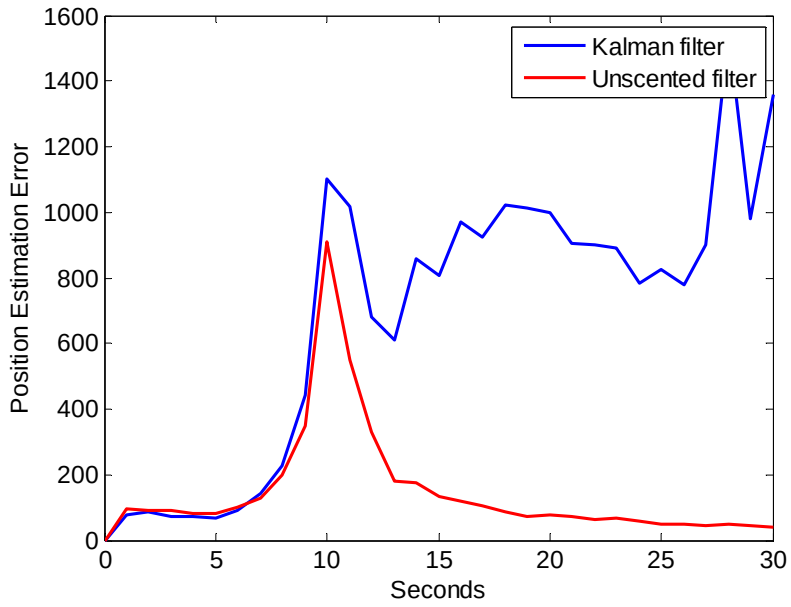


Figure 17: Comparison of the Position Estimation Error of the EKF and the UKF: T_s = 1 Hz, T_{sim} = 10 ms; Fourth-order Runge-Kutta Method

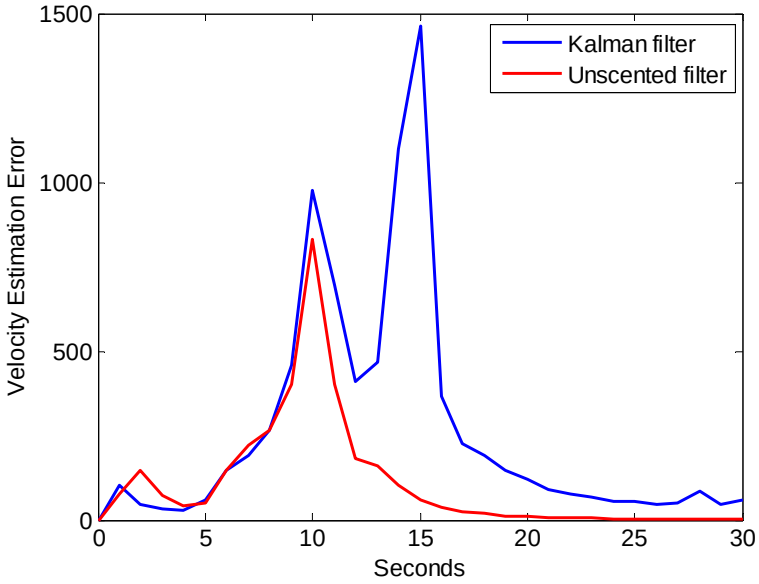


Figure 18: Comparison of the Velocity Estimation Error of the EKF and the UKF: T_s = 1 Hz, T_{sim} = 10 ms; Fourth-order Runge-Kutta Method

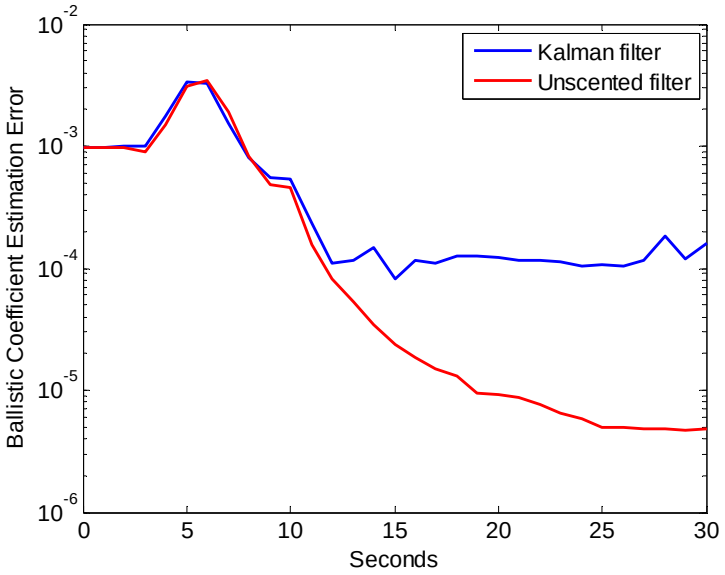


Figure 19: Comparison of the Ballistic Coefficient Estimation Error of the EKF and the UKF: T_s = 1 Hz, T_{sim} = 10 ms; Fourth-order Runge-Kutta Method

The estimation errors obtained for the EKF and the UKF for a measurement frequency of 1 Hz and a simulation step size of 0.1 ms are presented in Figure 20, Figure 21 and Figure 22.

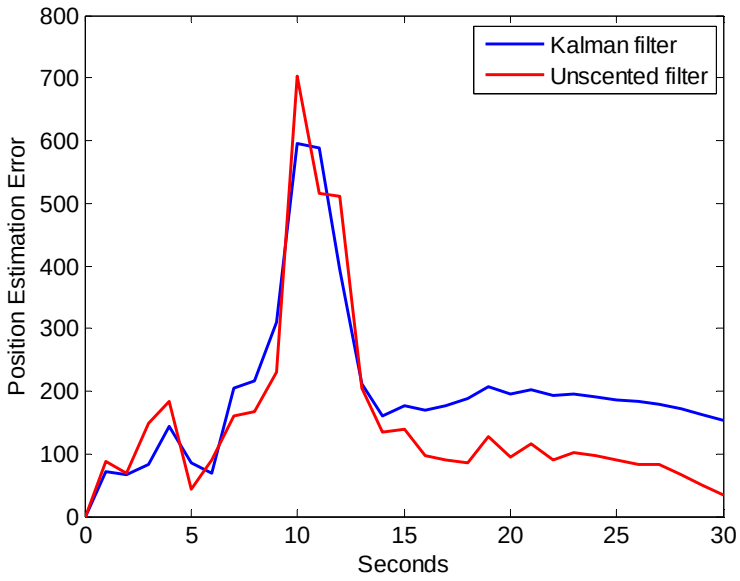


Figure 20: Comparison of the Position Estimation Error of the EKF and the UKF: T_s = 1 Hz, T_{sim} = 0.1 ms; Fourth-order Runge-Kutta Method

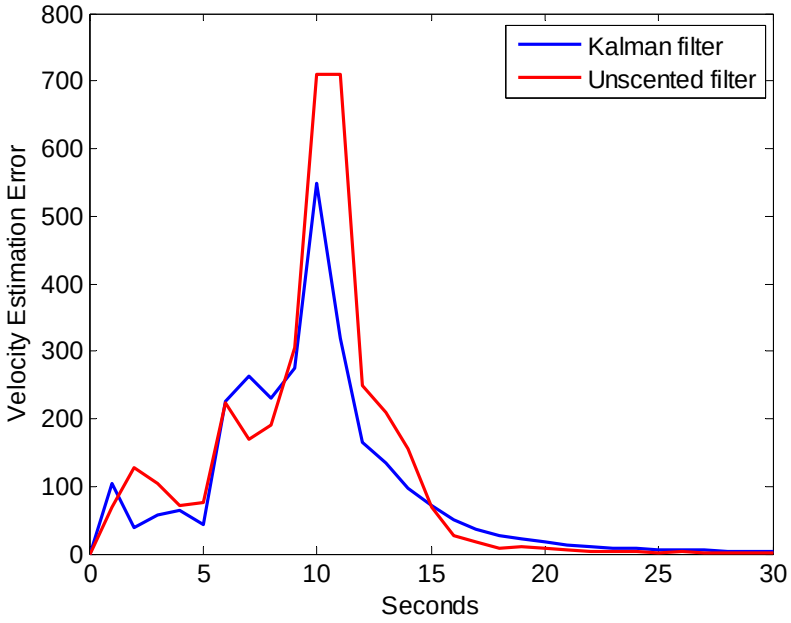


Figure 21: Comparison of the Velocity Estimation Error of the EKF and the UKF: T_s = 1 Hz, T_{sim} = 0.1 ms; Fourth-order Runge-Kutta Method

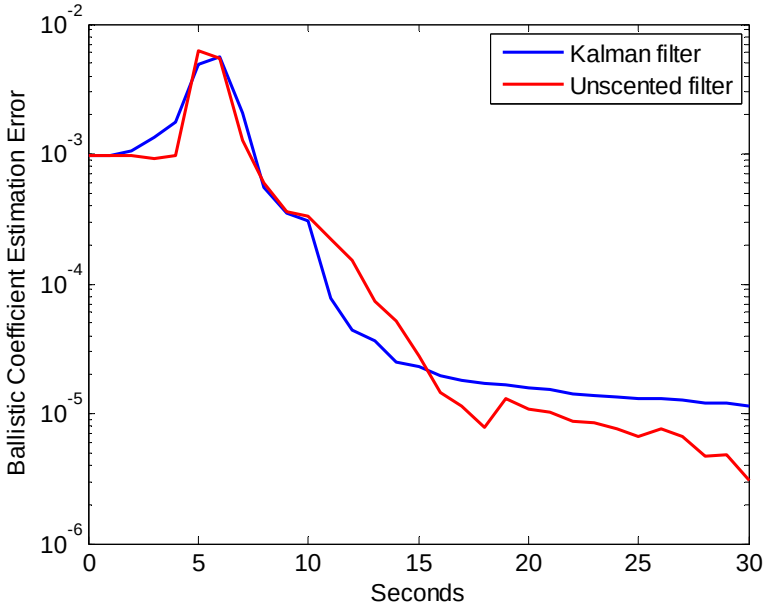


Figure 22: Comparison of the Ballistic Coefficient Estimation Error of the EKF and the UKF: T_s = 1 Hz, T_{sim} = 0.1 ms; Fourth-order Runge-Kutta Method

The results obtained in the two previous Monte Carlos simulations, lead to the conclusion that decreasing the simulation step size improves the accuracy of the EKF. This result could provide insight into the reasons for different claims about accuracy presented in the literature. In the first case, with a measurement frequency of 1 Hz and a simulation step size of 10 ms, it could be claimed that the improvement in accuracy of the UKF is considerable. However, from the results obtained in the second case, it could be claimed that both filters have similar accuracy performance. It is important to note that the only change between the two simulations was the simulation step size used for the numerical integration of the nonlinear system, the time update of the hybrid UKF and the time update of the hybrid UKF.

The computational cost for the first simulation is presented in Table 4.

Table 4: Simulation Time of Call for a Measurement Frequency of 1 Hz and a Simulation Steps Size of 10 ms

Simulation time for call		
Filter Type	Mean	Covariance
EKF	8.57 ms	0.23 ms
UKF	34.57 ms	1.58 ms

Table 4 data indicates that the UKF requires a greater computational time, which was an expected result. The UKF simulation time was 4 times greater than the EKF simulation time. The difference is not one order of magnitude higher but it cannot be considered similar. The simulation times for call obtained for the second simulation were

greater than in the first simulation. However, it was found that the rate of UKF simulation time was approximately four times than the EKF simulation time. In view of these results, further experiments were performed. The new experiments changed the measurement frequency to determine its effect on the accuracy of the estimations. The effects were also investigated in the accuracy comparison between the EKF and the UKF.

For a measurement frequency of 100 Hz and a simulation step of 1 ms, the estimation errors are presented in Figure 23, Figure 24 and Figure 25

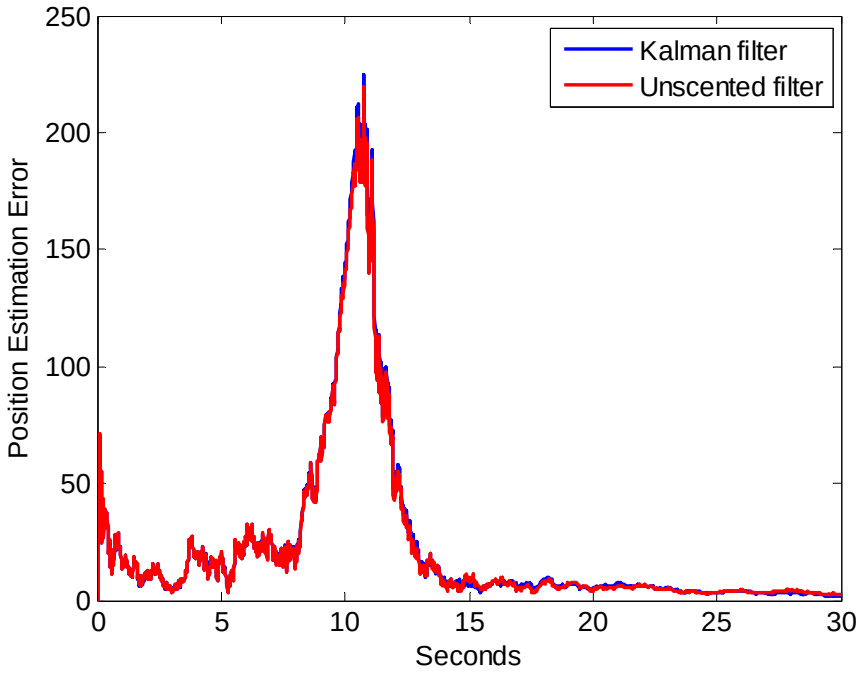


Figure 23: Comparison of the Position Estimation Error of the EKF and the UKF: T_s = 100 Hz, T_{sim} = 1 ms; Fourth-order Runge-Kutta Method

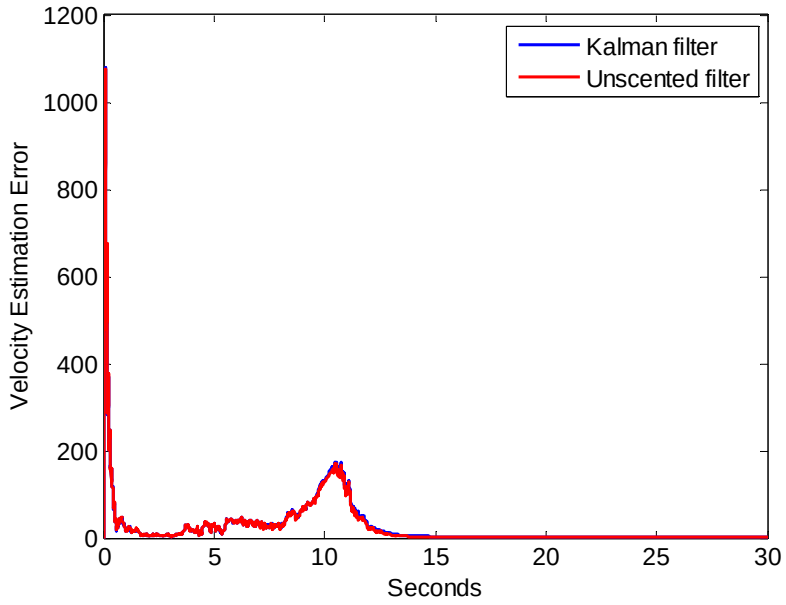


Figure 24: Comparison of the Velocity Estimation Error of the EKF and the UKF: T_s = 100 Hz, T_{sim} = 1 ms; Fourth-order Runge-Kutta Method

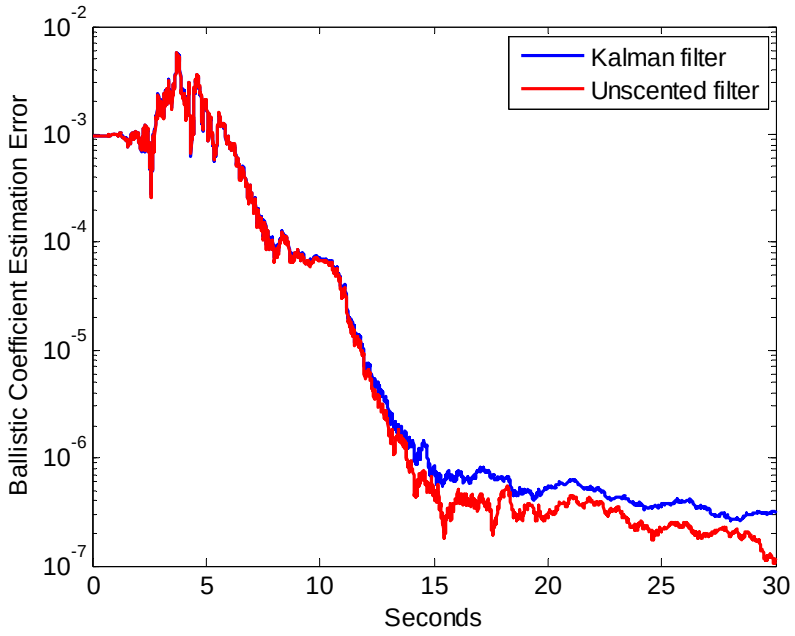


Figure 25: Comparison of the Ballistic Coefficient Estimation Error of the EKF and the UKF: T_s = 100 Hz, T_{sim} = 1 ms; Fourth-order Runge-Kutta Method

The results of the previous simulation, as expected, verified that reduction of the measurement frequency improved the accuracy of both filters, [100]. In this simulation, the estimation errors obtained from both filters were very similar.

The Euler's numerical integration method was also used to compare the effect of the measurement frequency and the simulation step size in the accuracy comparison of the EKF and UKF. The estimation errors obtained for a measurement frequency of 1 Hz and a simulation step size of 10 ms are presented in Figure 26, Figure 27 and Figure 28.

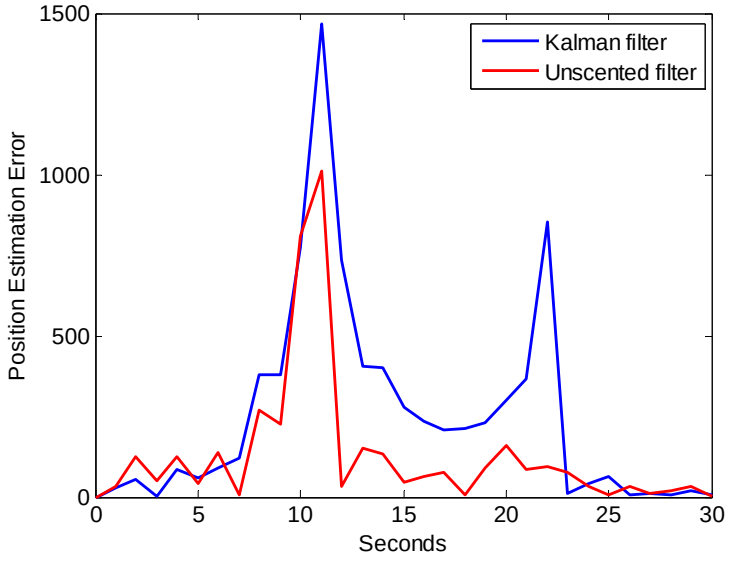


Figure 26: Comparison of the Position Estimation Error of the EKF and the UKF: T_s = 1 Hz, T_{sim} = 10 ms; Euler's Method

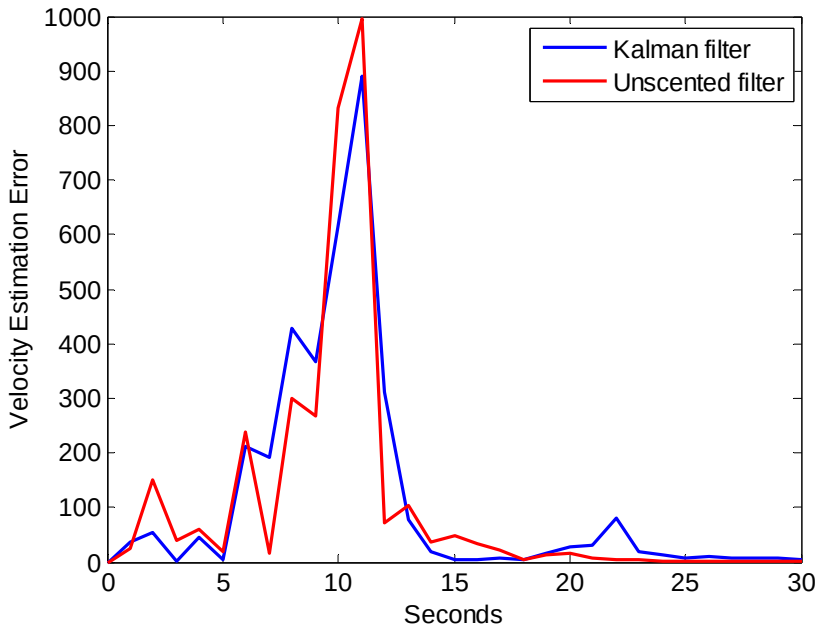


Figure 27: Comparison of the Velocity Estimation Error of the EKF and the UKF: T_s = 1 Hz, T_{sim} = 10 ms; Euler's Method

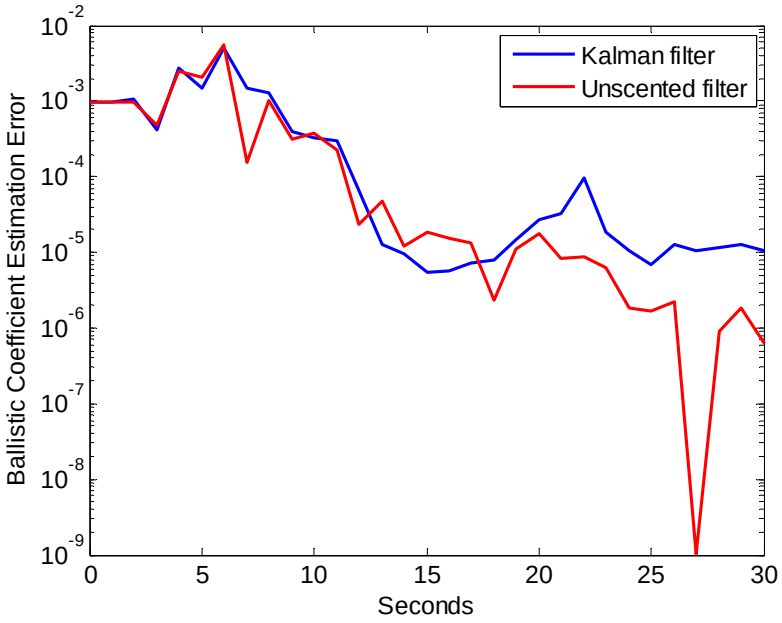


Figure 28: Comparison of the Ballistic Coefficient Estimation Error of the EKF and the UKF: T_s = 1 Hz, T_{sim} = 10 ms; Euler's Method

The estimation errors obtained for a measurement frequency of 1 Hz and a simulation step of 0.1 ms are presented in Figure 29, Figure 30 and Figure 31.

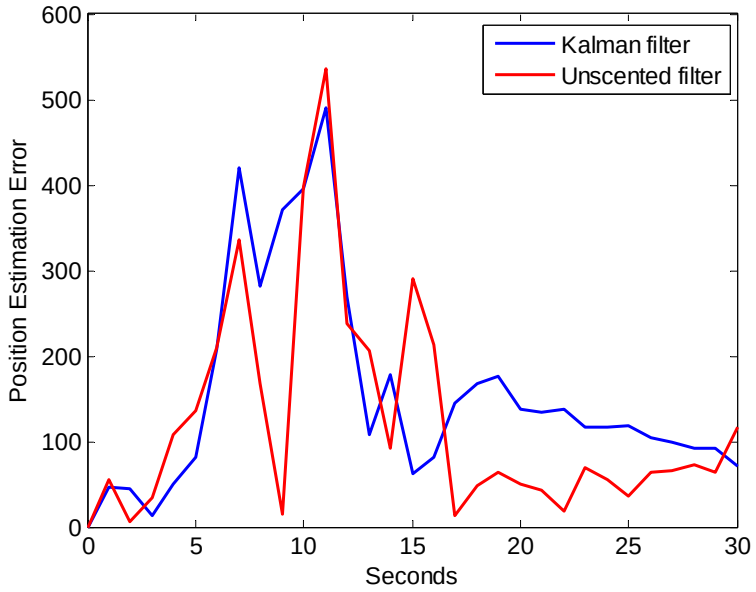


Figure 29: Comparison of the Position Estimation Error of the EKF and the UKF: T_s = 1 Hz, T_{sim} = 0.1 ms; Euler's Method

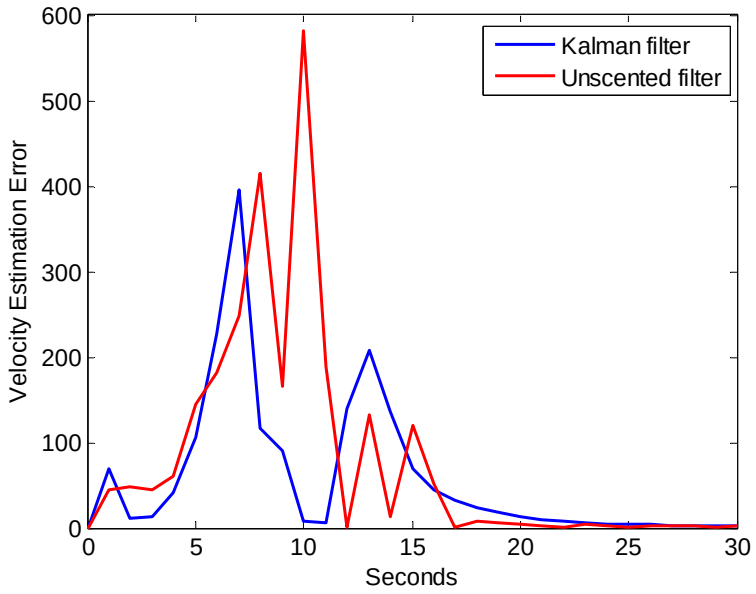


Figure 30: Comparison of the Velocity Estimation Error of the EKF and the UKF: T_s = 1 Hz, T_{sim} = 0.1 ms; Euler's Method

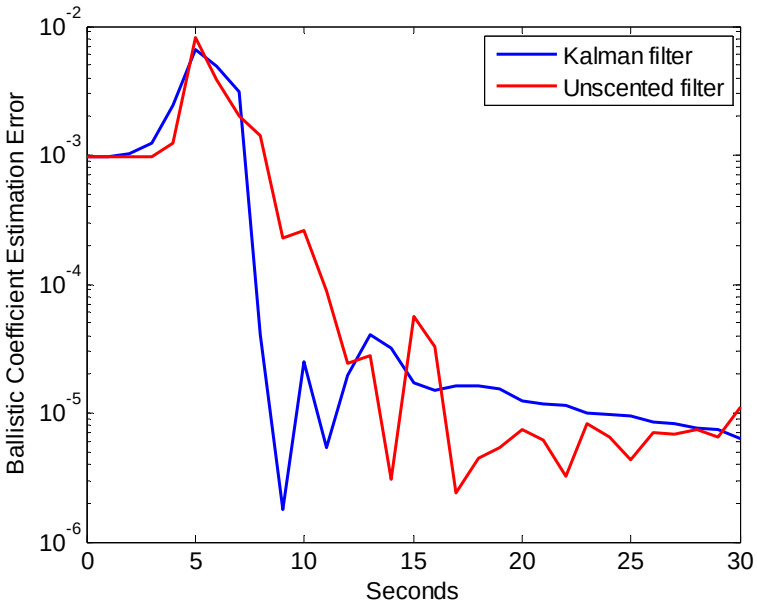


Figure 31: Comparison of the Ballistic Coefficient Estimation Error of the EKF and the UKF: T_s = 1 Hz, T_{sim} = 0.1 ms; Euler's Method

The estimation errors obtained for a measurement frequency of 100 Hz and a simulation step of 1 ms are presented in Figure 29, Figure 30 and Figure 31.

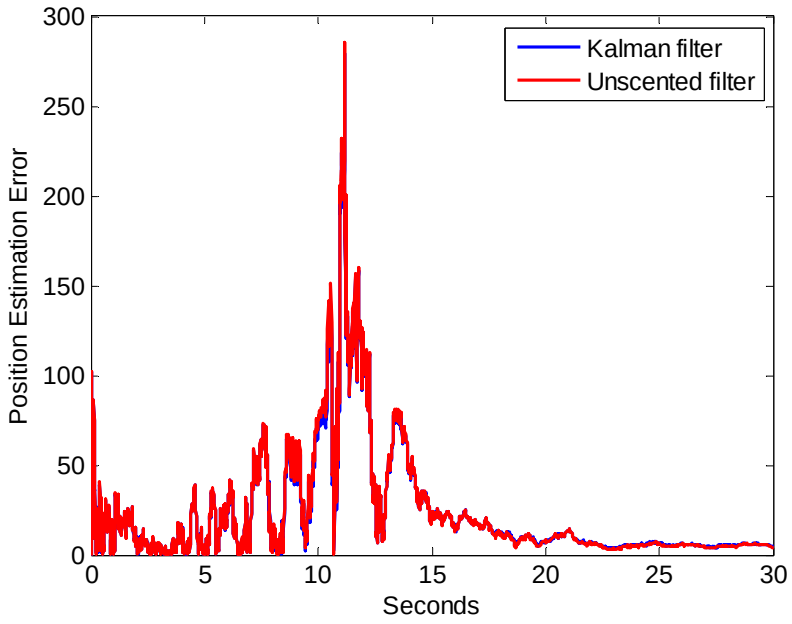


Figure 32: Comparison of the Position Estimation Error of the EKF and the UKF: T_s = 100 Hz, T_{sim} = 1 ms; Euler's Method

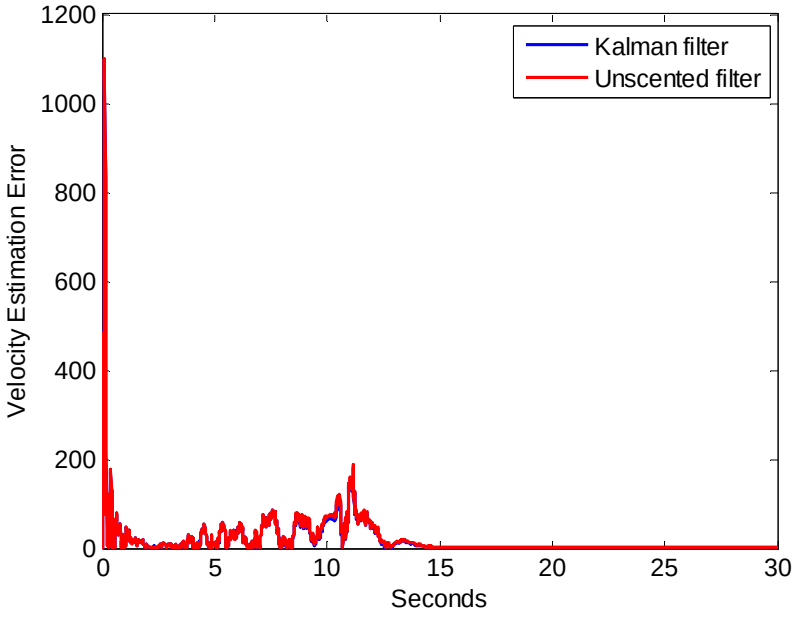


Figure 33: Comparison of the Velocity Estimation Error of the EKF and the UKF: T_s = 100 Hz, T_{sim} = 1 ms; Euler's Method

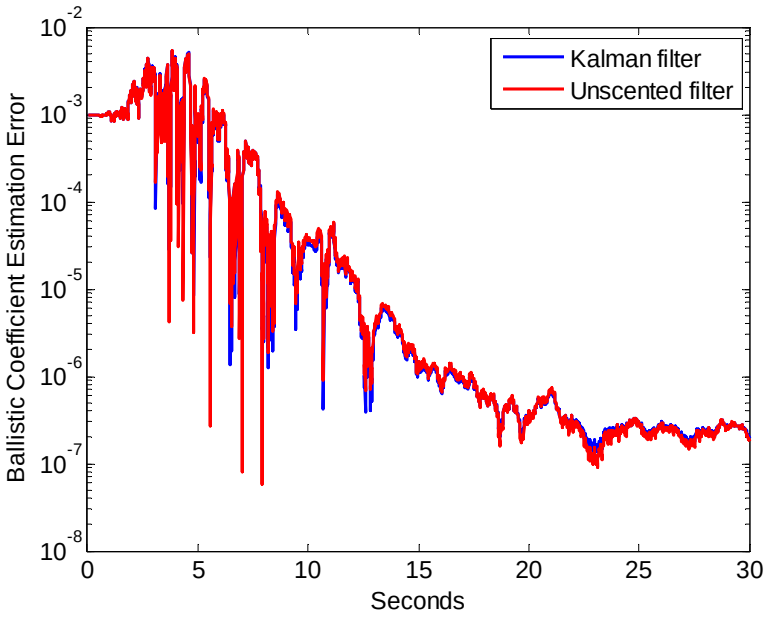


Figure 34: Comparison of the Ballistic Coefficient Estimation Error of the EKF and the UKF: T_s = 100 Hz, T_{sim} = 1 ms; Euler's Method

The results of the simulations indicate that the accuracy of the EKF improve considerably with frequency and simulation step size. In a real-world application, the measurement frequency could not be changed due to the specifications of the sensors used. However, the improvements obtained using smaller simulation step sizes could explain the opposing results obtained for different researchers.

The simulations also indicate that the computational cost for the UKF, as expected, exceed the computational costs for the EKF. It is important to note that in the simulations analyzed, the analytical Jacobians were used. It has been reported that the computational effort is similar when the Jacobians have to be calculated numerically, [93].

3.8. Comparison of the Performance of the EKF and the UKF for Parameter Estimation

Several simulations were implemented to study the performance of the EKF and the UKF for the task of tracking the parameters of a system.

The system used in this simulation was a small-scale helicopter model. Bernard Mettler, [102], developed two linear models for the small-scale Yamaha R-50. One model was concerned with the hover condition and the other model was concerned with the cruise condition. As a simple experiment of parameter estimation, one of the parameters, the stability derivative X_u, was perturbed and it varying value was estimated. A joint Kalman filter was used for the estimation of the X_u parameter from Mettler's model for the cruise flight condition. Even though the model was linear in the states, when parameters are estimated, the new augmented system becomes nonlinear. Therefore, it was necessary to use a variant of the Kalman filter for nonlinear systems. The hybrid EKF and the hybrid UKF versions were used in the experiments. In this case, the filters were implemented as Level-2 M-file S-functions to be used as a block in Simulink. This implementation permitted the use of the Kalman filters with the previously implemented Mettler's model, [103], for control of UAVs.

The equations for the augmented system were rearranged in a convenient order. They are presented in Table 5.

Table 5: Equations for the Augmented Mettler's Model for the Estimation of the Stability Derivative, X_u

$$\dot{u} = X_u * u - g * \theta + X_a * a$$

$$\dot{v} = Y_v * v + g * \phi + Y_b * b + Y_{ped} * \delta_{ped}$$

$$\dot{w} = Z_a * a + Z_b * b + Z_w * w + Z_r * r + Z_{col} * \delta_{col}$$

$$\dot{p} = L_u * u + L_v * v + L_b * b + L_w * w$$

$$\dot{q} = M_u * u + M_v * v + M_a * a + M_w * w + M_{col} * \delta_{col}$$

$$\dot{r} = N_v * v + N_p * p + N_w * w + N_r * r + N_{rfb} * r_{fb}$$

$$\tau_f \dot{a} = -\tau_f * q - a + A_b * b + A_c * c + A_{lat} * \delta_{lat} + A_{lon} * \delta_{lon}$$

$$\tau_f \dot{b} = -\tau_f + B_a * a - b + B_d * d + B_{lat} * \delta_{lat} + B_{lon} * \delta_{lon}$$

$$\tau_s \dot{c} = -\tau_s * q - c + C_{lon} * \delta_{lon}$$

$$\tau_s \dot{d} = -\tau_s * p - d + D_{lat} * \delta_{lat}$$

$$\dot{r}_{fb} = K_r * r + K_{rfb} * r_{fb}$$

$$\dot{\phi} = p$$

$$\dot{\theta} = q$$

$$\dot{X}_u = 0$$

The X_u parameter was assumed a constant, which is customary for parameter estimation. The measured outputs are given by:

$$y(k) = \begin{bmatrix} u \\ v \\ w \\ p \\ q \\ r \end{bmatrix} \tag{39}$$

In order to reproduce a "real world" situation for the small-scale helicopter simulation, the measured outputs were distorted with a "reasonable amount" of noise. The measurement noise covariance used for the simulation was

$$\mathbf{R} = \text{diag}([100\ 100\ 100\ 3\text{x}10^{-2}\ 3\text{x}10^{-2}\ \ 3\text{x}10^{-2}]).$$

Figure 35 and Figure 36 present the measured noisy outputs. After a careful tuning was completed, the simulation was run to measure the tracking of the filter.

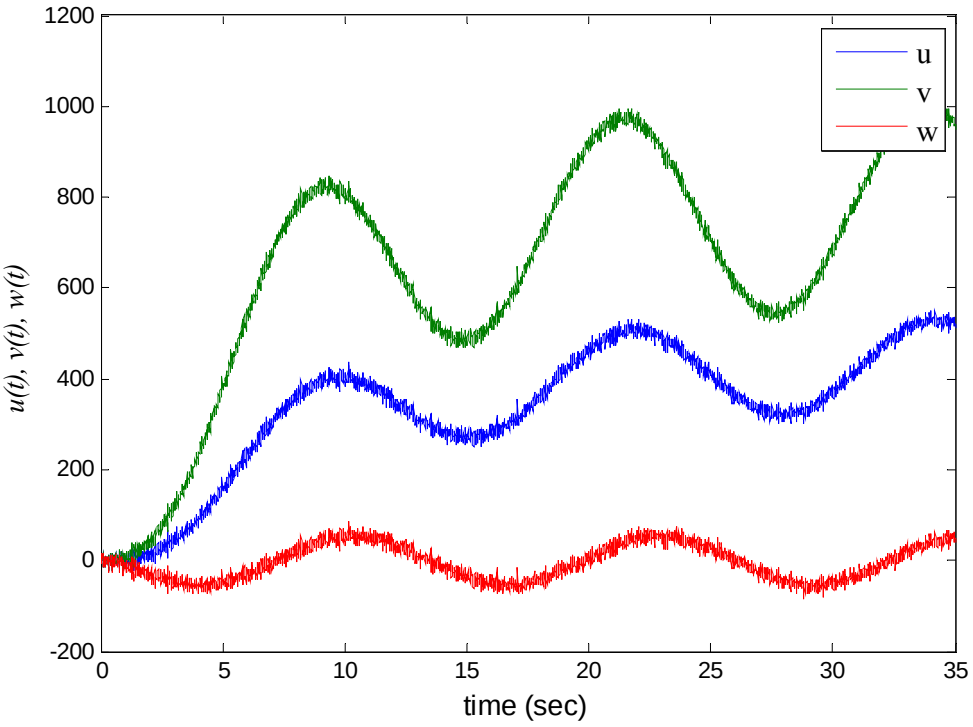


Figure 35: Noisy Translational Velocities, *u*, *v*, and *w*

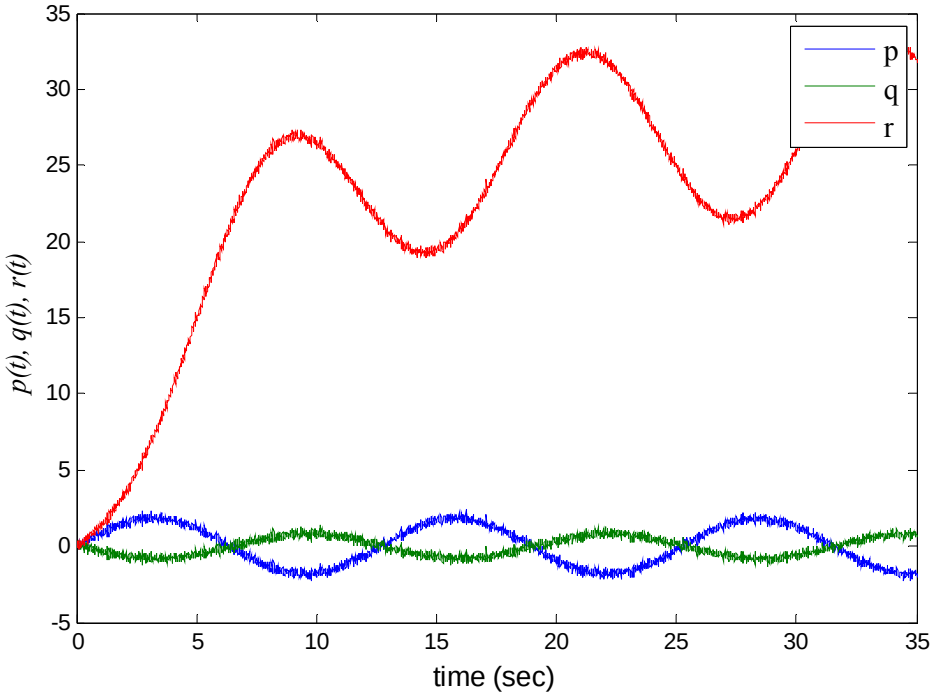


Figure 36: Noisy Rotational Rates, p, q, and r

Using an incorrect initial value for the X_u parameter, the ability of the filter to converge to the true values was tested. Figure 37 presents the results obtained for the hybrid EKF and the hybrid UKF.

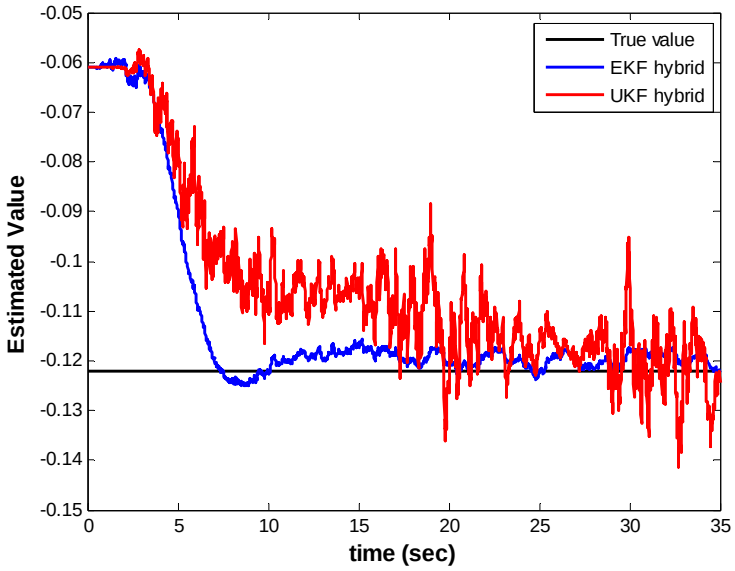


Figure 37: Tracking of the Parameter X_u from an Incorrect Value of -0.062. The Real Value was -0.122

The "tuning" was achieved by varying the values of a "fictitious" process noise, which is a common strategy used for estimating "constants", [94]. The UKF presents considerable sensitivity to changes in the process noise of X_u, which provided for tuning the tracking of the UKF. The EKF presented a lower sensitivity to the fictitious process noise but considerable sensitivity to the cross-covariance term $R_{X_u u}$. The UKF seemed to be completely insensitive to variations of the cross-covariance term. Both filters were completely insensitive to the cross-covariance term R_{uX_u}.

The hybrid EKF filter was observed to have a little faster convergence to the real value than the hybrid UKF. The Root Mean Square Error, (RMSE), and the Root Mean Absolute Error, (RMAE), are presented in Table 6 for an initial value of -.061. The data indicate that the better tracking performance was associated with the hybrid EKF.

Table 6: RMSE and RMAE for the Tracking of the Parameter X_u when the Initial Value was -0.061

Filter Type	RMSE	RMAE
Hybrid EKF	0.0218895218742776	0.104069651011363
Hybrid UKF	0.0259090409155798	0.13613440975281

The simulation results for the tracking with an initial value of -0.183 are presented graphically in Figure 38.

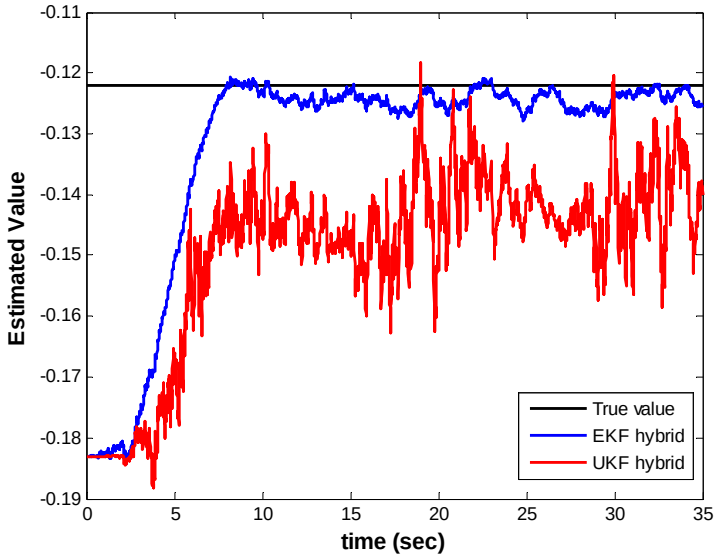


Figure 38: Tracking of the Parameter X_u from an Incorrect Initial Value of -0.183. The Real value of X_u was -0.122

The Root Mean Square Error, (RMSE), and the Root Mean Absolute Error, (RMAE), are presented in Table 7 for an initial value of -.183. The data indicate that the better tracking performance was associated with the hybrid EKF.

Table 7: RMSE and RMAE for the Tracking of the Parameter X_u when the Initial Value was -0.183

Filter Type	RMSE	RMAE
Hybrid EKF	0.0210274279728388	0.101713516955333
Hybrid UKF	0.0310201803077989	0.165126063672822

The simulation results for the tracking when the real value changed from -0.122 to zero are presented graphically in Figure 39.

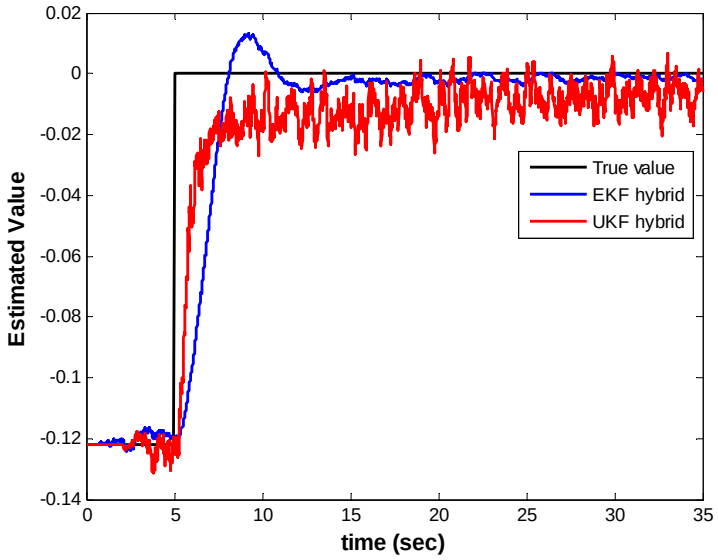


Figure 39: Tracking of the Parameter X_u when its Real Value Changed from -0.122 to Zero. A Correct Initial Value was used in the Simulation

The results obtained in the two previous simulations, and in other simulations, which were not presented, indicate that the hybrid EKF is better at estimating the correct value of the parameter X_u when the initial value is incorrect. The farther the initial value

is from the real value, the worse the tracking of the hybrid UKF. However, the hybrid EKF maintains a similar tracking performance for different incorrect initial values. It is possible to improve the tracking of the hybrid UKF by increasing the fictitious process noise of the parameter X_u. However, the result is a more noisy response.

In the next simulation, the tracking of the value of the parameter X_u was studied when the real value of the parameter changes from -0.122 to 0 in 5 sec. This time the correct initial value was used. The settings that were used in past simulation were maintained. The Root Mean Square Error, (RMSE), and the Root Mean Absolute Error, (RMAE), are presented in Table 8. The data indicate that the better tracking performance was associated with the hybrid EKF.

Table 8: RMSE and RMAE for the Tracking of the Parameter X_u when its Real Value Changed from -0.122 to 0

Filter Type	RMSE	RMAE
Hybrid EKF	0.0236099865712331	0.09085880805346
Hybrid UKF	0.0201290648298945	0.110209529260923

In the next simulation, the tracking of the value of the parameter X_u was studied when the real value of the parameter changes from -0.122 to -0.244 in 5 sec. The Root Mean Square Error, (RMSE), and the Root Mean Absolute Error, (RMAE), are presented in Table 9. The simulation results for the tracking when the real value changed from -0.122 to -0.244 are presented graphically in Figure 40.

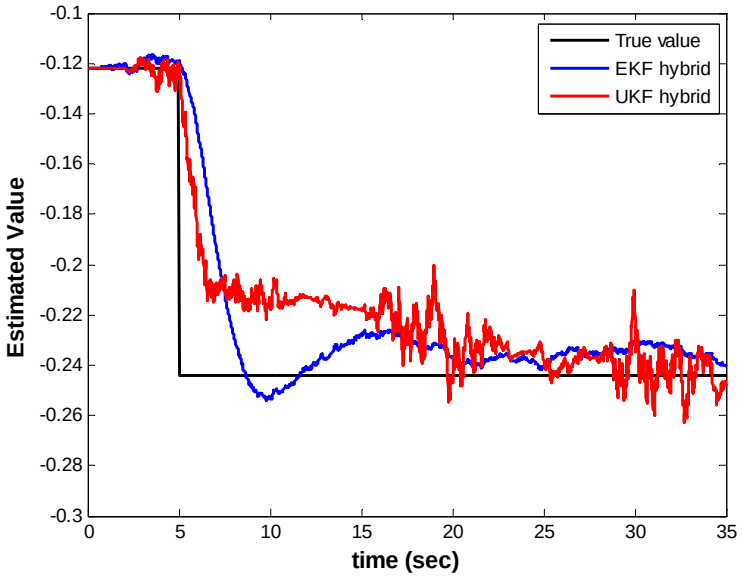


Figure 40: Tracking of the Parameter X_u when its Real Value Changed from -0.122 to -0.244. A Correct Initial Value was Used in the Simulation

Table 9: RMSE and RMAE for the Tracking of the Parameter X_u when its Real Value Changed from -0.122 to -0.244

Filter Type	RMSE	RMAE
Hybrid EKF	0.0256160351945344	0.11648595078295
Hybrid UKF	0.0246661803845061	0.130525724707613

The results obtained in the previous simulations indicate that the hybrid EKF converges faster to the real value of the parameter X_u. The estimation errors presented in

Table 8 and Table 9 favor the hybrid EKF. Again, it is possible to improve the tracking performance of the hybrid UKF. However, a more noisy response is obtained.

Figure 41, Figure 42 and Figure 43 present the responses in the case where the change in the parameter value was substantial. In the case of big positive changes, the hybrid UKF presented a faster response and the RMSE and the RMAE were less than the corresponding errors of the hybrid EKF.

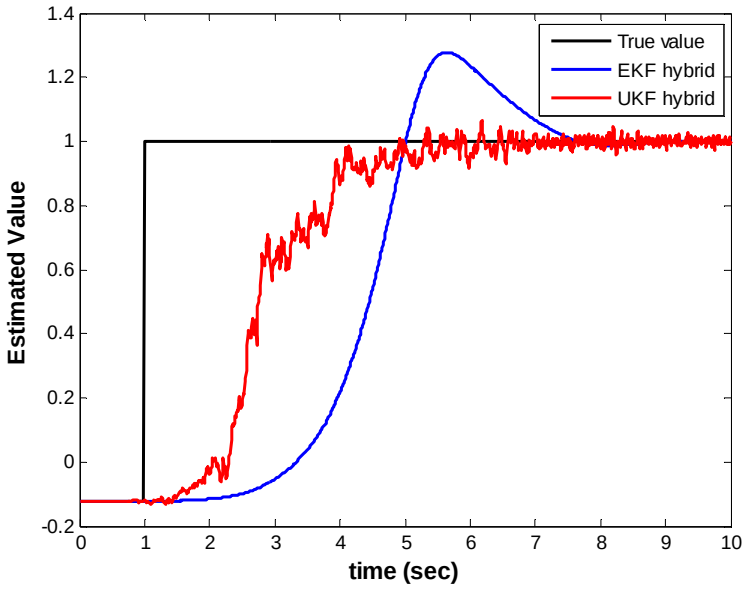


Figure 41: Tracking of the Parameter X_u when its Real Value Changed from -0.122 to 1. A Correct Initial Value was Used in the Simulation

The Root Mean Square Error, (RMSE), and the Root Mean Absolute Error, (RMAE), are presented in Table 10.

Table 10: RMSE and RMAE for the Tracking of the Parameter X_u when its Real Value Changed from -0.122 to 1

Filter Type	RMSE	RMAE
Hybrid EKF	0.606521236581858	0.631802091542201
Hybrid UKF	0.435385429859299	0.47536013578397

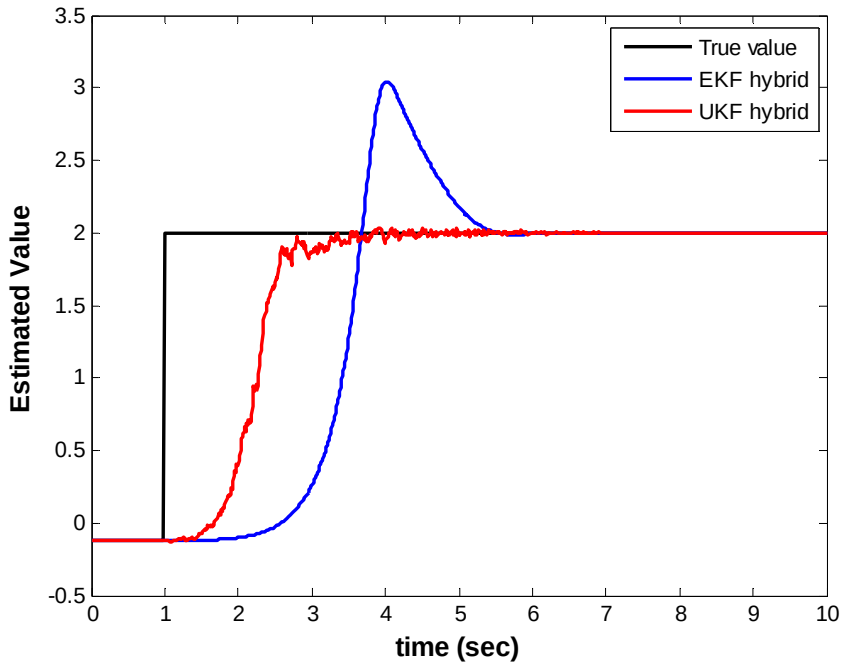


Figure 42: Tracking of the Parameter X_u when its Real Value Changed from -0.122 to 2. A Correct Initial Value was Used in the Simulation

The Root Mean Square Error, (RMSE), and the Root Mean Absolute Error, (RMAE), are presented in Table 11.

Table 11: RMSE and RMAE for the Tracking of the Parameter X_u when its Real Value Changed from -0.122 to 2

Filter Type	RMSE	RMAE
Hybrid EKF	1.00251427497877	0.755009947529238
Hybrid UKF	0.679443476319661	0.514003734843681

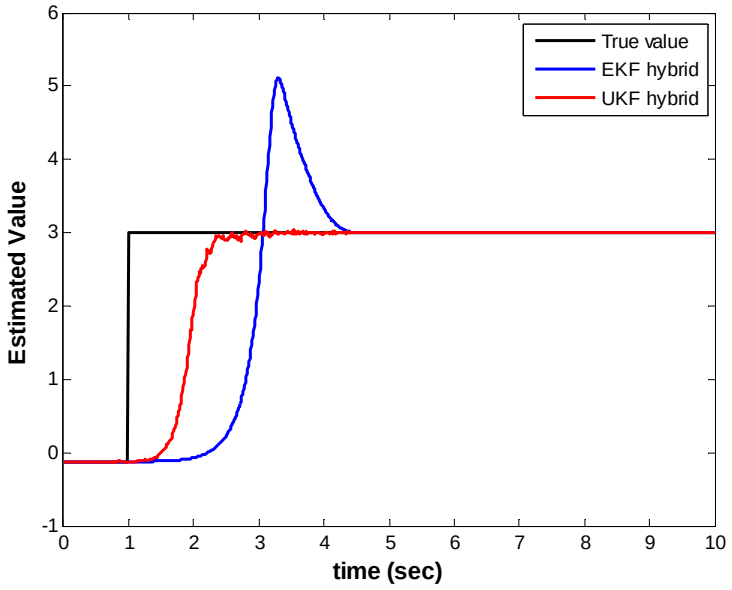


Figure 43: Tracking of the Parameter X_u when its Real Value Changed from -0.122 to 3. A Correct Initial Value was Used in the Simulation

The Root Mean Square Error, (RMSE), and the Root Mean Absolute Error, (RMAE), are presented in Table 12.

Table 12: RMSE and RMAE for the Tracking of the Parameter X_u when its Real Value Changed from -0.122 to 3

Filter Type	RMSE	RMAE
Hybrid EKF	1.34219238805557	0.82496449532556
Hybrid UKF	0.884991339642728	0.541659062185965

In the case of a big negative change in the value of the X_u parameter, Figure 44 presents the responses of the filters.

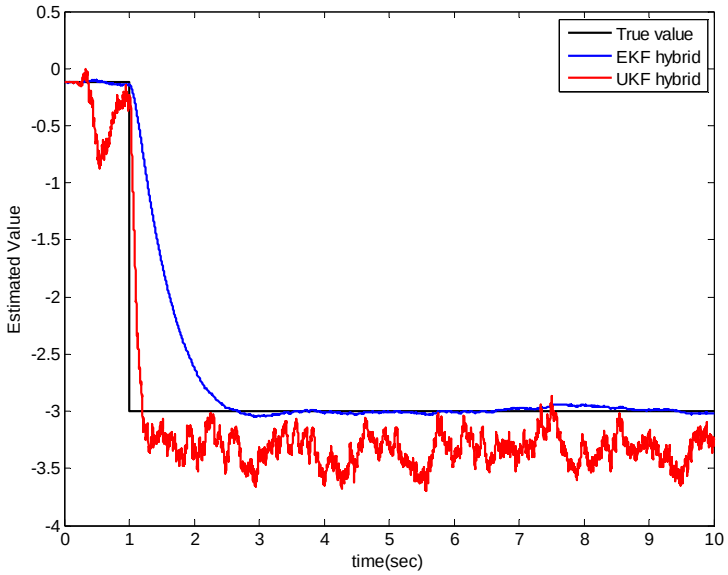
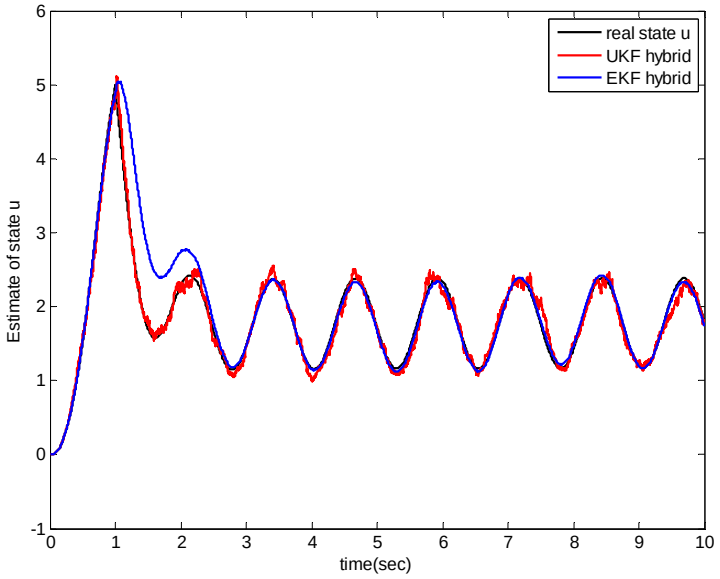
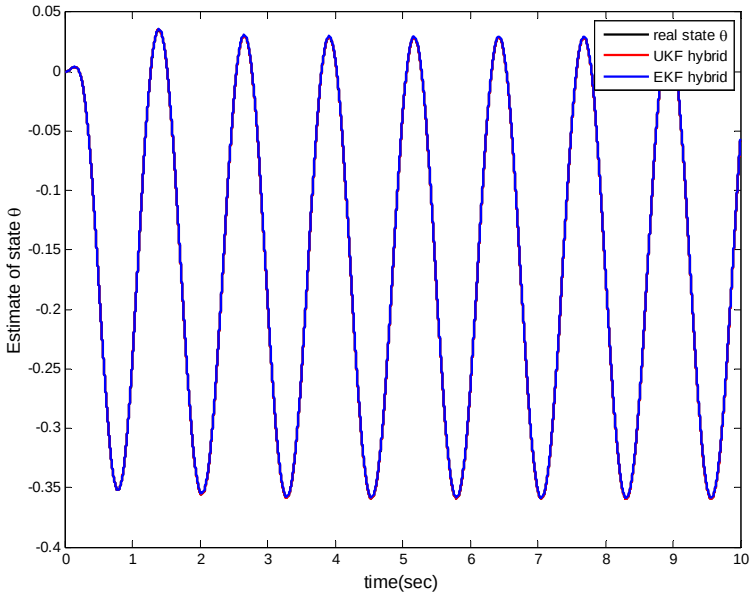


Figure 44: Tracking of the Parameter X_u when its Real Value Changed from -0.122 to -3

EKF converges to a more accurate final value. Even though, the filters were able to converge to the real values of all states.



(a)



(b)

Figure 45: Responses of the Filters when the X_u Parameter Changed its Value from -0.122 to -3. (a) Estimates of the State *u*, (b) Estimates of the State θ

Figure 45a indicates that the hybrid UKF converges faster than the EKF but to a biased value. Figure 45b shows that in the case of state θ, both filters converge accurately.

3.8.1. Noise Sensitivity

Simulations were run to establish the filters behavior when the values of the covariance of the measurement noises were multiplied by factor that varied from $1x10^{-10}$, which resulted in a noiseless system, to a factor of 1, which resulted in the original noisy system. Figure 46, Figure 47 and Figure 48 present the responses obtained for the different values of this factor.

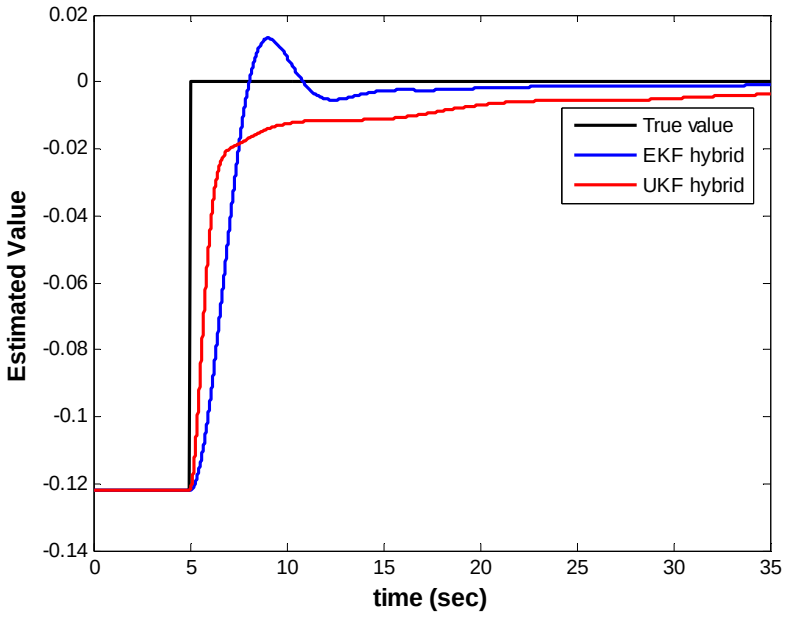


Figure 46: Response of the Filters to a Noiseless System: Factor = 10^{-10}

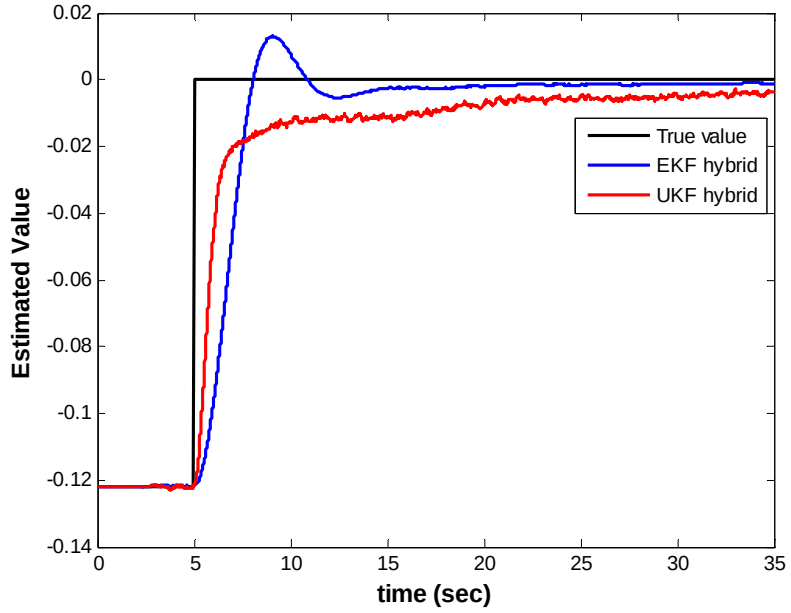


Figure 47: Response of the Filters to a Moderately Noisy System: Factor = 10^{-2}

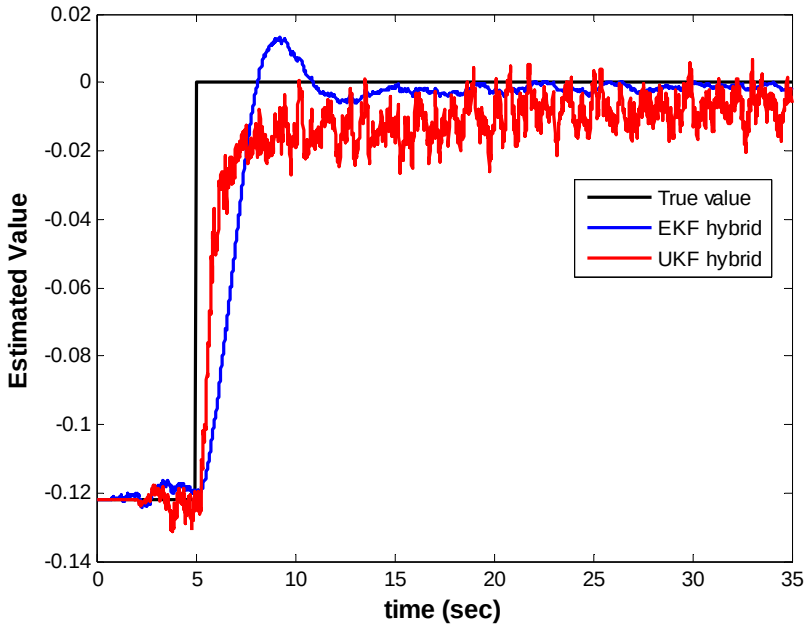


Figure 48: Response of the Filters to the Original Noisy System: Factor = 1

The previous three simulations indicate that the hybrid UKF is more sensitive to the measurement noise than the hybrid EKF.

3.9. Discussion

The results obtained from the simulations certain conclusions to be drawn. In the Vertically Falling Body example, it was demonstrated that the performance of the UKF was superior for a simulation time of 10 msec. However, when the simulation time was decreased to 0.1 ms; the performance of the filters was clearly similar. This result is significant since the only parameter changed in the simulation was the simulation step size, which caused a significant variation in the performance of the hybrid EKF. Both filters improved their performance. However, the difference between them was negligible. Computational cost also favored the EKF.

For the case of parameter estimation for small-scale helicopter simulation, several aspects were studied. Hybrid filters were used but in the Simulink environment. The continuous-time part of the simulations was under the control of the Simulink engine. The fixed-step fourth-order Runge-Kutta, (od4), was used for the simulations. The results obtained for the tracking of the parameter value when an incorrect initial value was assumed favored the hybrid EKF. This conclusion is supported by the data displayed in the corresponding figures and in the RMSE and RMAE values calculated for this case.

The results obtained for the case when a sudden change in the value of the parameter occurred also indicated a better performance associated with the hybrid EKF. The final simulations indicated that the hybrid UKF was more sensitive to noise than the hybrid EKF.

The performance changes seem to be related with the way the filters were simulated. This could be explained by the observation that in the hybrid filters, the time update was implemented using the continuous-time nonlinear model of the system under consideration. In the case of the hybrid EKF, the time update was implemented by the equations, [94]:

$$\dot{\hat{\mathbf{x}}} = f(\hat{\mathbf{x}}, \mathbf{u}, 0) \tag{40}$$

$$\dot{\mathbf{P}} = \mathbf{AP} + \mathbf{PA}^T + \mathbf{LQL}^T \tag{41}$$

For the case of the hybrid UKF, only the sigma points were propagated using equation (40). The hybrid EKF had the advantage of propagating the covariance matrix in a "more" exact way. The hybrid UKF calculated the covariance matrix by the equation, [101]:

$$\mathbf{P}_k^- = \sum_{i=0}^{2n} W_i \{\chi_i(k+1|k) - \hat{\mathbf{x}}(k+1|k)\} \tag{42}$$

where $\chi_i(k+1|k)$ were the sigma points and $\hat{\mathbf{x}}(k+1|k)$ was the predicted mean of the sigma points.

Chapter 4

Model Predictive Control Literature Review

This chapter presents a summary of the literature review on adaptive model predictive control, as well as a summary of the literature review on fault-tolerant model predictive control. The following chapter, Chapter five, will present the developed control scheme, which effectively blends the adaptive MPC with the fault-tolerant MPC through the use of an joint EKF.

4.1. Literature Review about Adaptive Model Predictive Control

Aggelogiannaki and Sarimveis, (August 2007), [104], presented a hierarchical multiobjective adaptive model predictive control. The Pareto optimal set of the multiobjective optimization problem was approximated using a Simulated Annealing, (SA), algorithmic approach. The algorithm returns a single solution, which corresponds to the lexicographic ordering approach. Different initial temperatures were assigned to each objective according to their position in the hierarchy. A major advantage of the proposed method was its low computational cost, which is a very critical issue for online applications. The MPC control scheme was an adaptive discrete-time model of the system, which was developed using a radial-basis-function, (RBF), neural network architecture. A key issue in the success of the adaptation strategy was the introduction of

a persistent excitation constraint, which was transformed to a top-priority objective. Only an unconstrained version of the adaptive MPC was considered.

Kim and Sugie, (January 2008), [105], presented an adaptive receding horizon predictive control for constrained discrete-time linear systems with parameter uncertainties. It was claimed that an adaptive parameter estimation algorithm suitable for MPC was proposed. This estimation was based on the methodology of the Moving Horizon Estimation. The estimation algorithm enables the prediction of a monotonically decreasing worst-case estimation error bound over the prediction horizon of MPC. This provided that future model improvement could be considered explicitly. Only the noise-free case and the state-feedback case were considered in the research.

Corona and De Schutter, (March 2008), [106], present an adaptive Cruise Control for a SMART car, which is used as a comparison benchmark for several mode predictive control methods for nonlinear and piecewise affine, (PWA), systems. The prediction model and control approaches were compared:

- A nonlinear MPC with the nonlinear prediction model was approximated using a first-order Euler approximation,
- MPC with a Piecewise Affine model was represented as a mixed logical dynamical, (MLD), model. The online optimization for this MPC approach was a mixed-integer linear program, (MILP),
- An offline PWA-MPC approach used a multi-parametric MILP. This strategy avoids solving optimization on-line and the online calculation was reduced to the mere search in a lookup table,

- An approach, which only considers a PWA, and the gear was presented. This approach is still a MILP and if the prediction horizon is short, it will reduce the computational cost of the online MILP,
- A tangent approximation of the nonlinear friction's nonlinearity was considered. The PWA was obtained for the current operating point. The MILP structure was similar to that of approach 2,
- A basic tangent approximation was considered but neglected the effect of the gear. This approximation has the advantage of leading to an online linear optimization problem, which requires less computational power,
- A basic gain scheduling approximation was implemented considering six linear models for the nonlinear friction curve,
- A proportional-integral, (PI), controller was considered. The controller first computed the desired acceleration and the actuators regulated the throttle, the gear and the braking action in order to better achieve the desired values of the acceleration.

The results obtained in this benchmark comparison indicate that in terms of performance the PI performed the worst. In terms of computational cost, the online PWA-MPC-MILP was the most demanding approach. In terms of constraints violations, the source of numerous constraint violations was the bigger mismatch of the linear methods compared with the MLD or NMPC methods.

C.-H. Lu, and C.-C Tsai, (March 2008), [107], presented an adaptive predictive control with recurrent neural networks, (RNN). The control was for a class of discrete-

time nonlinear systems described by a nonlinear autoregressive moving averaging, (NARMA), model. This class of discrete-time nonlinear control was approximated by the combination of a linear model and a RNN model. A recursive least-squares, (RLS), estimation method was used for determining the unknown linear dynamic system parameters. The RNN was used to develop a neural predictor for model errors. The model predictive control implemented was unconstrained, which provides for obtaining a closed form control law. The control horizon was set to one in order to reduce the computational load, even thought the sampling time of three seconds seems to provide enough time to allow a larger control horizon. The stability was claimed to rely on the convergence of the estimates of the linear parameters and the neural error predictions. To insure that the identification process would be successful, persistently exciting, (PE), signals were used as the testing signals for accomplishing PE conditions.

K. R. Muske, J.C. Peyton Jones, and E.M. Franceschi, (July 2008), [108], present an adaptive, linear state-space analytical model-predictive controller for spark ignition, (SI), engine air-fuel ratio control. The process model used for this research was a parameterized linear time-varying discrete-time state-space model. The input to the model was a multiplier of the base fuel flow rate calculated by the Engine Control Module, (ECM), obtained during engine calibration. The measured output was the equivalence ratio, which is the inverse of the air-fuel ratio. The output was determined from the precatalyst wide-ranging universal exhaust gas oxygen, (UEGO), sensor. The model's parameters were obtained from step responses of a Ford 2-L I-4 engine. Despite the significant complexity in the system dynamics due to the effects of fuel puddling,

manifold wall wetting and the intake manifold hydrodynamics, the engine's step response was approximated by a first-order plus dead time, (FOPDT), model. The model parameters were scheduled as:

- The model gain is assumed constant and equal to 0.9,
- The time constant was scheduled only by the engine's speed,
- The time delay was scheduled by the speed and load conditions of the engine.

A Kalman filter was used to obtain an estimate of the model's states.

4.2. Literature Review of Fault-Tolerant Model Predictive Control

Maciejowsky and Jones (June 2003), [109], demonstrated that the fatal crash of El Al Flight 1862 might have been avoided by using MPC-based fault-tolerant control. A detailed nonlinear model of the aircraft was used to show that it is possible to reconfigure the controller so the aircraft could be flown successfully down to the ground. The proposed fault-tolerant controller was composed of three components:

- The block FID, which performs detection and identification of the fault's effects. This block was not designed by the authors and was assumed to be present.
- A reference model, which uses the pilot commands to generate a *reference trajectory* for the state's state vector.
- A reconfigurable MPC.

The objective of the reconfigurable MPC was to track the reference trajectory using the output of the FDI block to update its internal model constraints.

Qi et al, (December 2007), [110], presented a fault adaptive control methodology against actuator failure. A Square Root Unscented Kalman Filter, (SR-UKF), was used for on-line estimation of both the flight states and the Actuator Healthy Level, (AHL), parameters of a rotorcraft UAV. The controller was designed using Feedback linearization. Since exact input-output linearization fails to linearize the whole system and results in unstable zero dynamics, the authors proposed to linearize the system, approximately, by neglecting the coupling of the model. Simulations indicated the scenario in which a proportional and bias joint type failure of the collective actuator occurred. The results obtained were quite satisfactory.

QI and Han, (June 2008), [111], basically, presented the same research they presented in, [110]. However, a more detailed presentation of the rotorcraft UAV dynamics and characteristics of the sensors were presented.

Miksch, Gambier, and Badreddin, (September 2008), [112], presented a comparison between a model predictive controller, a linear quadratic controller and the pseudo inverse method, (PIM). The controllers were tested in a real-time implementation under several cases of actuators faults such as saturation, freezing and total loss as well as under a structural fault. The Fault Detection and Identification/Diagnosis subsystem was considered to provide accurate information about the faults. An active fault recovery approach based on fault accommodation was pursued. A Three-Tank-System was used to test the algorithms in real-time. However, no information was provided about the

operating system and programming language used in the implementation of the algorithms. The results showed that MPC provided the best performance at the cost of a greater computation expense and an intensive use of the control signals. The fault-tolerant LQ controller displayed an acceptable performance in most of the fault scenarios and the PSM provided the worst results.

4.3. Summary

The literature reviews, presented above, show a snapshot of some of the most recent approaches in the area of fault-tolerant MPC and adaptive MPC. As far as the author is aware, the combination of joint EKF and MPC has not been presented before in the research literature. This provides for the justification of studying applications using the combination of joint EKF and MPC for small-scale helicopter

Chapter 5

Fault-Tolerant Adaptive Model Predictive Control for Flight Systems

5.1. Flight Control Systems

The typical architecture of the low-level flight control systems implemented in the literature have been described as multi-loop, [113], cascaded or nested controllers, [102]. The architecture presented in Figure 49 is a velocity tracking architecture.

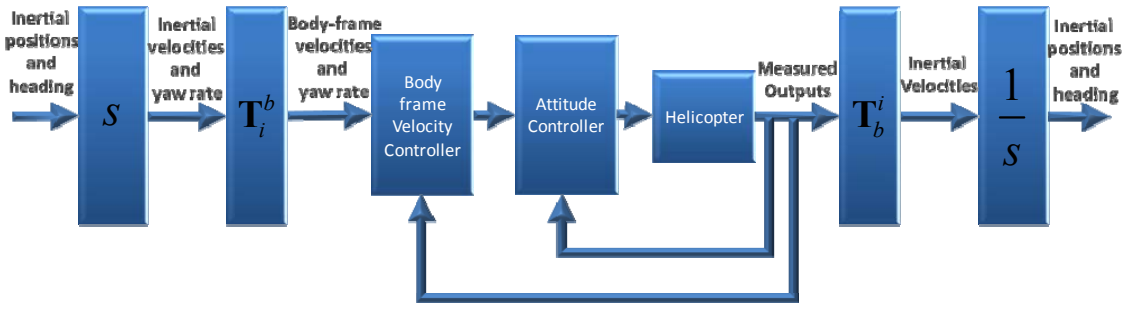


Figure 49: Typical Low-level Flight Control System Architecture

It is possible to include an additional controller loop, in the velocity tracking architecture, to obtain either a low-level, middle or high-level tracking position. A state estimation block and a navigation block, always present in UAV applications, have not been shown to simplify the diagrams. The block $\mathbf{T}_i^b$ is the inertial-frame to body-frame coordinate transformation. This transformation is given by:

$$\mathbf{T}_i^b = \begin{bmatrix} \mathrm{c}\theta\,\mathrm{c}\psi & \mathrm{c}\theta\,\mathrm{s}\psi & -\mathrm{s}\theta \\ \mathrm{s}\phi\,\mathrm{s}\theta\,\mathrm{c}\psi - \mathrm{c}\phi\,\mathrm{s}\psi & \mathrm{s}\phi s\theta s\psi + c\phi\,\mathrm{c}\psi & \mathrm{s}\phi c\theta \\ c\phi s\theta c\psi + s\phi s\psi & c\phi s\theta s\psi - s\phi c\psi & c\phi c\theta \end{bmatrix}, \tag{43}$$

where the Euler angles are ϕ, (roll), θ, (pitch), and Ψ, (yaw),. The c and s parameters represent the cosine and sine functions, respectively. The body-frame to inertial-frame coordinate transformation is given by:

$$\mathbf{T}_b^i = \left(\mathbf{T}_i^b\right)^T. \tag{44}$$

In the developed low-level flight control system, the model predictive control substitutes the body-frame velocity controller and the attitude controller. This approach, presented by the author in, [67], simplifies the design of the low-level flight controller.

As mentioned in chapter 2, it can be stated that parameter estimation and observer-based methods are the most frequently applied techniques for fault detection. The detection of some actuator faults, (LOE), and system or process faults can easily be represented as changes in the **A** and **B** matrices of the linear time-invariant model of the process, which is demonstrated in equations (2) and (3). Parameter estimation is also the technique used for the adaptive control technique, to determine the changes, which are occurring in the plant under control. A Joint Extended Kalman filter represents a straightforward and accurate approach to simultaneously estimate the states and parameters of the system's model.

An inspection of the Self-Tuning Regular block diagram depicted in Figure 6 and the basic block diagram of a fault-tolerant control system depicted on Figure 10 demonstrate the likeness of the developed control system as a form of self-tuning fault-tolerant control system.

The model predictive control technique possesses the intrinsic capability of handling input and states constraints. MPC uses an open-loop optimization to calculate the control signals, which minimize the generic objective function given by:

$$V(k) = \sum_{i=H_w}^{H_p} \left\| \hat{y}(k+i \mid k) - r(k+i \mid) \right\|_{Q(i)}^{3} + \sum_{i=0}^{H_u-1} \left\| \Delta\hat{u}(k+i \mid k) \right\|_{R(i)}^{2} \quad (45)$$

where

- $\hat{y}(k+i \mid k)$ is the predicted output,
- $r(k+i \mid k)$ is the predicted reference trajectory,
- $\Delta\hat{u}(k+i \mid k)$ is the predicted changes of the control signal,
- Hp is the prediction horizon,
- Hw is the window parameter,
- Hu is the control horizon,
- $Q(i)$ are the weighting matrixes applied to the predicted error during the prediction horizon,
- R(i) are the weighting matrices applied to the control moves during the control horizon.

MPC calculates the optimal control signal, which minimizes the objective function for the given parameter and the current internal model. Therefore, MPC pursues optimality even when there are changes or updates of the internal model.

The Fault-Tolerant Adaptive Model Predictive Control (FTA-MPC) combines the advantages of adaptive control techniques with fault-tolerant control techniques by inclusion of a Joint Kalman filter as parameter estimator. The adaptation is performed

each time the parameters of the internal model of the MPC are updated. Figure 50 shows the block diagram of the Fault-Tolerant Adaptive Model Predictive Controller.

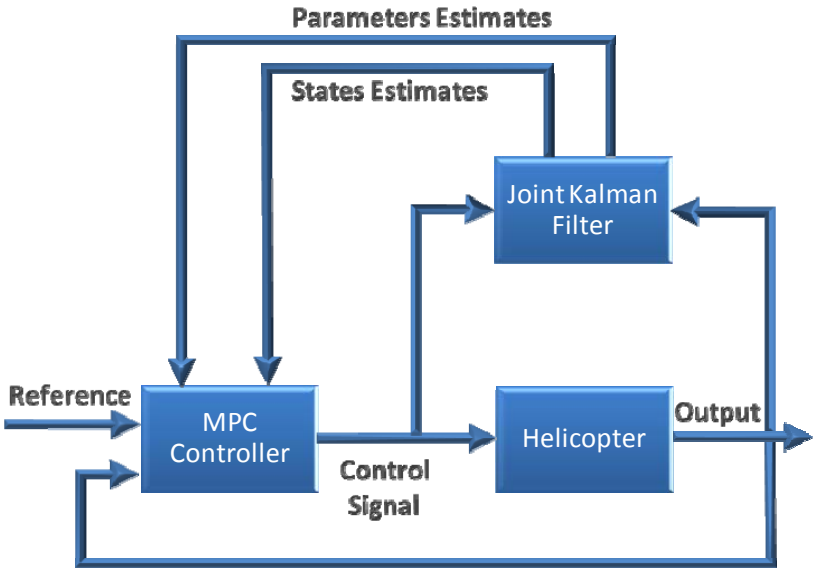


Figure 50: Generic Block Diagram of the Fault-Tolerant Adaptive Model Predictive Controller

A more detailed block diagram is presented in Figure 51.

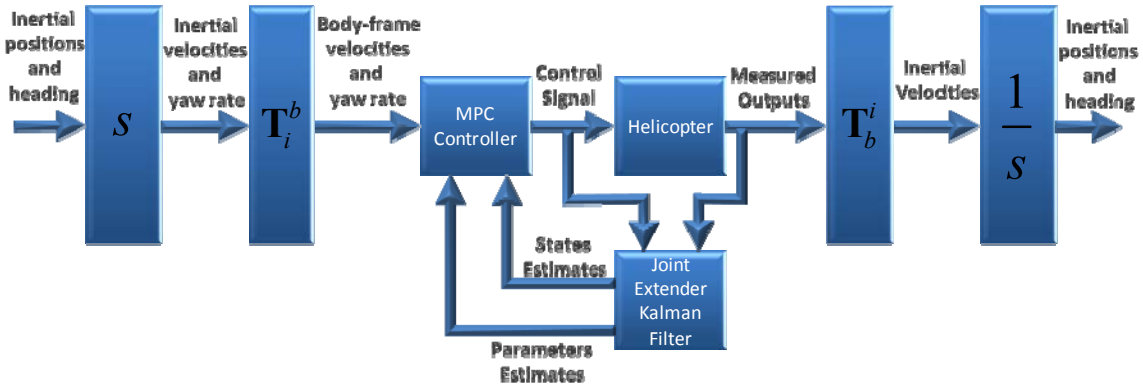


Figure 51: FTA-MPC Flight Control System

The measured output signals are the body-frame linear velocities, $[u, v, w]^T$, and the angular velocities, $[p, q, r]^T$. The control signals are the cyclic longitudinal, $\delta_{lon,}$ the cyclic lateral, δ_{lat}, the collective, δ_{col}, and the pedal, δ_{ped}. The controls signals are constrained to the range -1 to +1. The MPC set point signals are $u_{set\ point}$ $v_{set\ point}$, $w_{set\ point}$ *and* $r_{set\ point}$. The parameters to be estimated are X_u, Z_{col}, N_{col}, A_c and $A_{lon.}$ The waypoints are given as inertial positions x, y, z and the heading as a function of the time t.

The behavior of a simulated flight system utilizing a FTA-MPC is presented in the next chapter under several fault case studies. A stability test and robustness test under nominal conditions is also presented.

Chapter 6

Results

In this chapter several simulations are presented, which were performed to test different aspects of the developed control configuration. The simulations consisted of a performance comparison, a stability test, a passive fault, (robustness), test and several fault case scenarios.

6.1. Performance Comparison

A comparison of the robustness of the standard MPC and an H_∞ loop-shaping controller, previously developed, [68], was realized. The nominal or non-faulty case is presented. Figure 52 to Figure 56 presents the response of the system, to the set points of signals in the body frame and in the inertial frame.

Figure 52 displays the responses of the helicopter to the set points of the longitudinal velocity, *u*, and the lateral velocity, *v*. These velocities are in the body-frame.

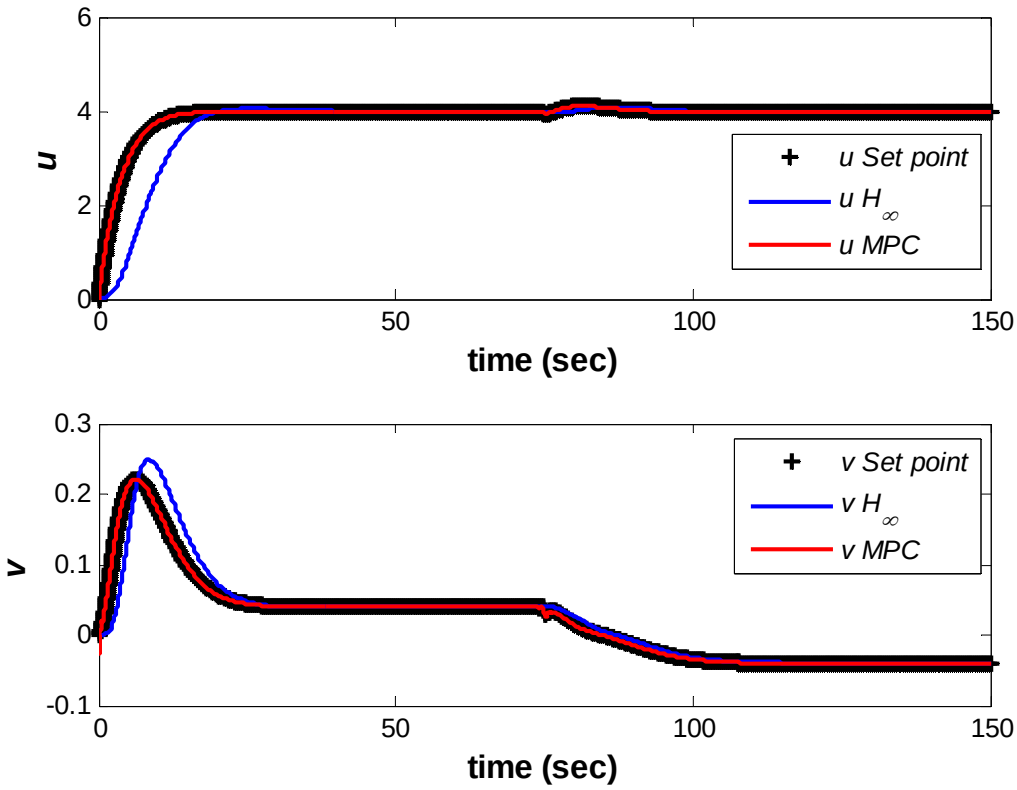


Figure 52: *u* and *v* Response of the System in the Nominal Case, No Fault

Figure 53 displays the responses of the helicopter to the set point of the body-frame vertical velocity, *w*, and the yaw rate, *r*.

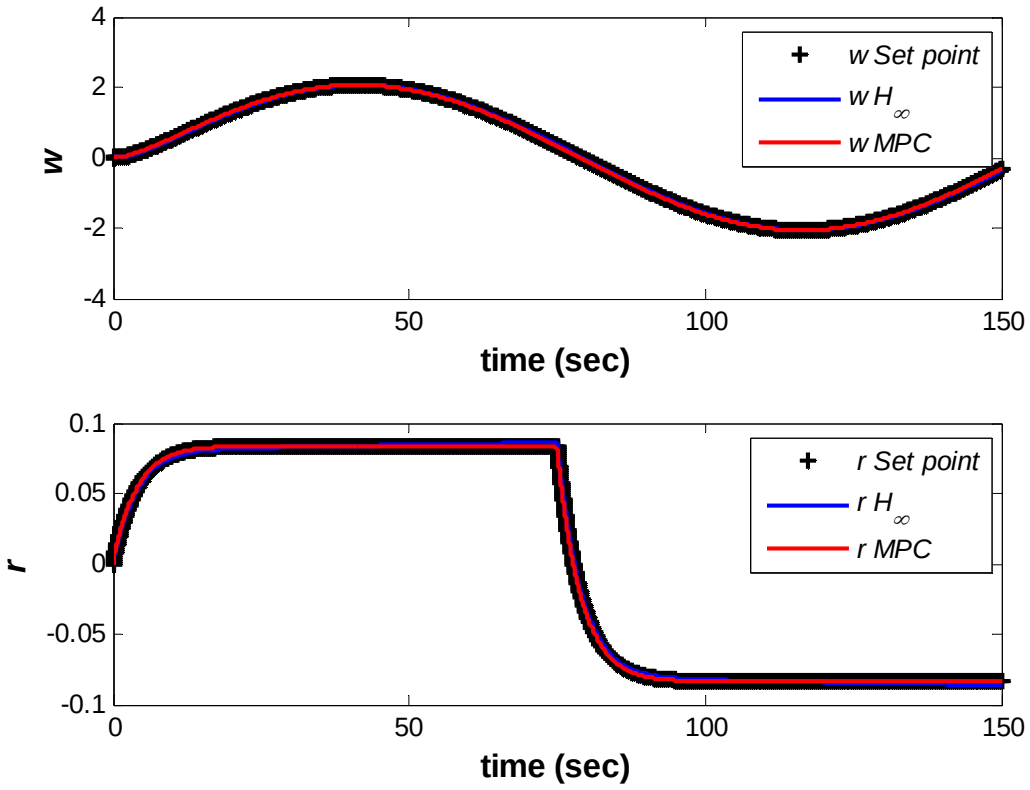


Figure 53: *w* and *r* Response of the System in the Nominal Case, (No Fault)

Figure 54 displays the inertial trajectory followed by the H_∞ controller in response to the set point trajectory.

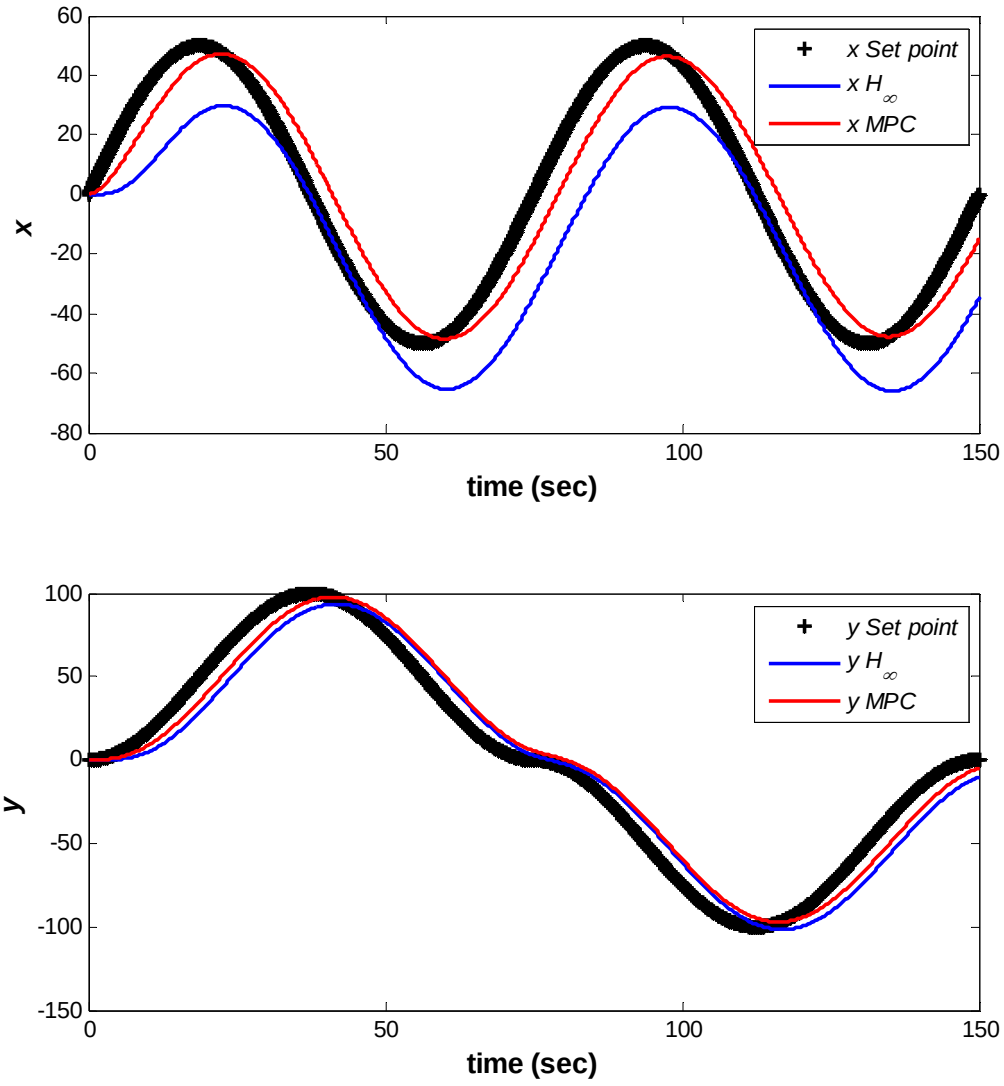


Figure 54: x and y Response of the System in the Nominal Case, (No Fault)

Figure 55 displays the inertial trajectory followed by the developed MPC in response to the set point trajectory.

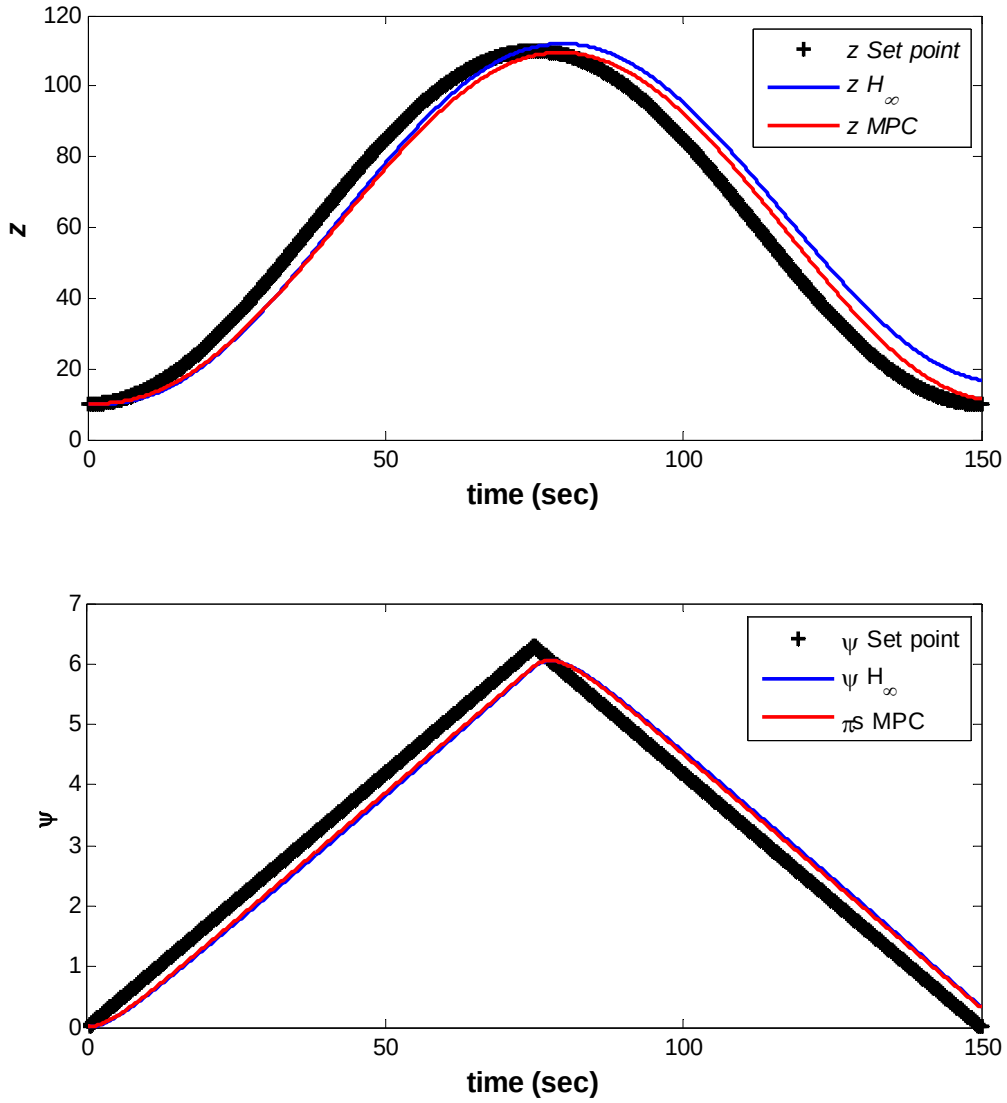


Figure 55: z and Ψ Responses of the System in the Nominal Case, (No Fault)

Figure 56 presents the 3D representation of the inertial trajectories.

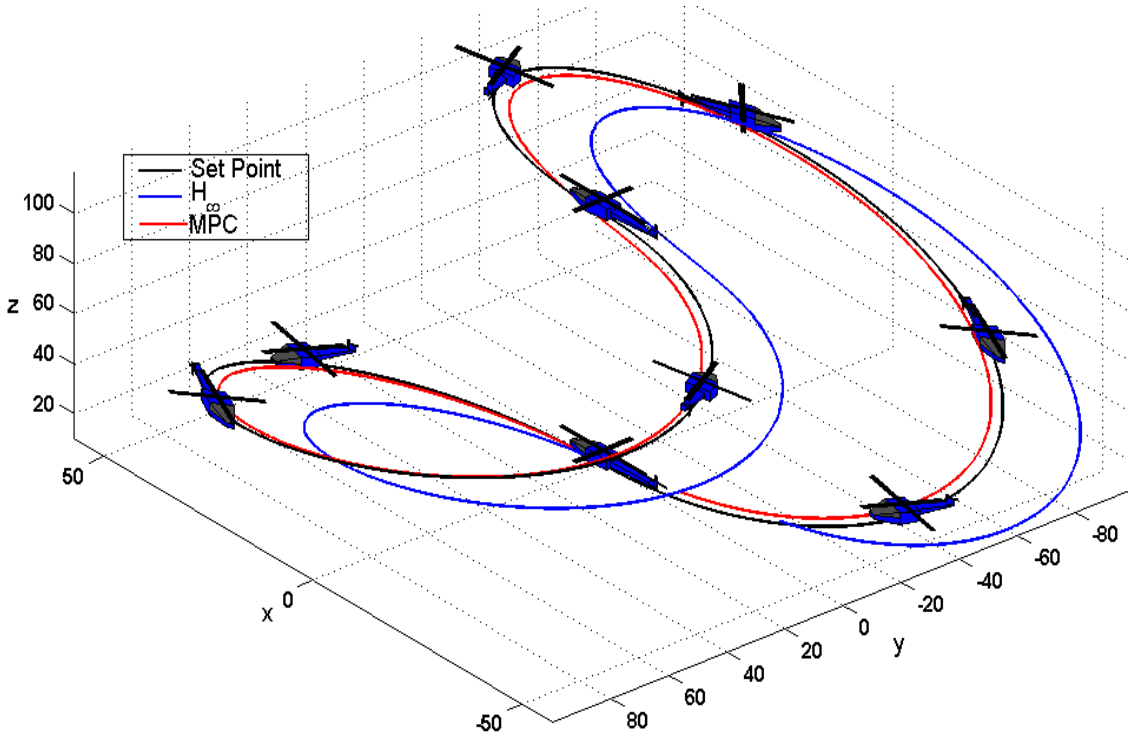


Figure 56: 3D Plot of the Response of the System to the Double Circle with Varying Altitude Trajectory in the Nominal Case, (No Fault)

The data presented in Figures 52 through 56, clearly demonstrate that the standard MPC outperforms the H_∞.

6.2. Stability Test

In order to test the nominal stability of the developed system, some initial values were assigned to the output variables and the system was simulated to verify its capability to bring the states to zero. Several sets of the initial values, of the state variables, were

used to test the stability of the designed control system. Figure 57 presents the results for the initial states/outputs given by:

$$y0=[6,-1,-1,-1,-1,-1,-1].$$

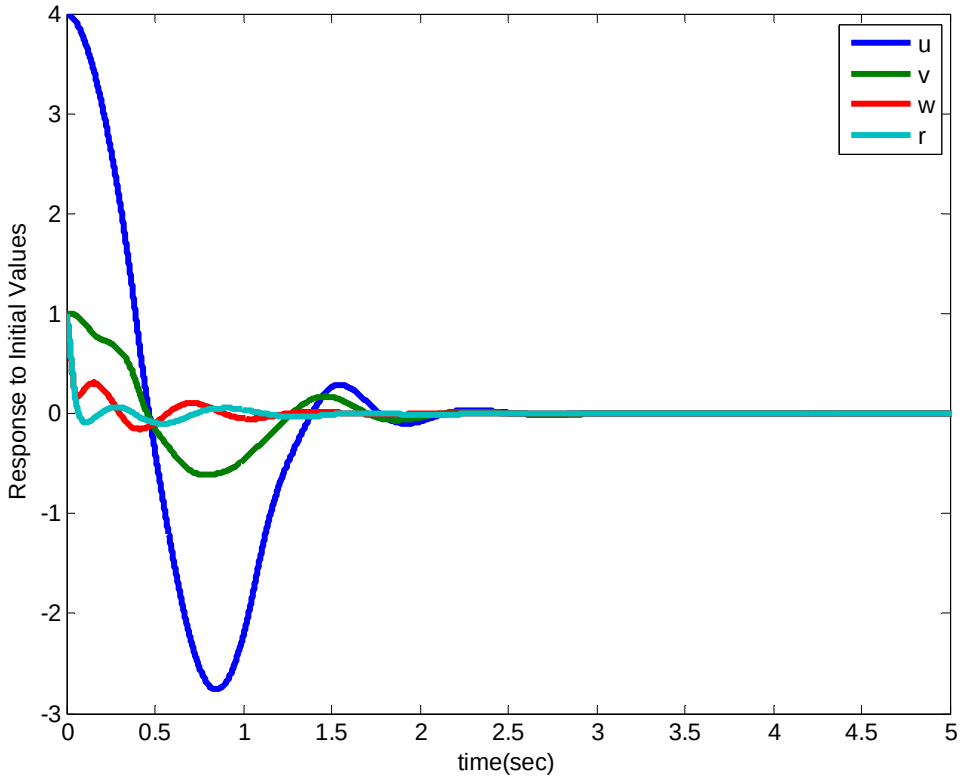


Figure 57: Stability Test of the System under Nominal Conditions with the Initial States/Outputs Given by y0 = [6,-1,-1,-1,-1,-1,-1]

Figure 58 presents the results with the with initial values given by

$$y0 = [6,-1,-1,-1,-1,-1].$$

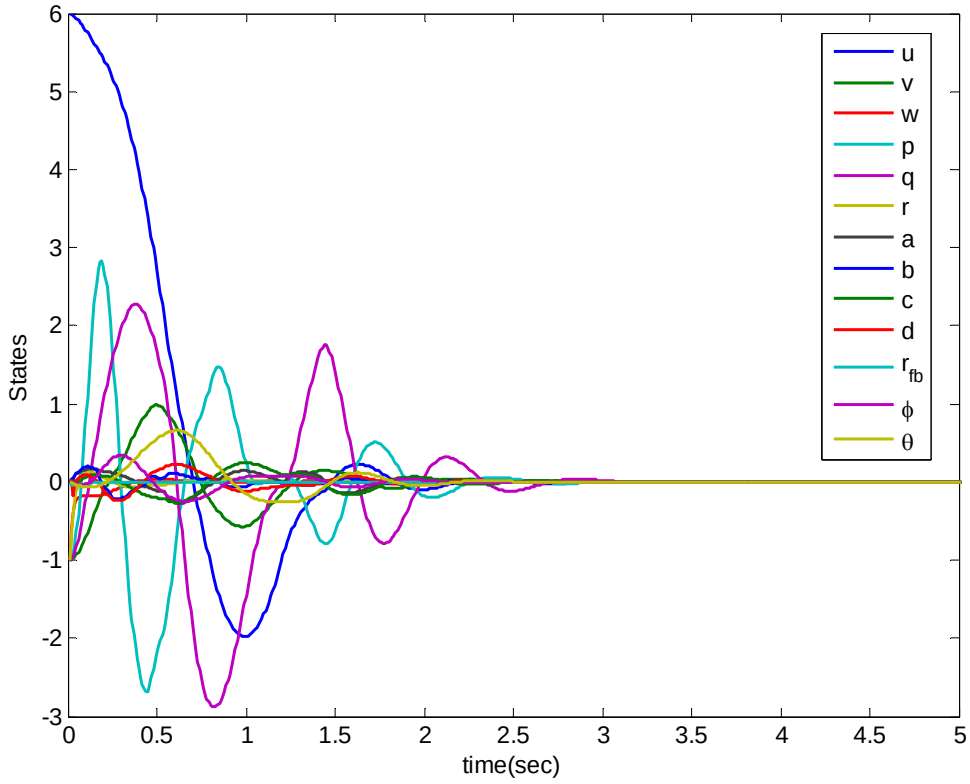


Figure 58: Stability Test of the System under Nominal Conditions with Initial Values Given by y0 = [6,-1,-1,-1,-1,-1]

Figure 59 presents the results with initial values given by

$$y0 = [6,-6, 5,-5, 4,-4].$$

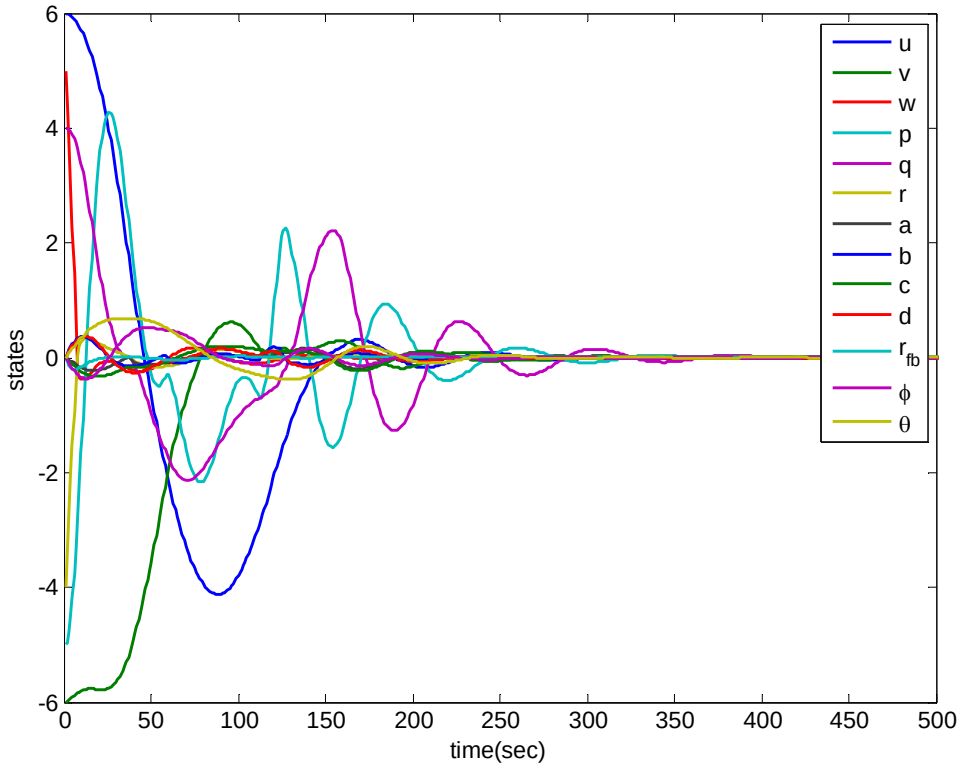


Figure 59: Stability Test of the System under Nominal Conditions with Initial Values Given by y0 = [6,-6, 5,-5, 4,-4]

6.3. Passive Fault Tolerance, (Robustness)

The case when a fault, such as changes in the nominal value of $X_{u,}$ occurs in the helicopter is considered. The value of the parameter, $X_{u,}$ was changed from its nominal value of -0.0505 to the faulty value of +0.3. The change and the manner in which it was handled by MPC and H_∞ are presented in Figure 60 to Figure 64. The data clearly

indicate that the standard MPC outperformed the H_∞. Figure 60 presents the response for the lateral velocity parameters *u* and *v*.

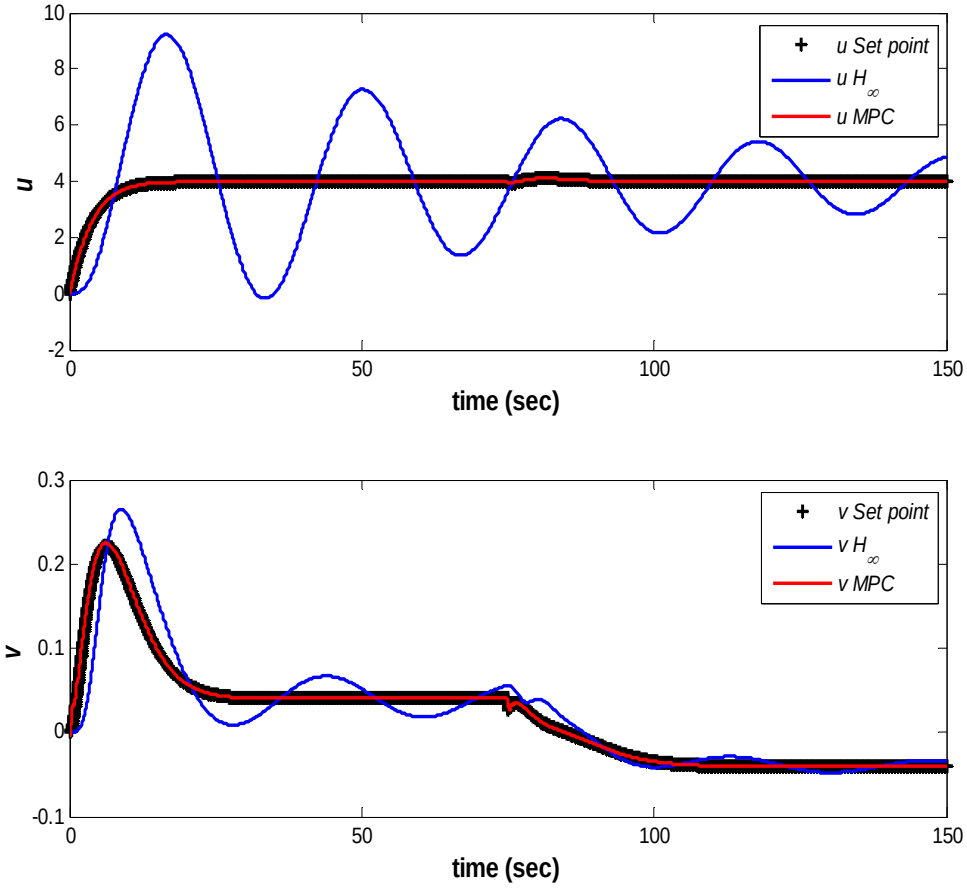


Figure 60: *u* and *v* Responses of the System When a Fault Occurs, (X_u = 0.3)

Figure 61 presents the responses for the rotational velocity parameters *w* and *r*.

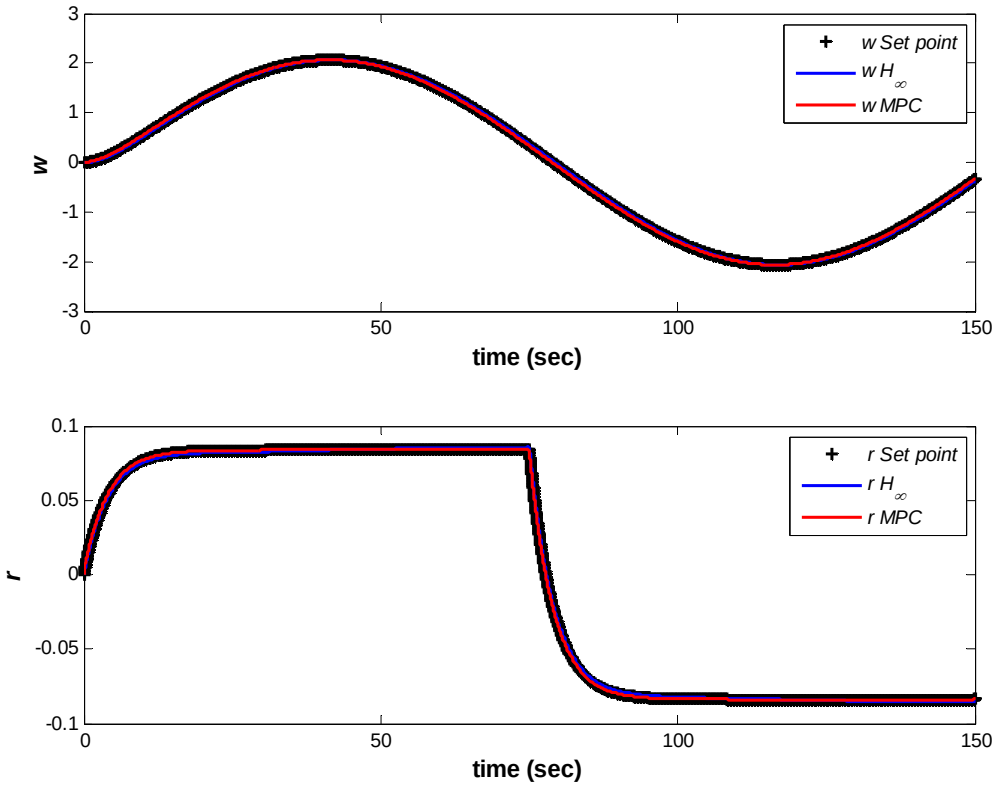


Figure 61: *w* and *r* Responses of the System When a Fault Occurs, ($X_u = 0.3$)

Figure 62 presents the responses for the translational parameters *x* and *y*.

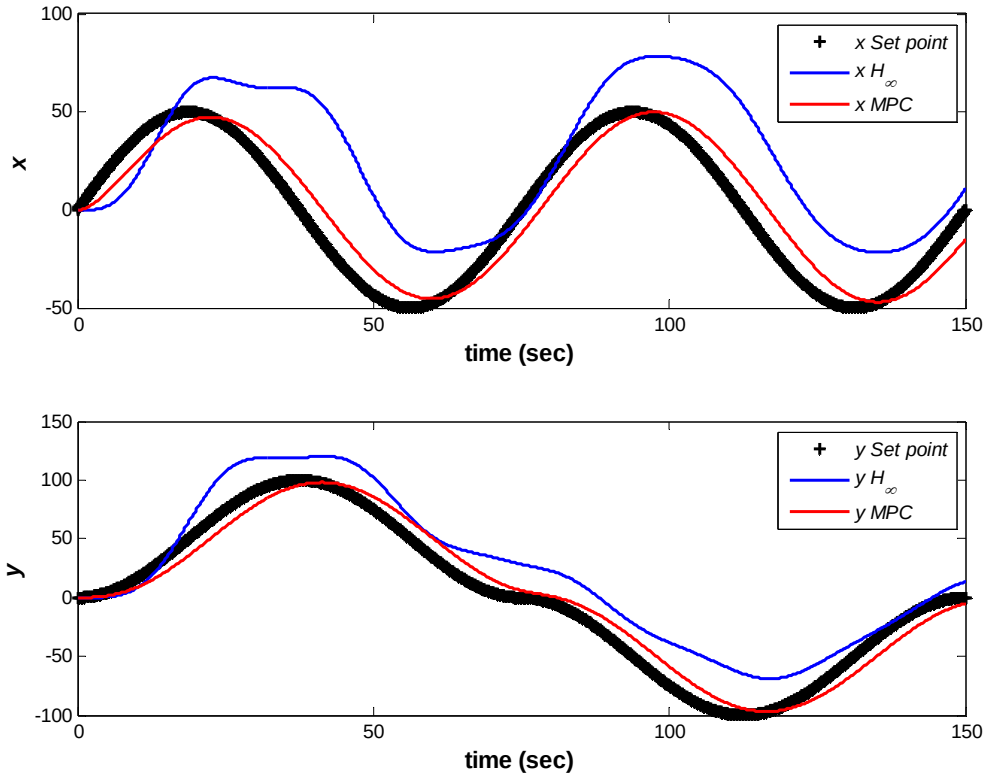


Figure 62: *x* and *y* Responses of the System When a Fault Occurs, ($X_u = 0.3$)

Figure 63 presents the responses of the vertical translation parameter, ‚*z*, and the yaw angle, Y, parameter.

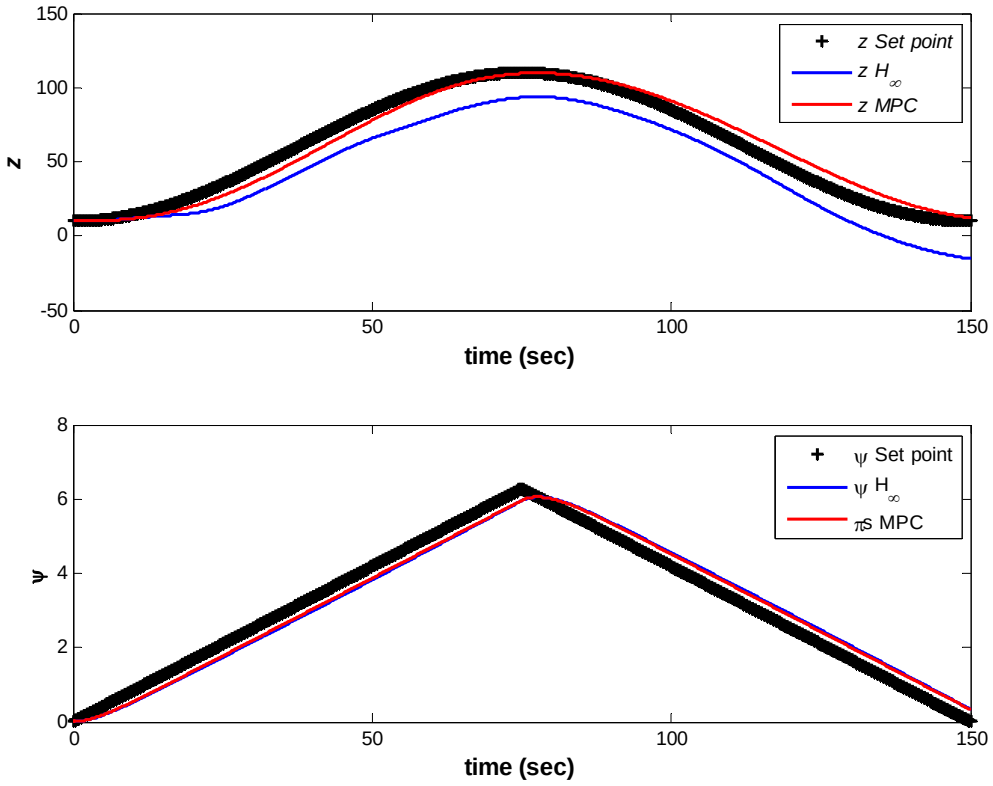


Figure 63: *z* and Ψ Responses of the System When a Fault Occurs, (X_u = 0.3)

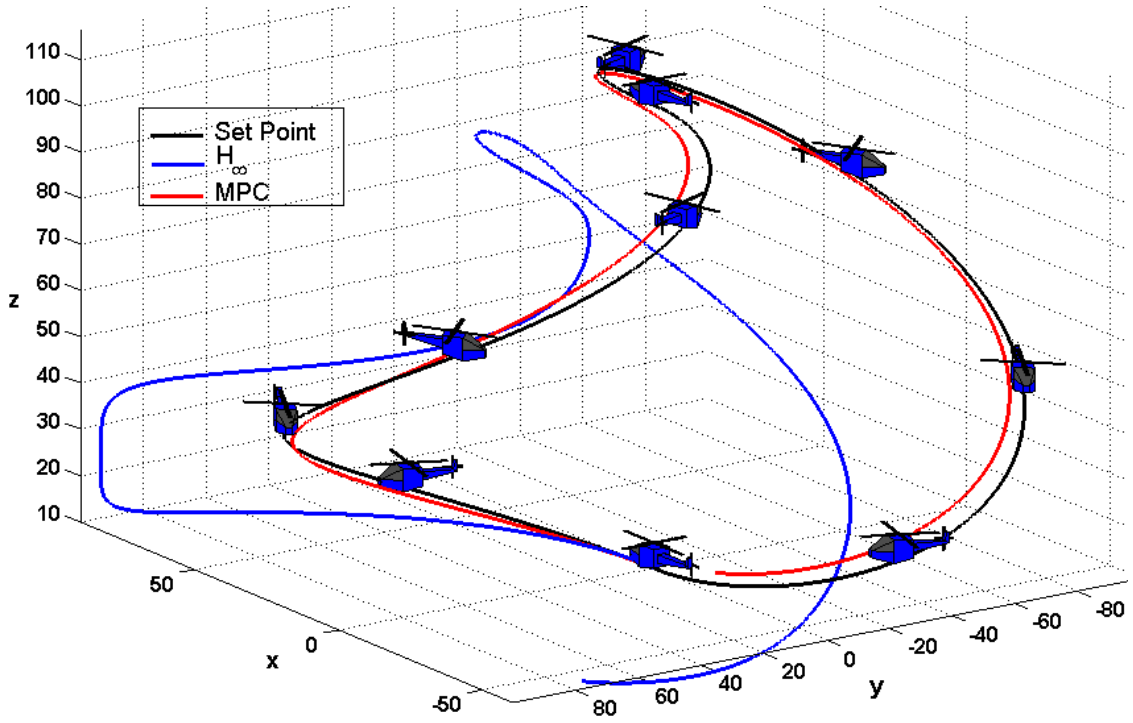


Figure 64: 3D Plot of the Response of the System to the Double Circle with Varying Altitude Trajectory when the Parameter Xu was Equal to 0.3, (Fault)

Figure 64 shows that the helicopter crashes at the end of the trajectory when it was being controlled by the H_∞ controller. The standard MPC was able to maintain stability and a performance close to the nominal case.

6.4. Fault-Tolerant Model Predictive Control

Several fault scenarios are presented in the next sections.

6.4.1. Fault Case 1

In this case, the value of the X_u parameter was changed from -0.0505 to 3. The H_∞ loop shaping controller was not able to maintain stability for any values of X_u greater

than 0.3. Hence, it was not used for further comparison with the Fault-Tolerant MPC. Figure 65 presents the response of the estimated value for the X_u parameter.

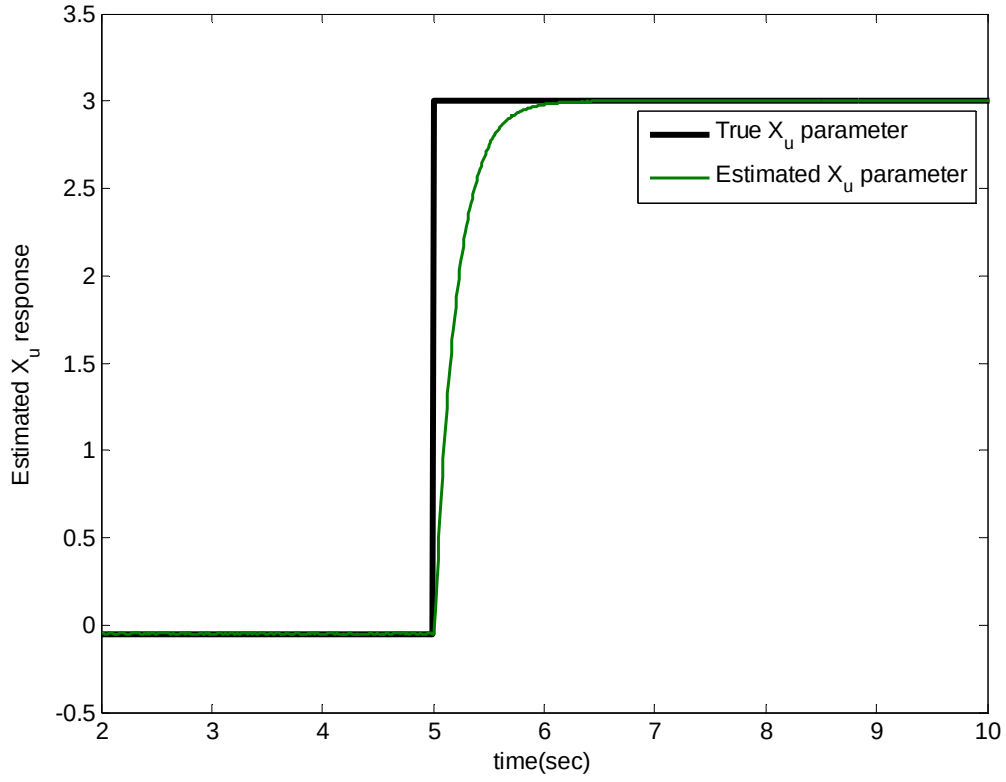


Figure 65: Response of the Estimated X_u Parameter: Fault Case 1

Figure 66 presents the covariance of the X_u parameter.

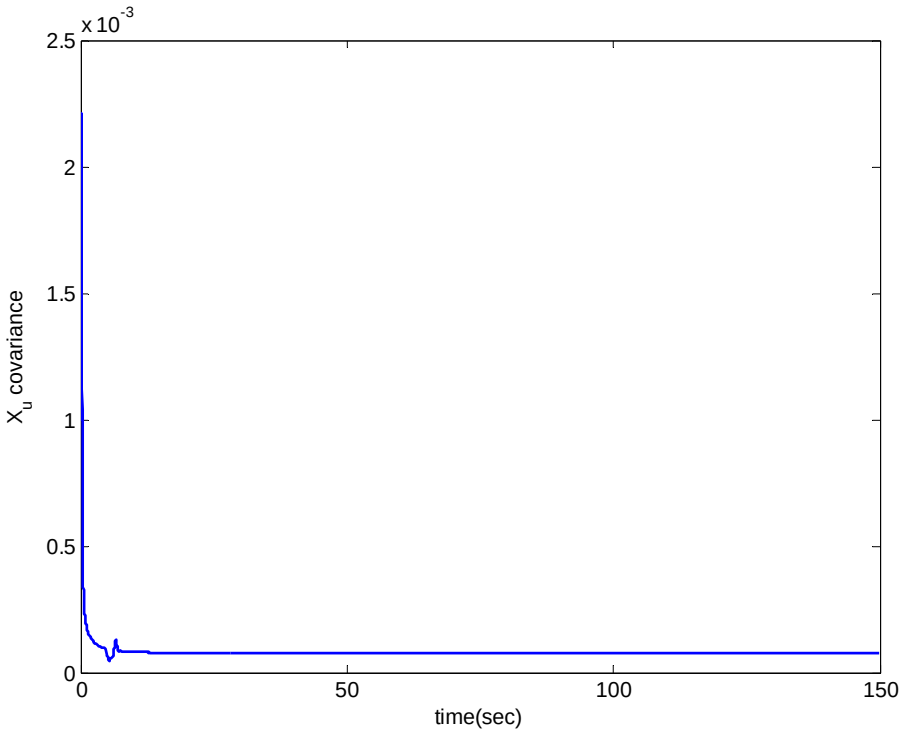


Figure 66: X_u Covariance: Fault Case 1

Figure 67 presents the response of the control system for the *u* translational velocity parameter when the fault was applied at 5 sec.

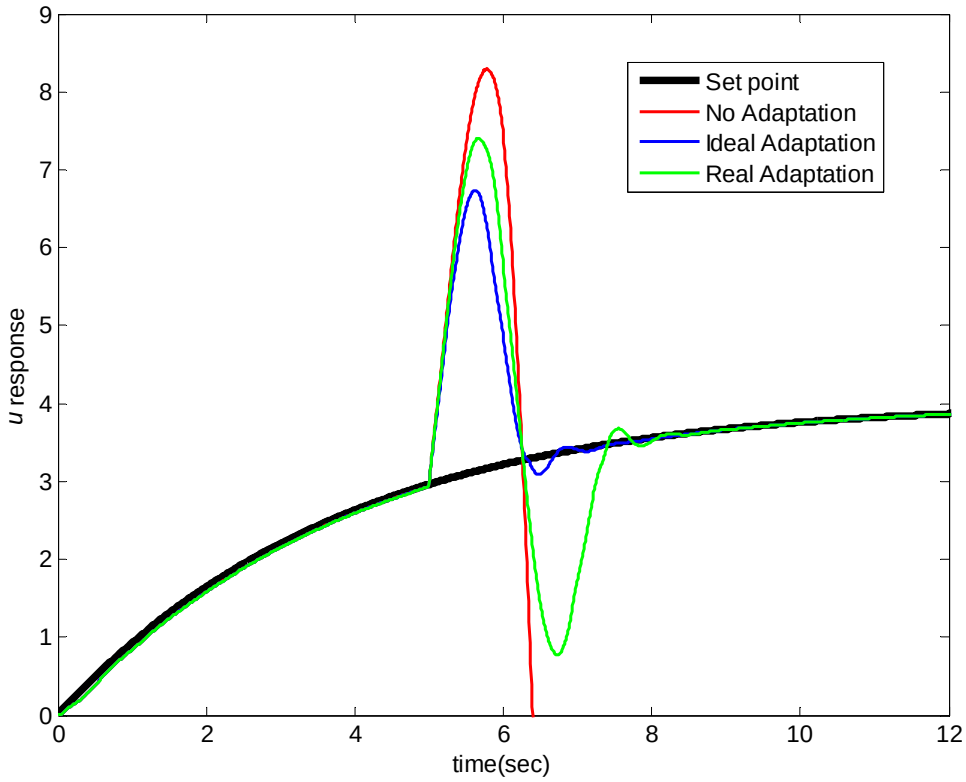


Figure 67: *u* Response of the Control System: Fault Case 1

The maximum fault magnitude, which could be controlled, was a step of 3.0505. The outputs were disturbed with Gaussian noise.

Figure 68 presents the control system response with respect to the *x* translational parameter.

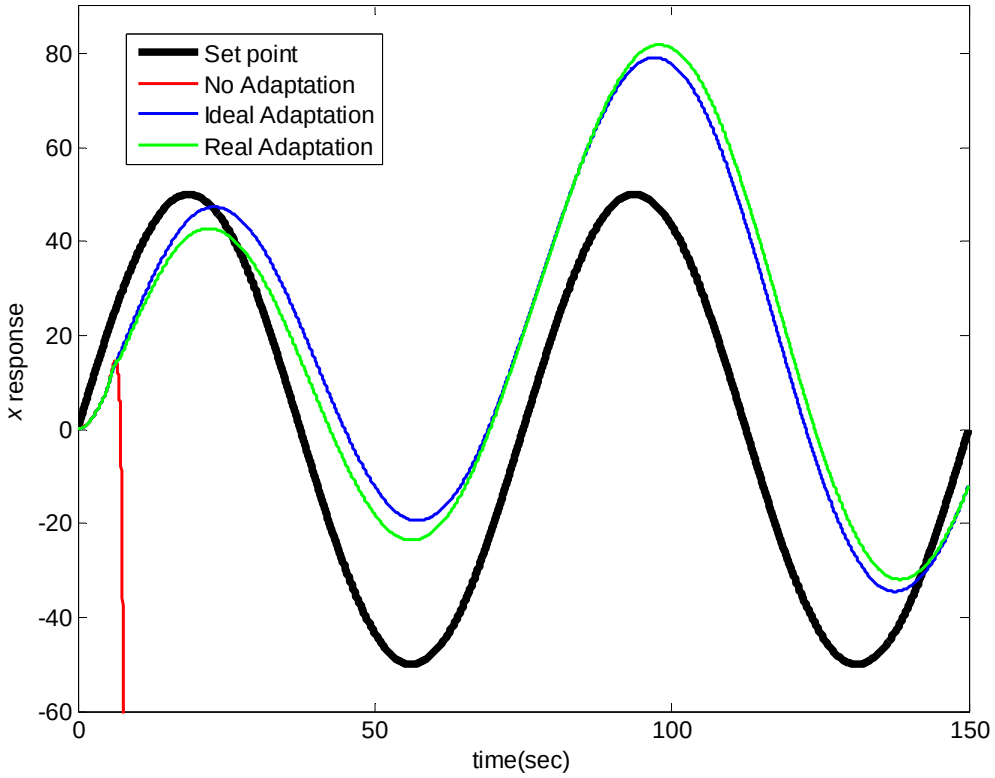


Figure 68: *x* Response of the Control System: Fault Case 1

Figure 69 presents the control system response with respect to the *y* translational parameter.

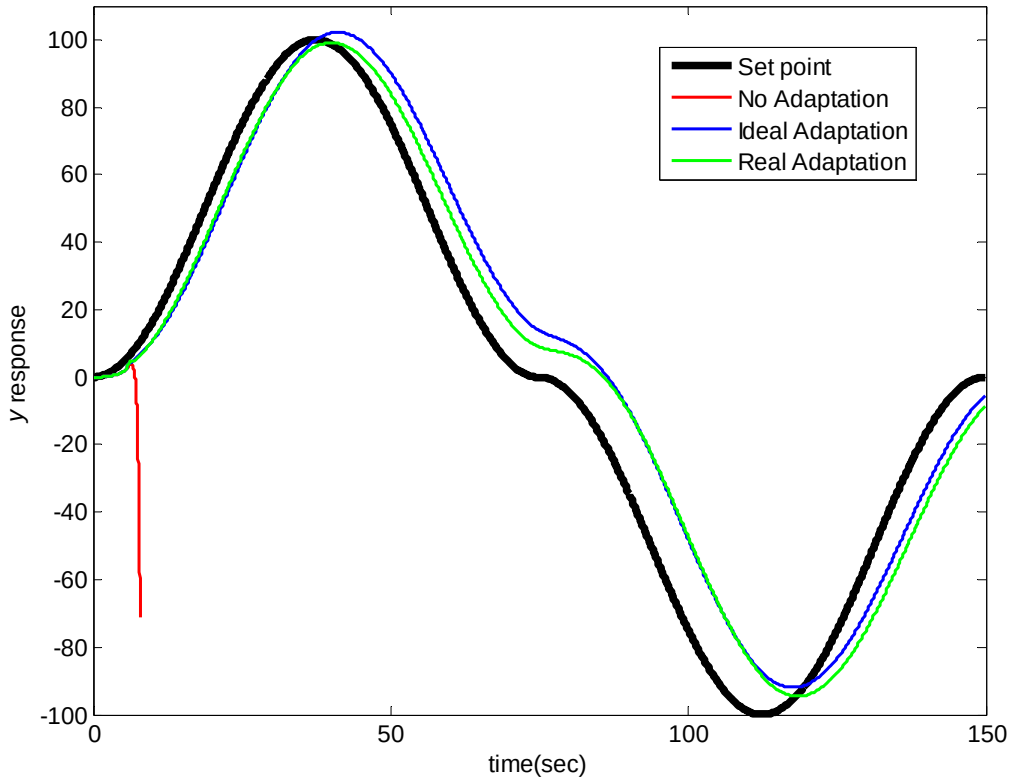


Figure 69: *y* Response of the Control System: Fault Case 1

Figure 70 presents the control system response with respect to the *z* translational parameter.

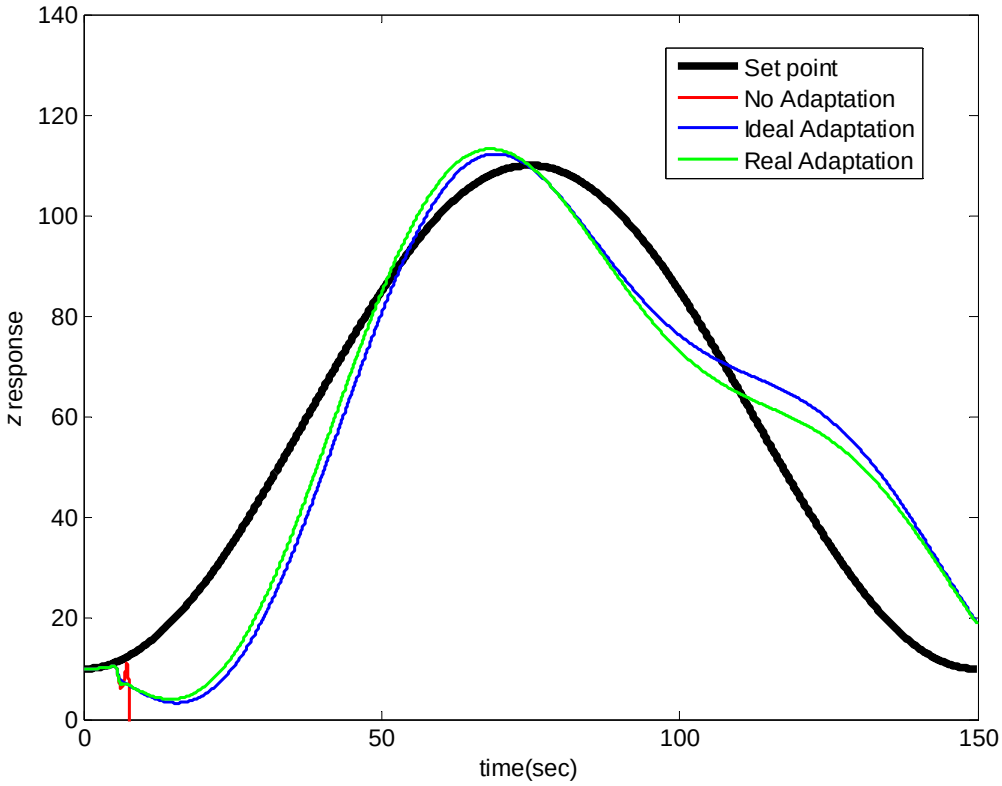


Figure 70: *z* Response of the Control System: Fault Case 1

Figure 71 presents the control system response with respect to the yaw angle, (Ψ), parameter.

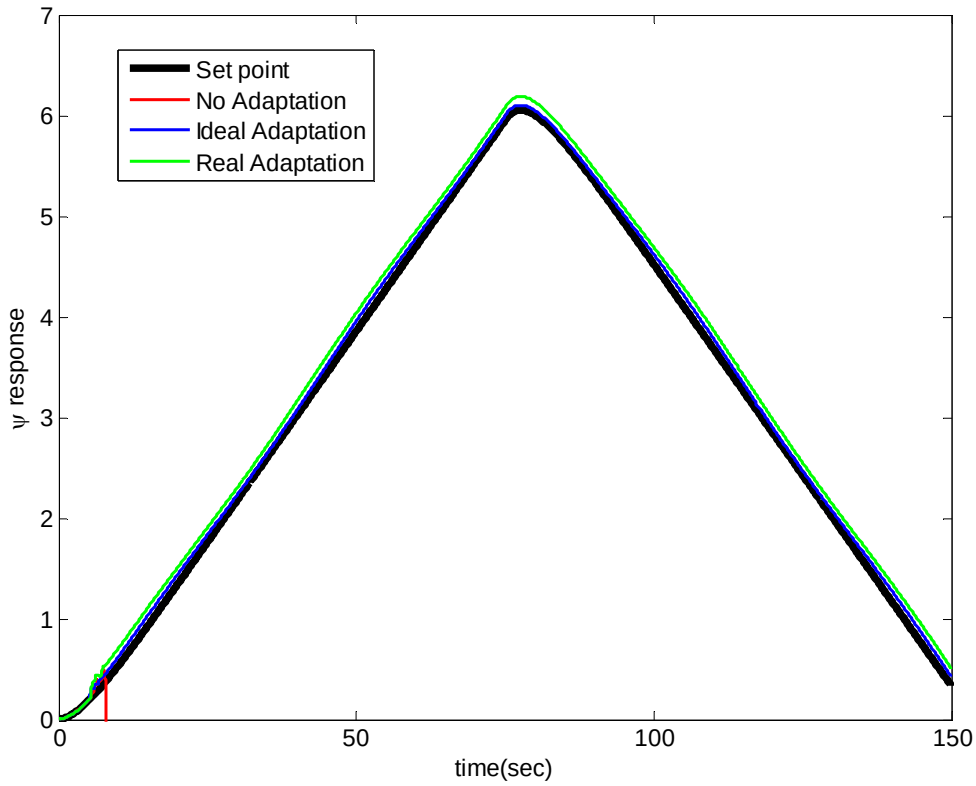


Figure 71: Ψ Response of the Control System: Fault Case 1

Figure 72 presents the 3D trajectory response of the control system.

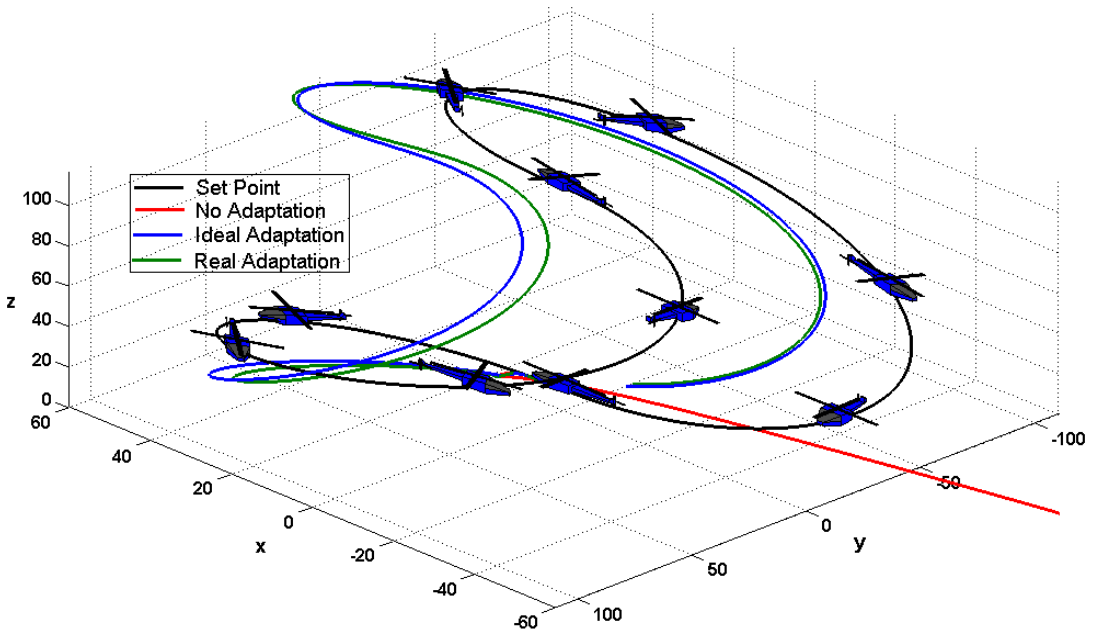


Figure 72: 3D Response of the Control System: Fault Case 1

6.4.2. Fault Case 2

In this case, the fault, which was a change in the value of the parameter X_u, was applied at 20 sec. The outputs were disturbed with Gaussian noise. The maximum fault magnitude, which could be controlled, was a step of 2.5505.

Figure 73 presents the response of the estimated value for the X_u parameter.

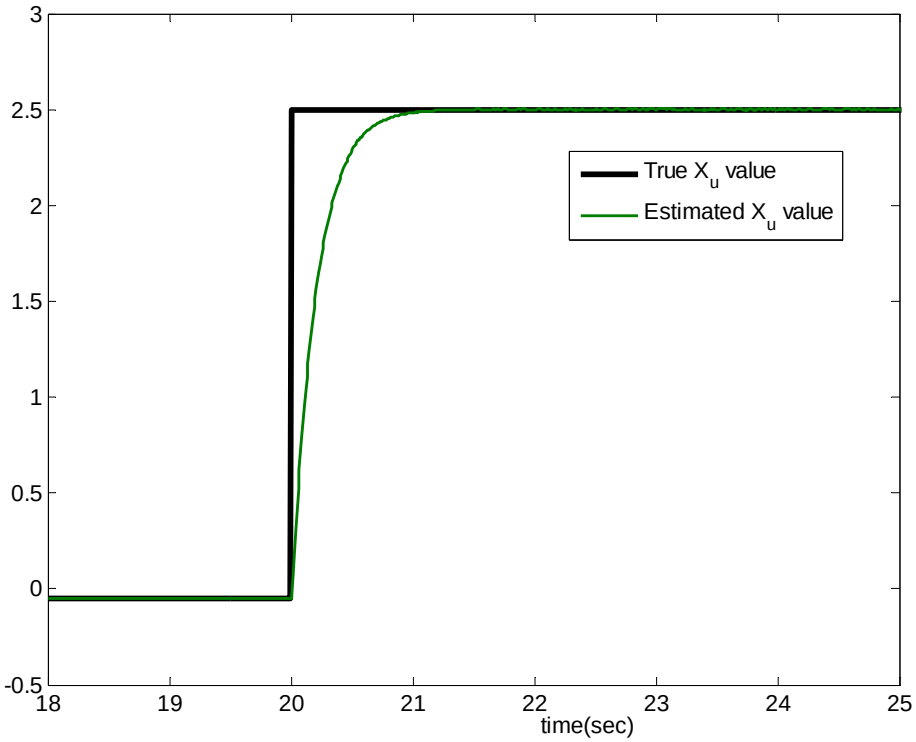


Figure 73: Response of the Estimated X_u Parameter: Fault Case 2

Figure 74 presents the covariance of the X_u parameter.

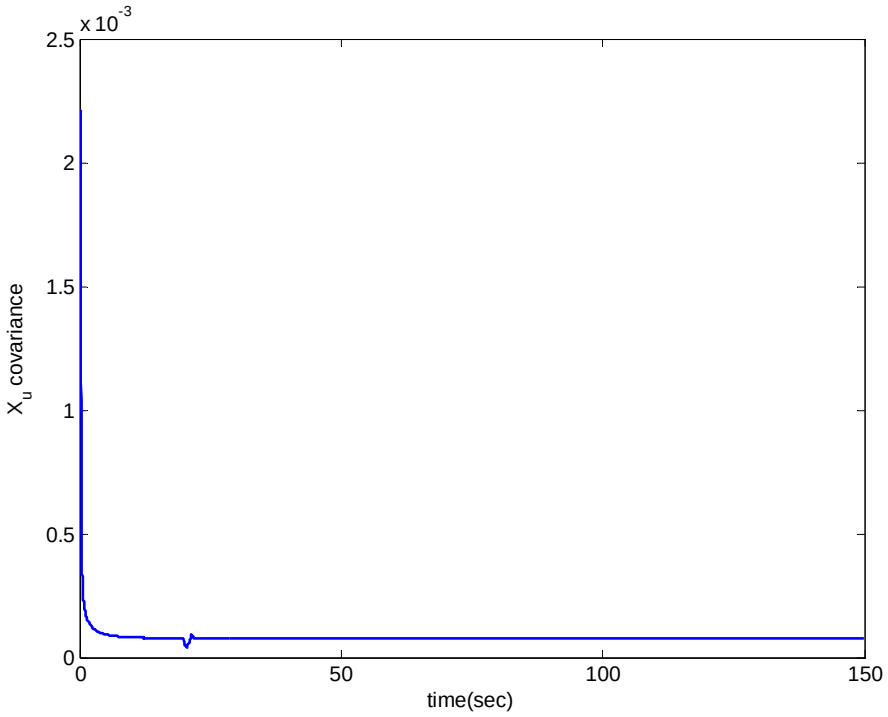


Figure 74: X_u Covariance: Fault Case 2

This case was determined to be the worst-case scenario for the occurrence of a change of the X_u parameter.

Figure 75 presents the response of the control system for the *u* translational velocity parameter when the fault was applied at 20 sec.

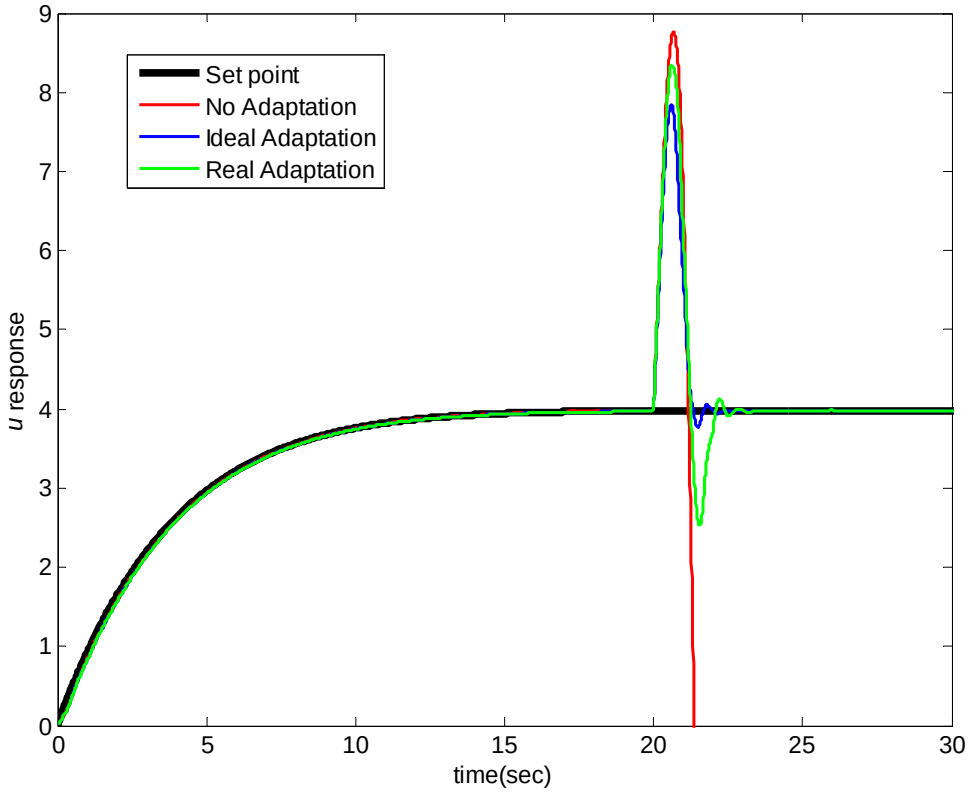


Figure 75: *u* Response of the Control System: Fault Case 2

Figure 76 presents the control system response with respect to the *x* translational parameter.

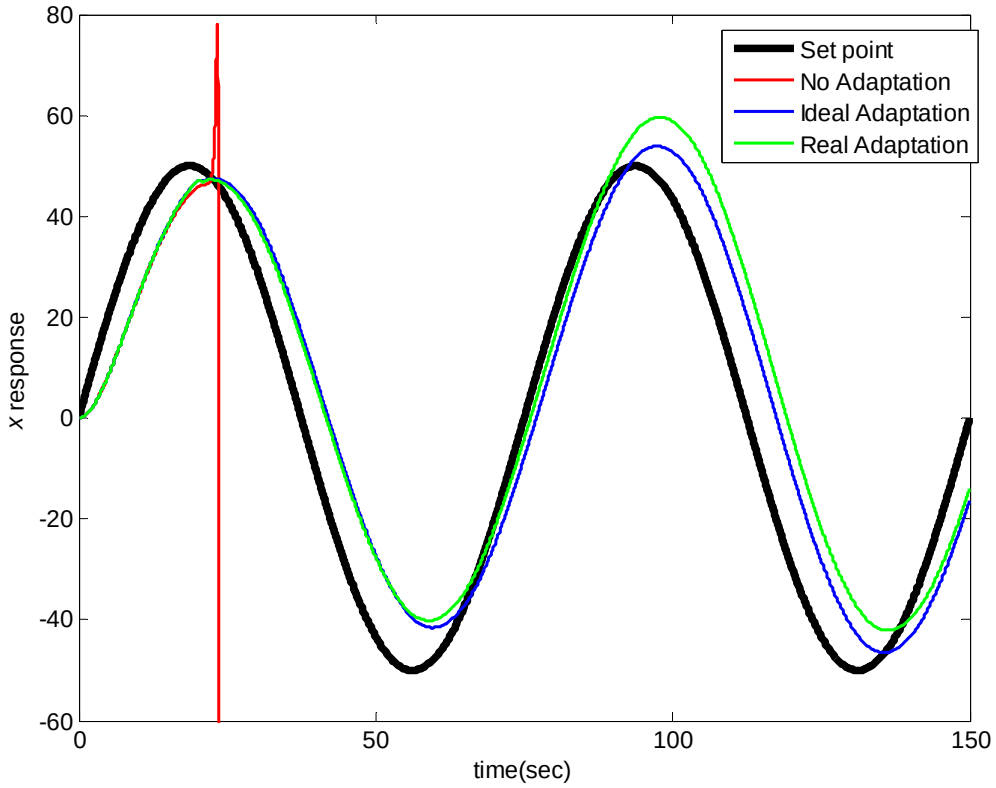


Figure 76: *x* Response of the Control System: Fault Case 2

Figure 77 presents the control system response with respect to the *y* translational parameter.

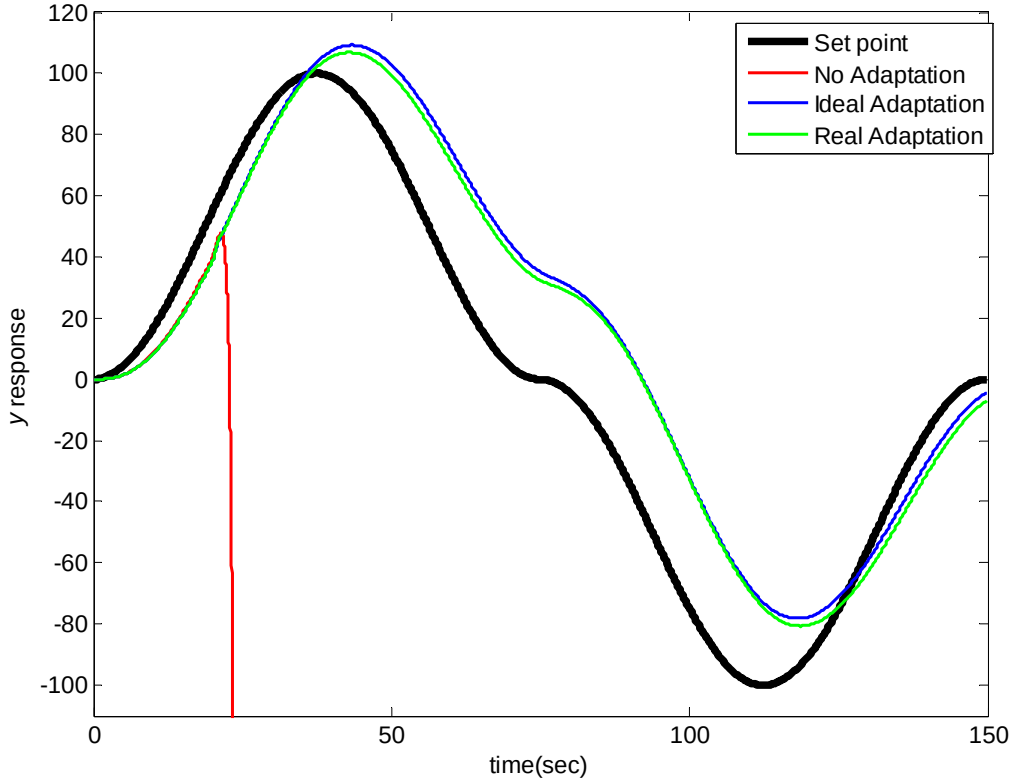


Figure 77: *y* Response of the Control System: Fault Case 2

Figure 78 presents the control system response with respect to the *z* translational parameter.

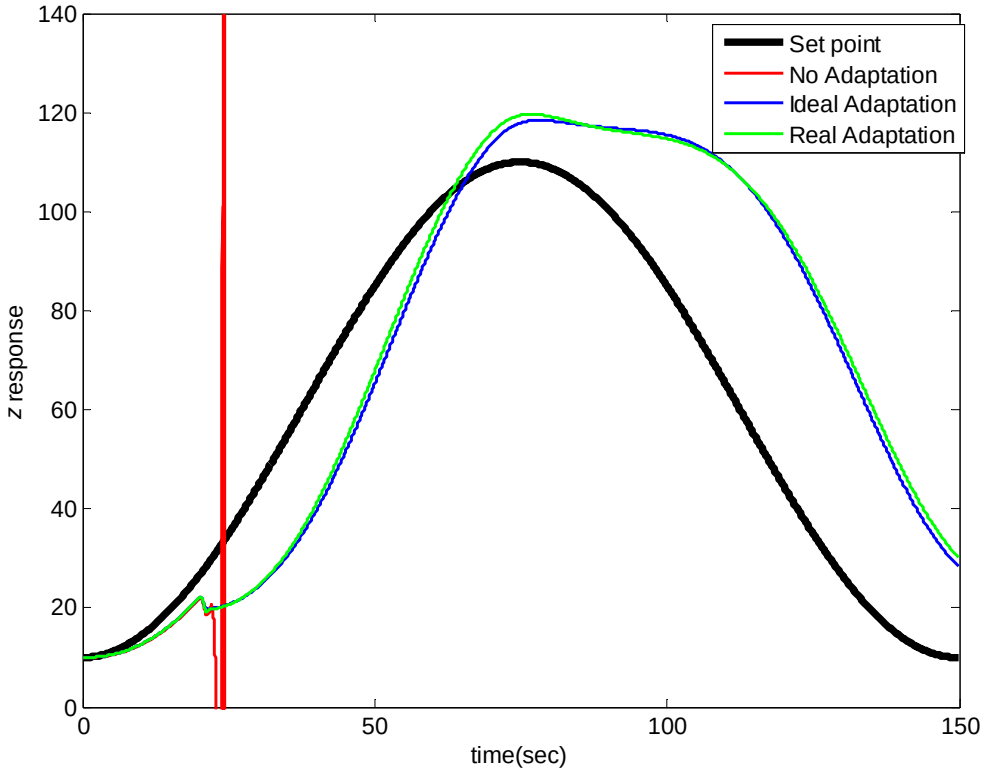


Figure 78: *z* Response of the Control System: Fault Case 2

Figure 79 presents the control system response with respect to the yaw angle, (Ψ), parameter.

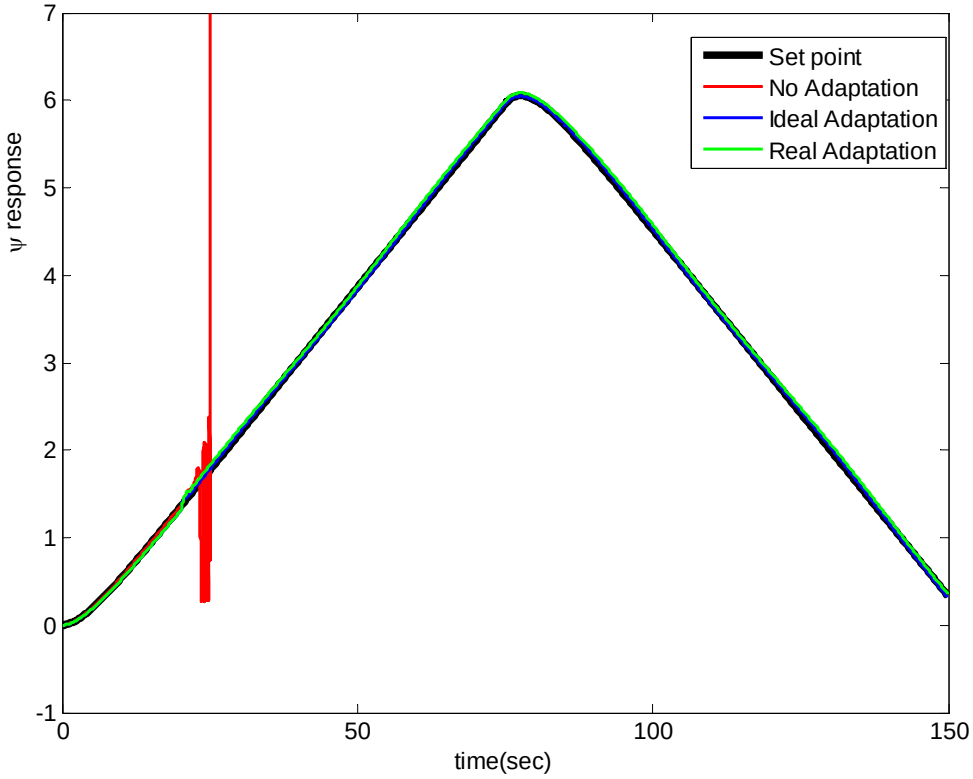


Figure 79: Ψ Response of the Control System: Fault Case 2

Figure 80 presents the 3D trajectory response of the control system.

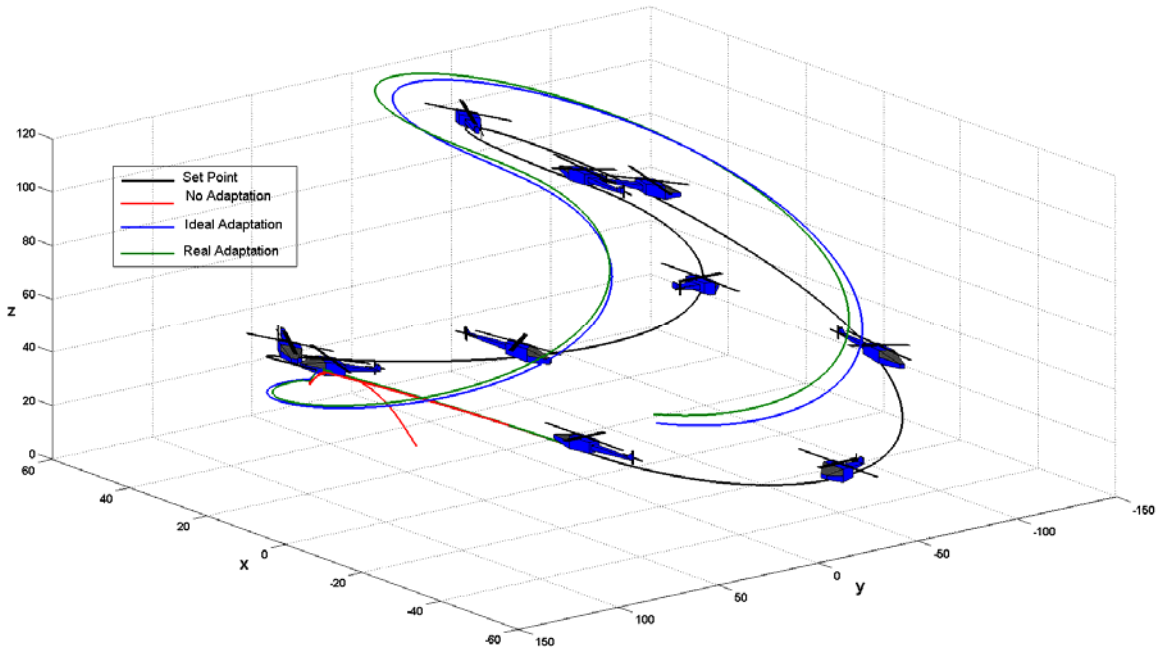


Figure 80: 3D Response of the Control System: Fault Case 2

6.4.3. Fault Case 3: Bell Mixer

The Bell mixer is a mechanical mixer between the stabilizer bar and the main blade pitch control. The action of the mixer is to impose a command on the main blade pitch, which is proportional to the flapping magnitude of the stabilizer bar, [102].

A change in the value of the parameter A_c, was assumed to represent an indication of a fault in the Bell mixer. This Bell mixer fault was applied at 20 sec. The maximum magnitude of the fault, which could be controlled, was a step of 4.356. The outputs were disturbed with Gaussian noise.

Figure 81 presents the response of the estimated value for the A_c parameter.

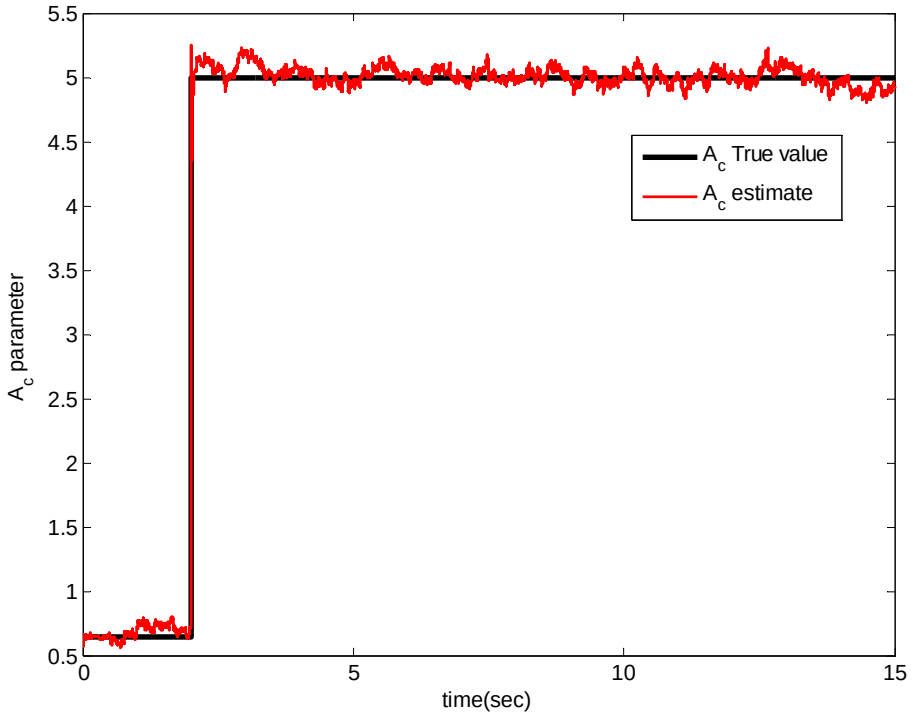


Figure 81: Response of the Estimated A_c Parameter: Bell Mixer Fault

Figure 82 presents the covariance of the A_c parameter.

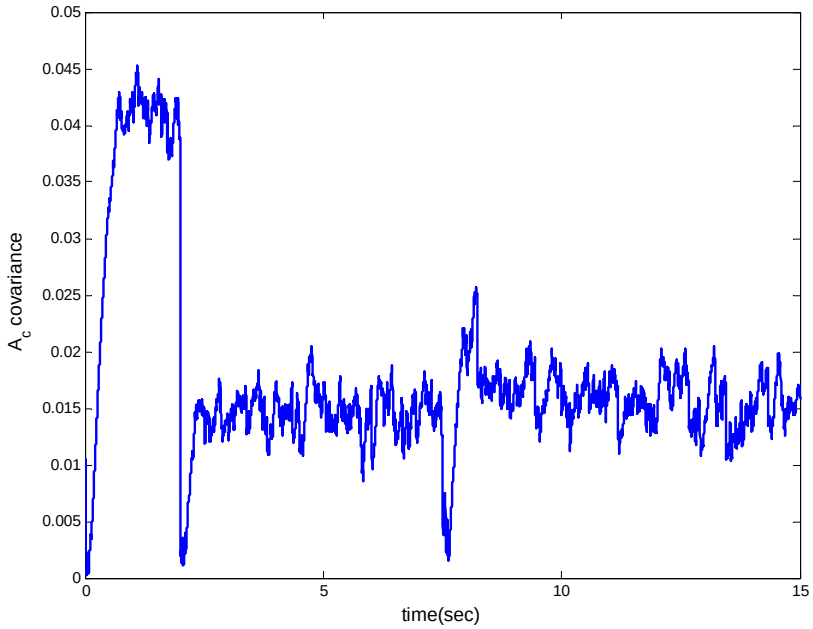


Figure 82: A_c Covariance: Bell Mixer Fault

Figure 83 presents the response of the control system for the *u* translational velocity parameter when the fault was applied at 20 sec.

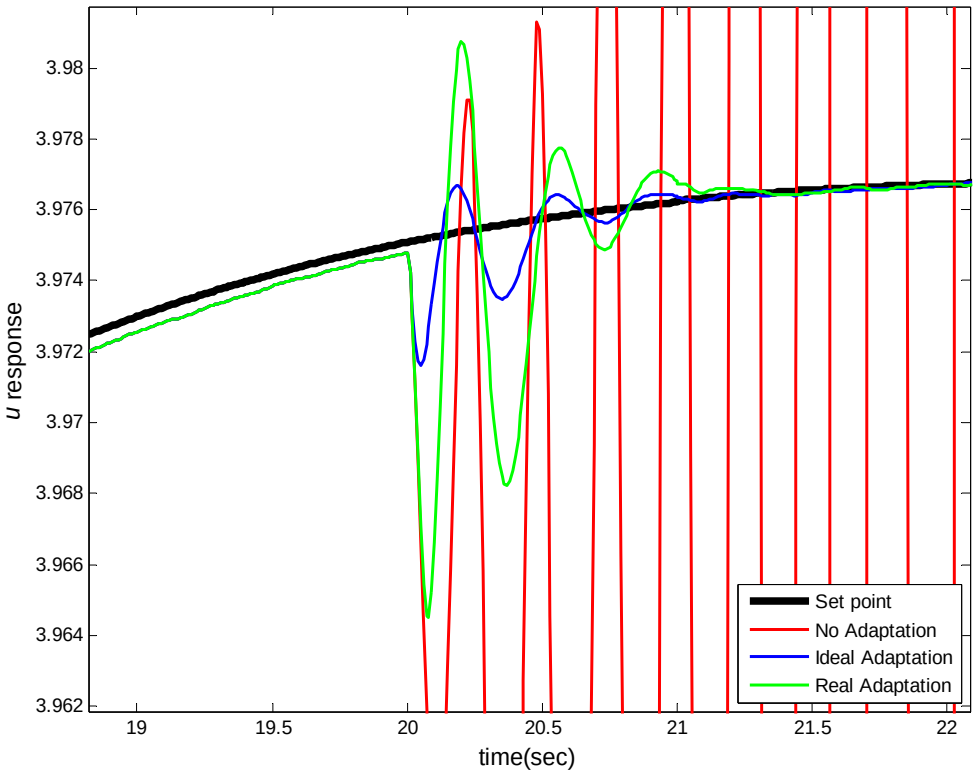


Figure 83: *u* Response of the Control System: Bell Mixer Fault

Figure 84 presents the control system response with respect to the *x* translational parameter.

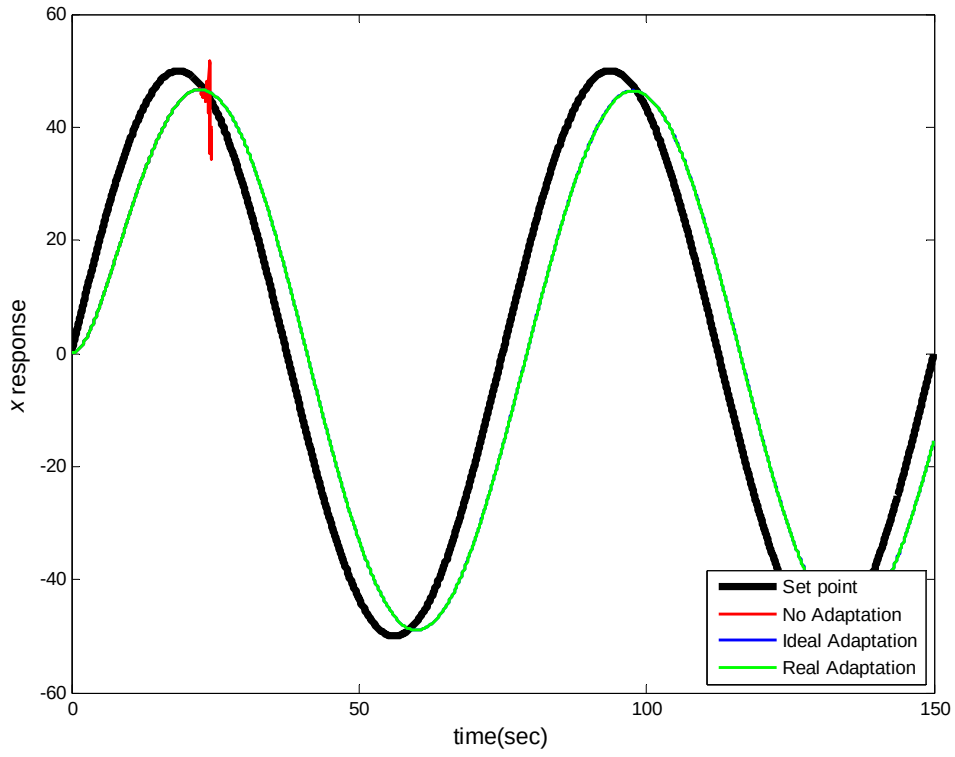


Figure 84: *x* Response of the Control System: Bell Mixer Fault

Figure 85 presents the control system response with respect to the *y* translational parameter.

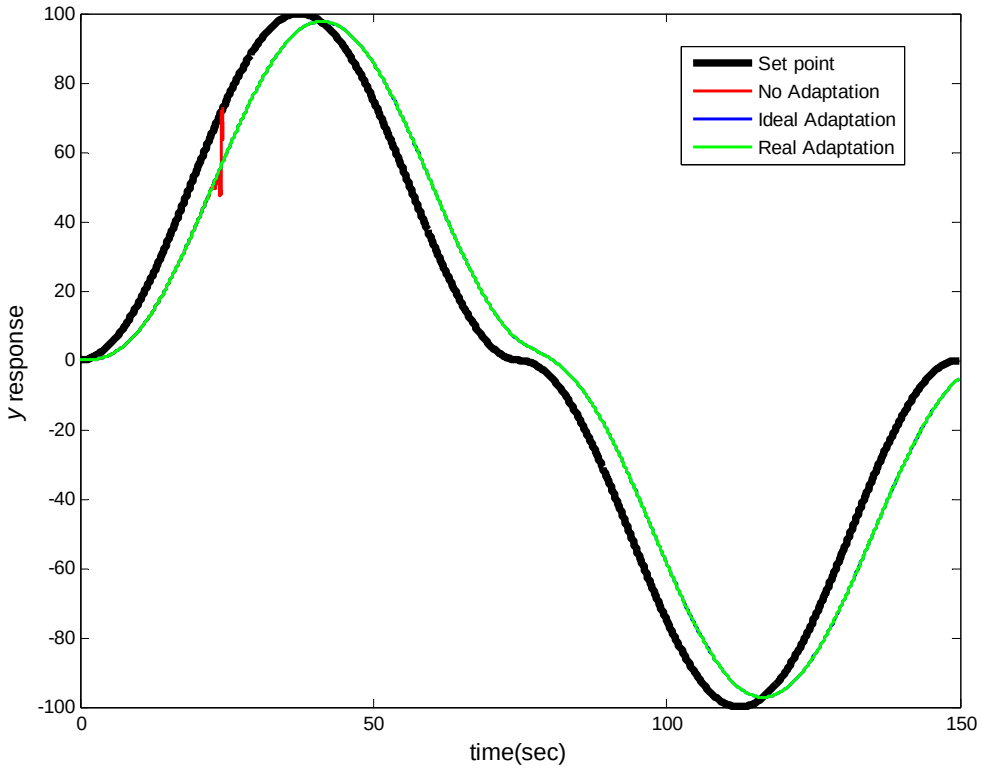


Figure 85: *y* Response of the Control System: Bell Mixer Fault

Figure 86 presents the control system response with respect to the *z* translational parameter.

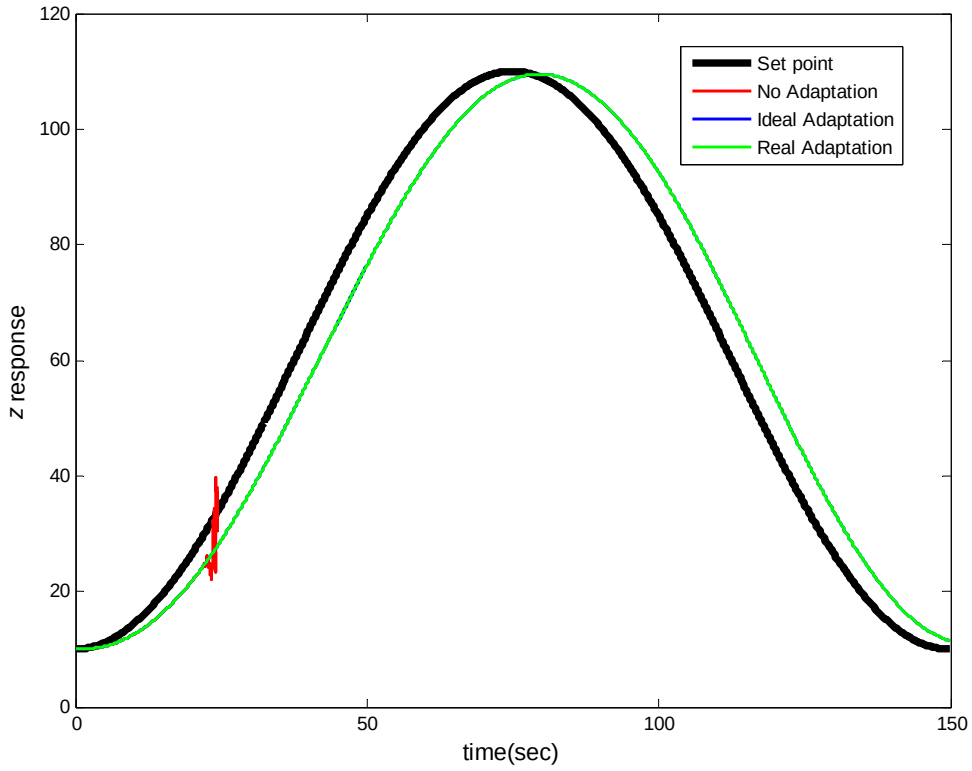


Figure 86: *z* Response of the Control System: Bell Mixer Fault

Figure 87 presents the control system response with respect to the yaw angle, (Ψ), parameter.

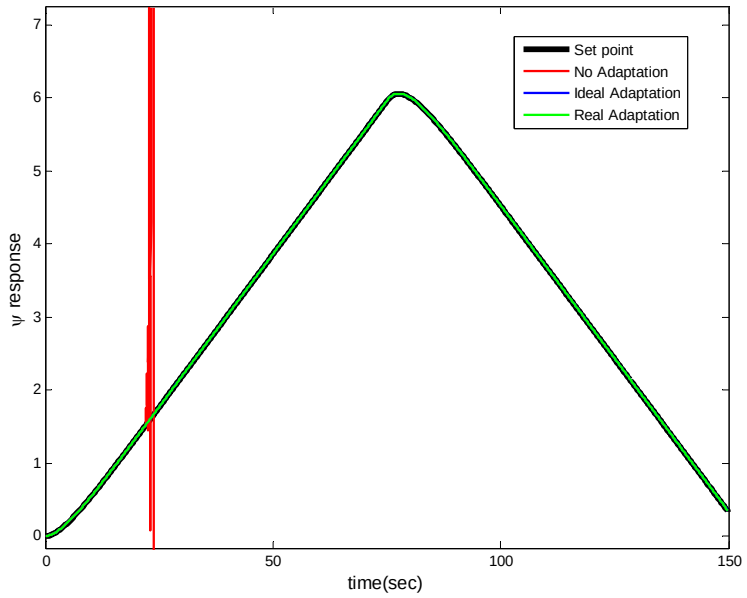


Figure 87: ΨResponse of the Control System: Bell Mixer Fault

6.4.4. Fault Case 4: Loss of Effectiveness

An actuator fault was simulated as a Loss of Effectiveness, (LOE). This fault was implemented as a factor multiplying the parameter Z_{col} and N_{col} in the **B** matrix of equation (2). In this fault case, two parameters were varied at the same time. The data demonstrate that the Kalman Filter accurately estimated both parameters. The LOE fault was applied at 20 sec. The maximum fault magnitude, which could be controlled, was a factor of 0.05. The outputs were disturbed with Gaussian noise. Figure 88 to Figure 91

present the responses of the estimated values and its covariance for the parameters Z_{col} and N_{col}. Figure 88 presents the response of the estimated value for the Z_{col} parameter.

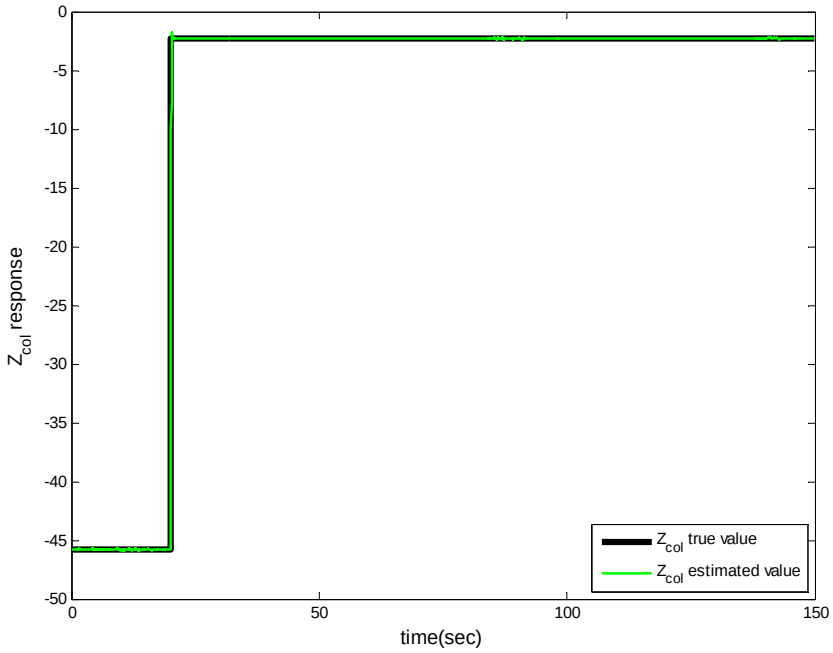


Figure 88: Response of the Estimated Z_{col} Parameter: LOE Fault

Figure 89 presents the covariance of the Z_{col} parameter.

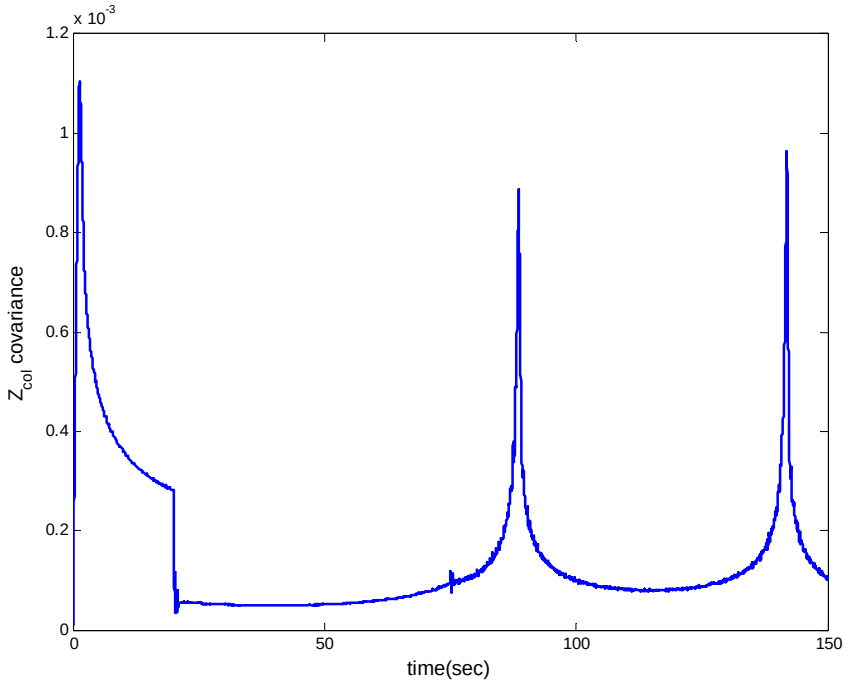


Figure 89: Z_{col} Covariance: LOE Fault

Figure 90 presents the response of the estimated value for the N_{col} parameter.

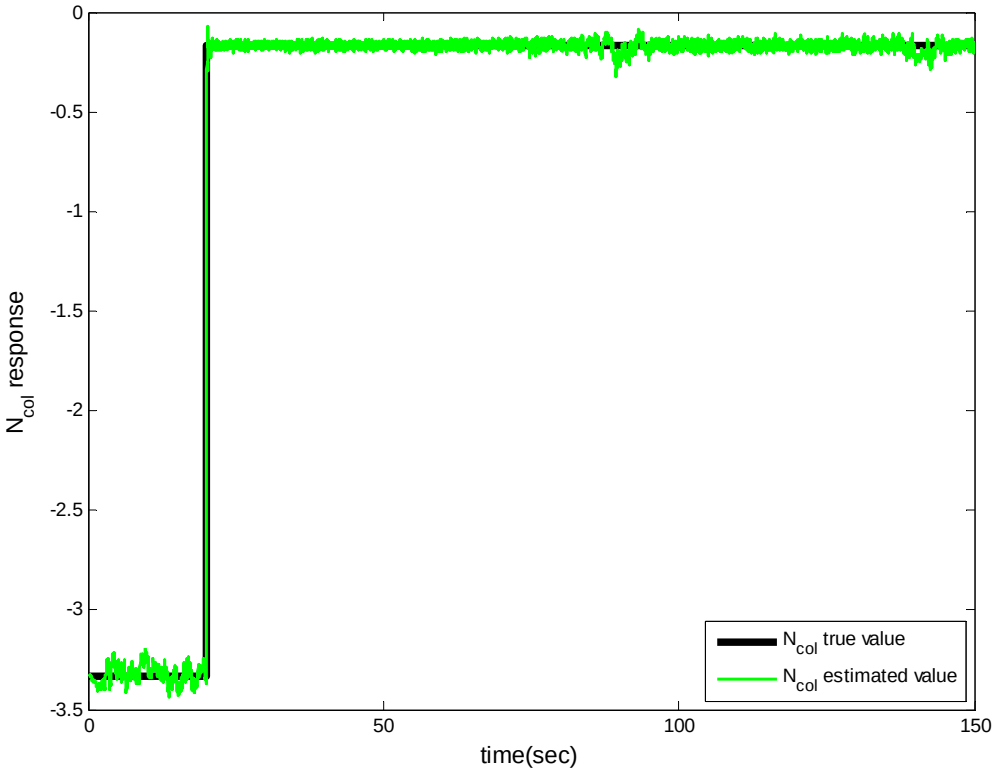


Figure 90: Response of the Estimated N_{col} Parameter: LOE Fault

Figure 91 presents the covariance of the N_{col} parameter.

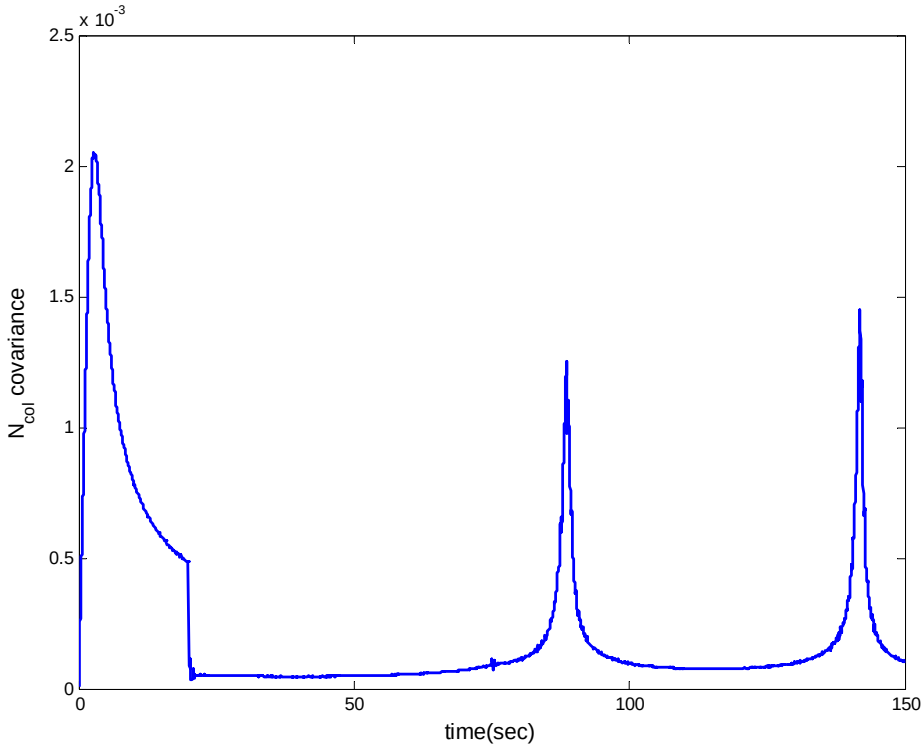


Figure 91: N_{col} Covariance: LOE Fault

Figure 92 to 94 present the responses of the body-frame velocities *v*, *w*, and the yaw rate, *r*. Figure 92 presents the response of the control system for the *v* translational velocity parameter when the fault was applied at 20 sec.

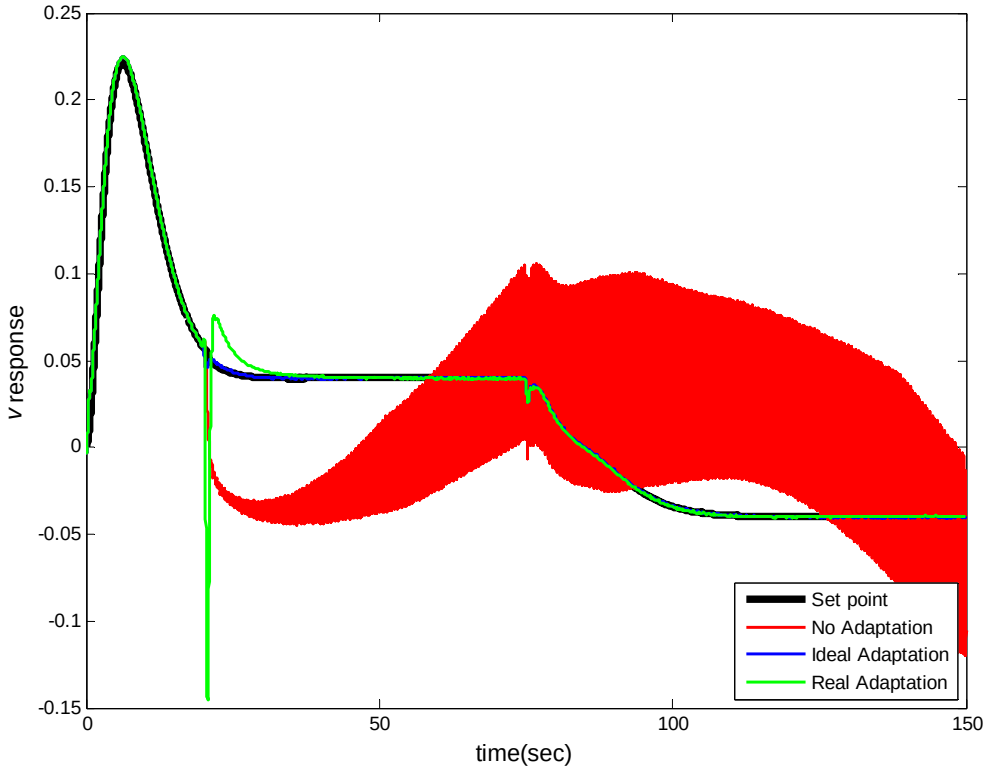


Figure 92: *v* Response of the Control System: LOE Fault

Figure 93 presents the response of the control system for the *w* translational velocity parameter when the fault was applied at 20 sec.

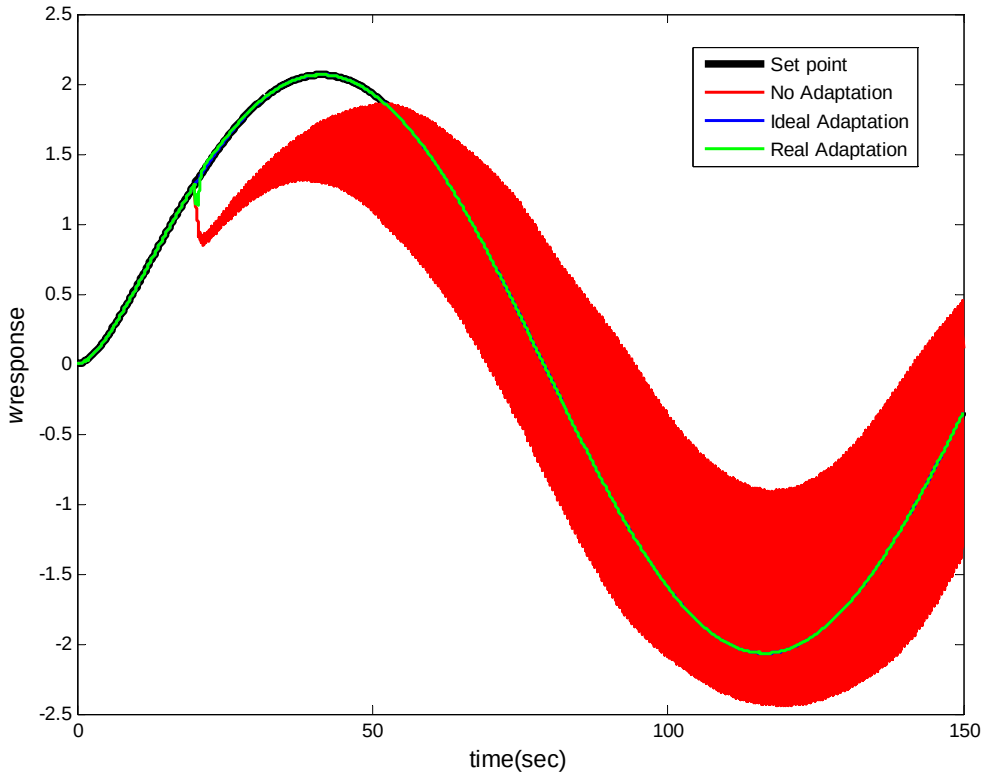


Figure 93: *w* Response of the Control System: LOE Fault

Figure 94 presents the responses of the body-frame yaw rate, *r*.

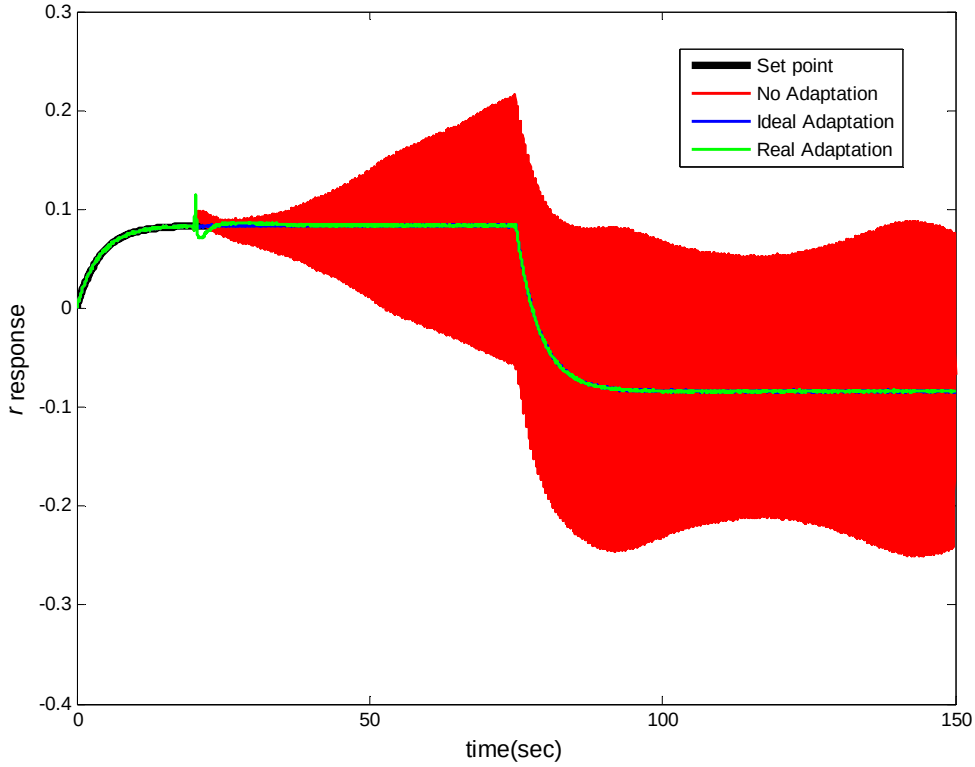


Figure 94: *r* Response of the Control System: LOE Fault

Figure 95 to 98 present the responses of x, y, z, and the yaw angle (Ψ).

Figure 95 presents the control system response with respect to the x translational parameter.

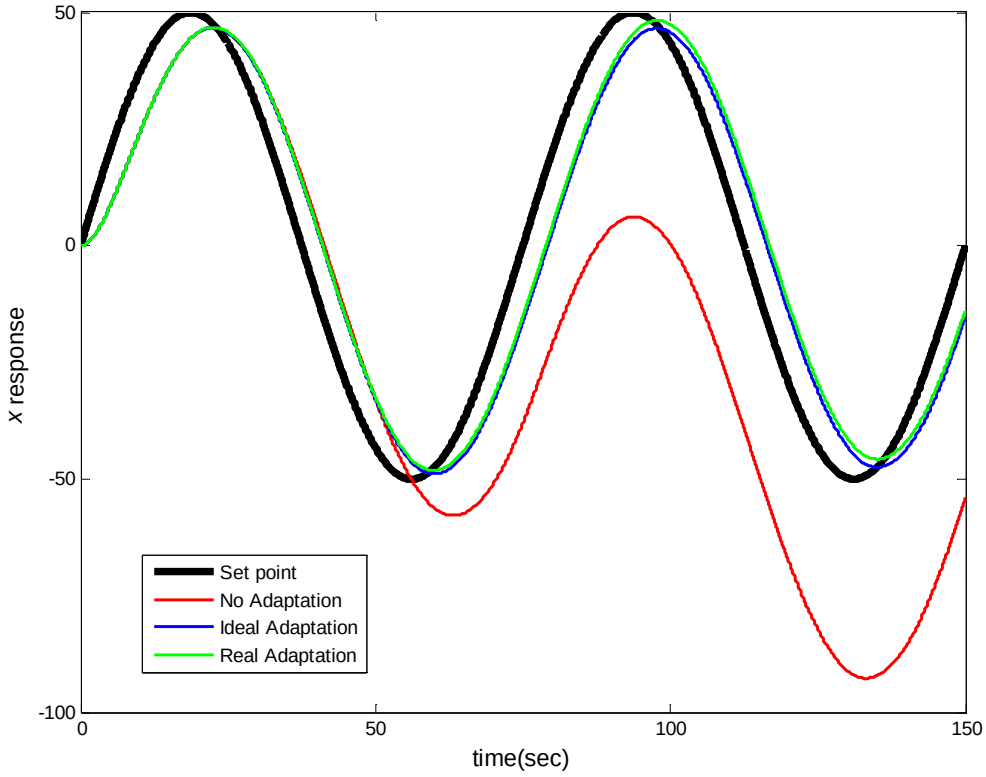


Figure 95: x Response of the Control System: LOE Fault

Figure 96 presents the control system response with respect to the *y* translational parameter.

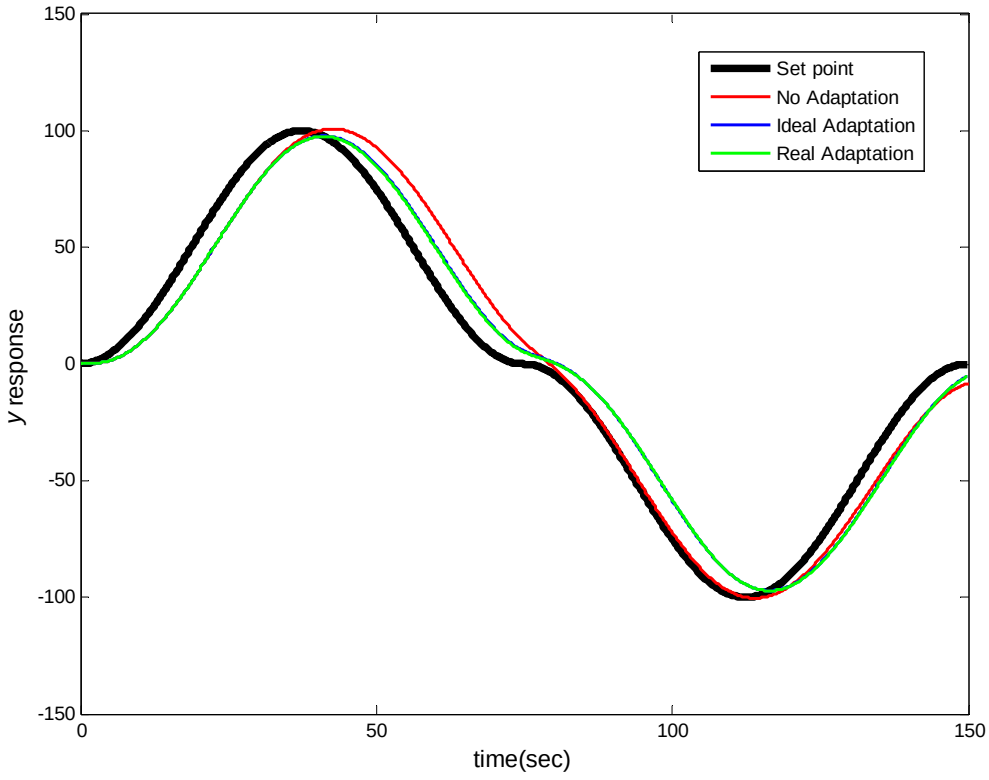


Figure 96: *y* Response of the Control System: LOE Fault

Figure 97 presents the control system response with respect to the *z* translational parameter.

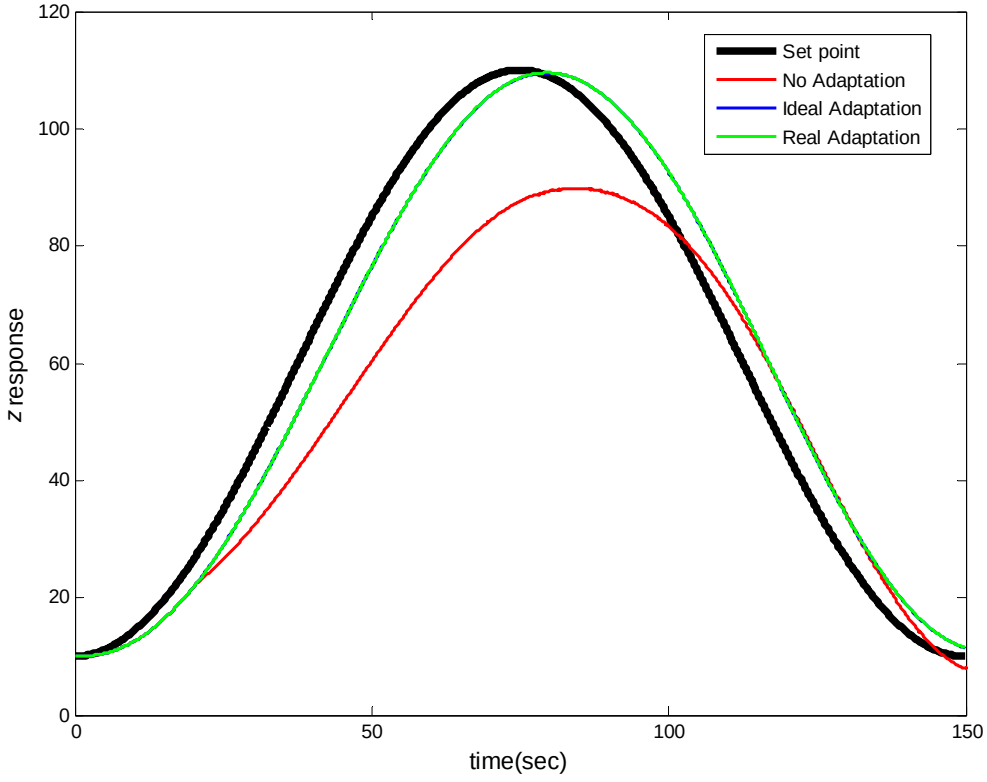


Figure 97: *z* Response of the Control System: LOE Fault

Figure 98 presents the control system response with respect to the yaw angle, (Ψ), parameter.

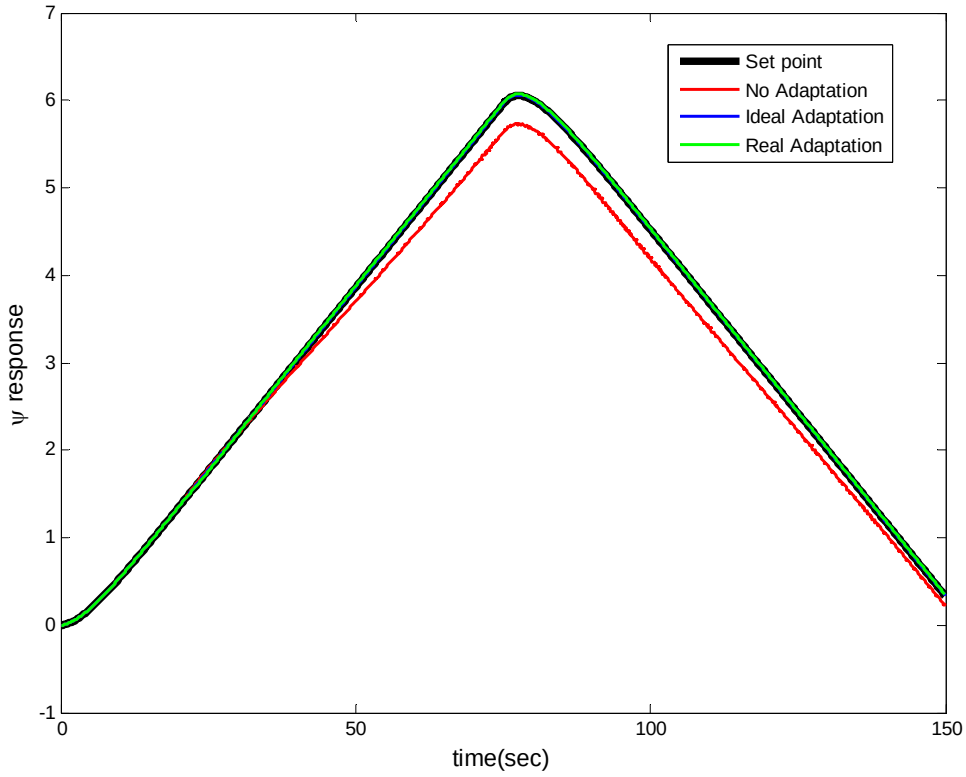


Figure 98: Ψ Response of the Control System: LOE Fault

Chapter 7

Conclusions and Future Work

7.1. Conclusions

The focus of this research was the improvement of the reliability of small-scale helicopters. The reliability was realized through the novel combination of a joint Extended Kalman filter and model predictive control techniques.

Before the development of the control techniques, a comprehensive comparison of the Extended Kalman Filter and the Unscented Kalman Filter was required to select the best implementation. The comparison of the Extended Kalman filter and the Unscented Kalman Filter demonstrated that the performance of the filters is dependent upon the approximation used for the nonlinear model of the system. The UKF presented a higher sensitivity to noise. For this reason, the EKF was selected as the method providing the most robust form of parameter estimation when utilized in conjunction with the MPC.

The estimation of the model's parameters and control design are the fundamental concepts involved in the implementation of adaptive control systems. These estimations are particularly relevant to the self-tuning regulator approach. Similarly, fault-tolerant control systems are based on the detection and identification of faults and the controller re-design concepts. This research took advantage of these similarities and proposed a

flight control system based on model predictive control, which possesses the advantages of both adaptive control and fault-tolerant control.

The developed framework highlighted some potential capabilities, not studied in this research, which were inherited from the self-tuning regulator controller scheme. Some of these capabilities most relevant to this research were;

- The minimization of the performance degradation produced by the normal wear of the helicopter's components,
- Changes of the dynamic characteristics of the system when the operating points change,
- Changes in the load carried for the helicopter,
- Change of the mass of the helicopter due to the consumption of the combustible.

A joint Extended Kalman filter simultaneously estimated the states and parameters of the system. Successful estimation of changes of parameters in the system and/or input matrices was performed. The behavior and magnitude of the covariance of the estimated parameters showed that the joint EKF possessed fast convergence and was able to estimate the parameters with low uncertainty.

The use of the joint Extended Kalman filter provided a straightforward approach to implement the function of fault detection and identification, (FDI). An additional module based on the calculation of the residual and the heuristic selection of thresholds normally provides this function. An additional advantage of the joint EKF was that it possesses the ability of detect slow time-varying changes of the parameters of the system.

The standard FID approach, which uses residual calculation and heuristic threshold selection, has difficulties detecting these types of changes or faults.

A Loss-of-Effectiveness of the collective actuator was represented as changes of the Z_{col} and the N_{col} parameters of the helicopter model. A fault of the mechanical mixer between the stabilizer bar and the main blade pitch control, the Bell mixer, was represented by changes of the A_c parameter. A generic system fault was represented as changes of the X_u parameter. The developed fault-tolerant adaptive approach was able to detect faults and handle them while maintaining excellent performance.

The developed control system was able to increase the reliability of small-scale helicopters through an effective handling of faults. The developed fault-tolerant controller combined the advantages of adaptive control techniques and fault-tolerant control techniques by the use of a Joint Kalman filter as parameter estimator.

7.2. Future Work

Some of the avenues for future research are:

- The use of a first-principles nonlinear model of the helicopter, which provides a better correlation between parameter changes and real physical changes. The model should provide a greater certainty of the fault tolerance capabilities of the flight control system.
- The use of a nonlinear MPC for tracking of the inertial position coordinates. This configuration will include the transformation from the body-frame to inertial-frame coordinates and vice versa. These transformations will introduce nonlinearities.

- Online retuning of the model predictive controller. In the research, retuning of the MPC was not necessary for the magnitude of the faults studied. However, it would be interesting to research the manner in which the online retuning of the MPC could increase its capability to handle more severe faults.
- Online determination of close-to-failure conditions. This investigation should determine the magnitude and performs localization for un-recoverable faults.

Some important aspects need further research for obtaining a clearer view of the advantages and disadvantages of the UKF with respect to the EKF. Some of these aspects are:

- Analyze the performance of the filter for other benchmark problems presented in the research literature.
- Analyze the effect of using parameter/states constraints in the performance of the UKF.
- Study the use of persistent excitation for improving the estimation accuracy

References

[1] G. F. Franklim, J. D. Powel and A. Emani-Naeini, "Feedback Control of Dynamic References Systems", Fifth ed., Pearson Prentice Hall, 2006

[2] R. M. Murray, K. J. Åström, S. P. Boyd, R. W. Brockett and G. Stein, "Future Directions in Control in an Information-Rich World", IEEE Control Systems Magazine, pp 20-33, 2003

[3] "Unmanned Aircraft Systems Roadmap", 2005-2030, http://www.acq.osd.mil/usd/-Roadmap%20Final2.pdf

[4] "UAVs Coming Soon to a Sky Near You", http://www.jpdo.gov/newsArticle.-asp?ID=14

[5] D.-W.Gu, P.Hr. Petkov and M.M. Konstantinov, "Robust Control Design with MATLAB", London, Springer, 2005

[6] K. Zhou and J.C. Doyle, "Robust and Optimal Control", Prentice Hall, 1996

[7] C. W. McFall, "Integrated Fault Detection and Isolation and Fault-Tolerant Control of Nonlinear Process Systems", Ph.D thesis, Chemical Engineering Department, University of California Los Angeles, 2008

[8] A. Gani, "Fault-Tolerant Process Control: Handling Actuator and Sensor Malfunctions", Ph.D thesis, Chemical Engineering Department, University of California Los Angeles, 2007

[9] MIT Aerial Robotics Club, "The MIT Entry into the 1998 AUVS International Aerial Robotics Competition", Technical report, June 1, 1998

[10] E. N. Johnson, P. A. Debitetto, C. A. Trott and M. C. Bosse, "The 1996 MIT/Boston University /Draper Laboratory Autonomous Helicopter System", Proceedings 15th Digital Avionincs Systems Conference, pp 381–386, Atlanta, Georgia, October 27-31, 1996

[11] J. F. Montgomery, A. H. Fagg, and G. A. Bekey, "The USC AFV-I: A Behavior-Based Entry in the 1994 International Aerial Robotics Competition", IEEE Expert, (see also IEEE Intelligent Systems and Their Applications), volume 10, pp 16–22, 1995

[12] "Southwest Research Institute", http://www.swri.org/4org/d09/avionics/avmodisp/home.htm

[13] "Insitu, Inc.", http://www.insitu.com/

[14] "Boeing/Insitu ScanEagle", http://www.insitu.com/scaneagle

[15] "Aerovironment, Inc.", http://www.avinc.com/

[16] "Lew Aerospace Inc.", http http://www.lewaerospace.com/products.htm

[17] O. Amidi, T. Kanade, and J. R. Miller, "Vision-Based Autonomous Helicopter Research at Carnegie Mellon Robotics Institute 1991-1997," American Helicopter Society International Conference, number T7-3, Heli, Japan, April 1998

[18] http://web.mit.edu/newsoffice/1997/robocopter-0910.html

[19] J. Charles, "CMU's Autonomous Helicopter Explores New Territory", IEEE Intelligent Systems, pp 85–87, September-October 1998

[20] B. Mettler, M. B. Tischler, and T. Kanade, "System Identification of Small-Scale Unmanned Helicopter Dynamics", American Helicopter Society 55th Forum, May 25-27, 1999

[21] B. Mettler, T. Kanade, M. B. Tischler, and W. Messner, "Attitude Control Optimization for a Small-Scale Unmanned Helicopter", AIAA Guidance, Navigation and Control Conference, 2000

[22] J. A. Agnell and J. G. Scheider, "Autonomous Helicopter Control using Reinforcement Learning Policy Search Methods", 2001 IEEE International Conference on Robotics & Automation, Seoul, Korea, May 21- 26, 2001

[23] M. La Civita, W. C. Messner, and T. Kanade, "Modeling of Small-Scale Helicopter with Integrated First-Principles and System Identification Techniques", American Helicopter Society 58th Annual Forum, Montreal, Canada, June 11-13, 2002

[24] M. La Civita, G. Papageorgiou, W. C. Messner, and T. Kanade, " Design and Flight Testing of a Gain-Scheduled H_∞ Loop Shaping Controller for Wide-Envelope Flight of a Robotic Helicopter", Proceedings of the American Control Conference, Denver, Colorado, June 4-6 2003

[25] M. La Civita, G. Papageorgiou, W. C. Messner, and T. Kanade, "Design and Flight Testing of an H_∞ Controller for a Robotic Helicopter", Journal of Guidance, Control, and Dynamics, 29(2), March-April 2006

[26] V. Gavrilets, B. Mettler, and E. Feron, "Nonlinear Model for a Small-Size Acrobatic Helicopter", AIAA Guidance Navigation and Control Conference, volume AIAA 2001-4333, Montreal, QC, 2001

[27] T. Schouwenaars, B. Mettler, E. Feron, and J. How, "Hybrid Architecture for Full-Envelope Autonomous Rotorcraft Guidance", American Helicopter Society 59th Annual Forum, Phoenix, Arizona, May 6-8, 2003

[28] T. Schouwenaars, B. Mettler, E. Feron, and J. How, "Hybrid Model for Receding Horizon Guidance Of Agile Autonomous Rotorcraft", 16th IFAC Symposium on Automatic Control in Aerospace, St. Petersburg, Russia, June 2004

[29] T. Schouwenaars, E. Feron, and J. How, "Multi-Vehicle Path Planning for Non-Line of Sight Communication", 2006 American Control Conference, Minneapolis, Minnesota, USA, June 14-16, 2006

[30] M. Valenti, B. Bethke, D. Dale, A. Frank, J. McGrew, S. Ahrens, J. How, and J. Vian, "The MIT Indoor Multi-Vehicle Flight Testbed", 2007 IEEE International Conference on Robotics and Automation, Roma, Italy, April 10 -14, 2007

[31] M. Piedmonte and E. Feron, "Aggressive Maneuvering of Autonomous Aerial Vehicles: A Human-Centered Approach", International Symposium on Robotics Research 1999, Snowbird, UT, October 1999

[32] E. Frazzoli, M. A. Dahleh, and E. Feron, "A Hybrid Control Architecture for Aggressive Maneuvering of Autonomous Helicopters", Proceeding of the 38th Conference on Decision, Phoenix, Arizona USA, December 1999

[33] T. Samad and G. Balas, "Software-Enabled Control: Information Technology for Dynamical Systems", Wiley Interscience - IEEE Press, Piscataway, NJ, 2003

[34] E. Frazzoli, M. A. Dahleh, and E. Feron, "Trajectory Tracking Control Design for Autonomous Helicopters using Backstepping Algorithm", American Control Conference, pp 4102–4107, June 2000

[35] B. Mettler, V. Gavrilets, E. Feron, and T. Kanade, “Dynamic Compensation for High-Bandwidth Control of Small-Scale Helicopter”, American Helicopter Society Test and Evaluation Technical Specialist Meeting, January 2002

[36] L. Wills, S. Sander, S. Kannan, A. Kahn, J. V. R. Prasad, and D. Schrage, “An Open Control Platform for Reconfigurable, Distributed, Hierarchical Control Systems”, Proceedings of the Digital Avionics Systems Conference, Philadelphia, PA, October 2000

[37] F. Rufus, B. Heck, and G. Vachtsevanos, “Software-Enabled Adaptive Mode Transition Control for Autonomous Unmanned Vehicles”, Proceedings of the 19th Digital Avionics Systems Conferences, Philadelphia, PA, USA, October 7-13, 2000

[38] G. Vachtsevanos, “Neuro-Fuzzy Approach to Mode Transitioning in Aerospace Applications”, Joint 9th IFSA World Congress and 20th NAFIPS International Conference, volume 2, pp 1080–1085, Vancouver, BC, Canada, July 25-28, 2001

[39] D. P. Schrage, Y. K. Yillikci, S. Liu, J. V. R. Prasad, and S. V. Hanagud, “Instrumentation of the Yamaha R-50/rmax Helicopter Testbeds for Airloads Identification and Follow-On Research”, 25th European Rotorcraft Forum, 1999

[40] E. N. Johnson and D. P. Schrage, “The Georgia Tech Unmanned Aerial Research Vehicle: GTMAX”, AIAA Guidance, Navigation and Control Conference, number AIAA-2003-5741, Austin, TX, August 2003

[41] E. N. Johnson and D. P. Schrage, “System Integration and Operation of a Research Unmanned Aerial Vehicle”, Journal of Aerospace Computing, Information, And Communication, 1(1):5–18, January 2004

[42] A. J. Calise, B. S. Kim, J. Leitner, and J. V. R. Prasad, “Helicopter Adaptive Flight Control using Neural Networks”, 33rd Conference on Decision and Control, Lake Buena Vista, FL, December 2004

[43] J.V.R. Prasad, A.J. Calise, Y. Pei, and J. E. Corban, “Adaptive Nonlinear Controller Synthesis and Flight Test Evaluation on an Unmanned Helicopter”, 1999 IEEE International Conference on Control Applications, Kohala Coast, Hawaii, USA, August 22-27, 1999

[44] E. N. Johnson, A. J. Calise, and J. E. Corban, “Adaptive Guidance and Control for Autonomous Launch Vehicles”, IEEE Proceeding of the Aerospace Conference, volume 6, pp 2669–2682, Big Sky, MT, USA, March 10-17, 2001

[45] E. N. Johnson and A. J. Calise, "Neural Network Adaptive Control of Systems with Input Saturation", Proceeding of the American Control Conference, volume 6, pp 2669–2682, Arlington, VA, June 25-27, 2001

[46] S. K. Kannan and E. N. Johnson, "Adaptive Trajectory Based Control for Autonomous Helicopters", 21st Digital Avionics Systems Conference, volume 2, pp 8D.1–1–8D.1–12, Irvine, CA, October 27-31, 2002

[47] R. Rysdyk and A. J. Calise, "Robust Nonlinear Adaptive Flight Control for Consistent Handling Qualities", IEEE Transactions on Control Systems Technology, 13(6):896–910, 2005

[48] M. Idan, M. Johnson, and A. J. Calise, "Intelligent Aerodynamic/Propulsion Flight Control for Flight Safety: A Nonlinear Adaptive Approach", Proceedings of the 2001 American Control Conference, volume 4, pp 2918–2923, Arlington, VA, USA, June 25-27, 2001

[49] G. R. Drozeski, B. Saha, and G. J. Vachtsevanos, "A Fault Detection and Reconfigurable Control Architecture for Unmanned Aerial Vehicles," 2005 IEEE Aerospace Conference, Big Sky, MT, March 5-12, 2005

[50] G. R. Drozeski, "A Fault-Tolerant Control Architecture for Unmanned Aerial Vehicles", Ph.D thesis, Georgia Institute of Technology, December 2005

[51] G. Vachtsevanos, W. Kim, S. Al-Hasan, F. Rufus, M. Simon, D. Shrage, and J. V. R. Prasad, "Autonomous Vehicles: From Flight Control to Mission Planning using Fuzzy Logic Techniques", 13th International Conference on Digital Signal Processing, volume 2, pp 977–981, Santorini, Greece, July 2-4, 1997

[52] H. Shim, T. J. Koo, F. Hoffmann, and S. Sastry, "A Comprehensive Study of Control Design for an Autonomous Helicopter", Proceedings of the 37th IEEE Conference on Decision and Control, vol. 4, pp 3653–3658, Tampa, FL, 1998

[53] David. H. Shim, H. Jin Kim, and S. Sastry, "Decentralized Nonlinear Model Predictive Control of Multiple Flying Robots", Proceedings of the 42nd IEEE Conference on Decision and Control, pp 3621–3626, Maui, Hawaii, USA, December 9-12, 2003

[54] D. H. Shim and S. Sastry, "A Situation-Aware Flight Control System Design using Real-Time Model Predictive Control for Unmanned Autonomous Helicopters", AIAA Guidance, Navigation and Control Conference and Exhibit, number AIAA 2006-6101, Keystone, Colorado, August 21-24 2006

[55] H. Jin Kim, D. H. Shim, and S. Sastry, "Nonlinear Model Predictive Tracking Control for Rotorcraft-Based Unmanned Aerial Vehicles", Proceedings of the American Control Conference, vol. 5, pp 3576–3581, Anchorage, AK, May 8-10, 2002

[56] C. J. Tomlin and S. Sastry, "Bounded Tracking for Nonminimum Phase Nonlinear Systems with Fast Zero Dynamics", Proceedings of the 35th Conference on Decision and Control, vol. 2, pp 2058–2063, Kobe, Japan, December 11-13, 1996

[57] T. J. Koo and S. Sastry, "Output Tracking Control Design of a Helicopter Model Based on Approximate Linearization", Proceedings of the 37th IEEE Conference on Decision and Control, vol. 4, pp 3635–3640, Tampa, FL, Dec 16-18, 1998

[58] H. J. Kim, D. H. Shim, and S. Sastry, "Flying Robots: Modeling, Control and Decision Making", Proceeding of the 2002 IEEE International Conference on Robotics and Automation, vol. 1, pp 66–71, Washington, DC, May 11-15, 2002

[59] D. H. Shim, H. Chung, and S. Sastry, "Conflict-Free Navigation in Unknown Urban Environments", IEEE Robotics & Automation Magazine, vol. 13, pp 27–33, September 2006

[60] J. Sprinkle, J. M. Eklund, H. J. Kim, and S. Sastry, "Encoding Aerial Pursuit/Evasion Games with Fixed Wing Aircraft into a Nonlinear Model Predictive Tracking Controller", 43rd IEEE Conference on Decision and Control, vol. 3, pp 2609–2614, Atlantis, Paradise Island, Bahamas, December 14-17, 2004

[61] J. M. Eklund, J. Sprinkle, and S. Sastry, "Implementing and Testing a Nonlinear Model Predictive Tracking Controller for Aerial Pursuit Evasion Games on a Fixed Wing Aircraft", Proceedings of American Control Conference (ACC) 2005, pp 1509–1514, June 2005

[62] S. Saripalli, D. J. Naffin, and G. S. Sukhatme, "Autonomous Flying Vehicle Research at the University Of Southern California", A. Schultz and L. E. Parker, editors, First International Workshop on Multi-Robot Systems, Multi-Robot Systems: From Swarms to Intelligent Automata, pp 73–82, Kluwer Academic Publishers, 2002

[63] L. O. Mejias, S. Saripalli, P. Cervera, and G. S. Sukhatme, "Visual Servoing of an Autonomous Helicopter in Urban Areas using Feature Tracking", Journal of Field Robotics, 23(3):185–199, 2006

[64] S. Saripalli and G. S. Sukhatme, "Landing a Helicopter on a Moving Target", International Conference on Robotics and Automation, pp 2030-2035, April 2007

[65] A. H. Fagg, M. A. Lewis, J. F. Montgomery, and G. A. Bekey, “The USC Autonomous Flying Vehicle: An Experiment in Real-Time Behavior-Based Control”, IEEE/RSJ International Conference on Intelligent Robots and Systems, pp 1173–1180, Yokohama, Japan, July 1993

[66] C. L. Castillo, W. Alvis, M. Castillo-Effen, W. Moreno, and K. Valavanis, “Small-Scale Helicopter Analysis and Design for Non-Aggressive Flights”, 2005 IEEE International Conference on Systems, Man, and Cybernetics, Hawaii, 2005

[67] C. Castillo, W. Moreno and K. Valavanis, "Unmanned Helicopter Waypoint Trajectory Tracking Using Model Predictive Control", 15th Mediterranean Conference on Control and Automation (MED'07), Athens, Greece, June 27-29, 2007

[68] M. Castillo-Effen, C. Castillo, W. Moreno, and K. Valavanis, “Robustification of Decentralized PID Control for Small Unmanned Rotorcraft,” 2007 IEEE/RSJ International Conference on Intelligent Robots and Systems, San Diego, Oct. 29 - Nov. 2, 2007

[69] C. N. Jones, “Reconfigurable Flight Control: First Year Report”, Department of Engineering, University of Cambridge, March 4, 2005

[70] K. J. Åström, and B. Wittenmark, “Adaptive Control”, 2nd ed., Adisson-Wesley, 1995

[71] P.A. Ioannou and J. Sun, “Robust Adaptive Control”, Prentice Hall, 1996

[72] W. J. Rugh and J. S. Shamma, “Research on Gain Scheduling”, Automatica, vol. 36, pp 1401-1425, 2000

[73] N. M. Filatov and H. Unbehauen, “Adaptive Dual Control”, Springer-Verlag, 2004

[74] J. M. Maciejowski, “Predictive Control with Constraints”, Prentice Hall, May 2002

[75] J. Richalet, A. Rault, J. L. Testud, and J. Papon, “Model Predictive Heuristic Control: Applications to Industrial Processes”, Automatica, 14:413–428, 1978

[76] C.R. Cutler and B.L. Ramaker, “Dynamic Matrix Control - A Computer Control Algorithm”, Proceedings, Joint American Control Conference, San Francisco, 1980

[77] J. M. Sanchez, “Adaptive-Predictive Control System”, US patent 4197576, August 1976

[78] E. F. Camacho and C. Bordon, “Model Predictive Control”, Springer-Verlag, 2004

[79] A. Bemporad, and M. Morari, "Robust Model Predictive Control: A Survey," in Robustness in Identification and Control. vol. 245, pp 207-226A., Garulli, A. Tesi, and A. Vicino, Eds., Springer-Verlag, 1999

[80] S. Simani, C. Fantuzzi, and R. J. Patton, “Model-Based Fault Diagnosis in Dynamic Systems Using Identification Techniques”, Springer-Verlag, 2002

[81] M. Blanke, M. Kinnaert, J. Lunze, and M. Staroswiecki, “Diagnosis and Fault-Tolerant Control”, Springer 2003

[82] R. J. Patton, “Fault-Tolerant Control Systems: The 1997 Situation”, IFAC Symposium on Fault Detection, Supervision and Safety for Technical Processes, vol. 3, pp 1033-1054, Kingston Upon Hull, UK, August 26-28, 1997

[83] Y. Zhang and J. Jiang, “Bibliographical Review on Reconfigurable Fault-Tolerant Control Systems”, 5th IFAC on Fault Detection, Supervision and Safety for Technical Processes, pp 265-276, Washington, D. C., USA, June 9-11, 2003

[84] J. Lunze and J. H. Richter, “COSY Three-Tank Benchmark Problem for Control Reconfiguration”, http://www.ruhr-uni-bochum.de/atp, October 29, 2004

[85] H. Alwi, “Fault-Tolerant Sliding Model Control Schemes with Aerospace Applications”, Ph.D thesis, Department of Engineering, University of Leicester, February 2008

[86] J. D. Boskovic and R. K. Mehra, “Failure Detection, Identification and Reconfiguration in Flight Control”, Fault Diagnosis and Fault Tolerance for Mechatronic Systems: Recent Advances, F. Caccavale and L. Villani, Eds.: Springer-Verlag, 2003

[87] R. E. Kalman, “A New Approach to Linear Filtering and Prediction Problems”, Transactions of the ASME: Journal of Basic Engineering, 82:35–45, March 1960

[88] G. Welch and G. Bishop, “An Introduction to the Kalman Filter”, Technical report, Department of Computer Science, University of North Carolina at Chapel Hill, Chapel Hill, NC 27599-3175, April 5, 2004

[89] S. J. Julier and J. K. Uhlmann, “A New Extension of the Kalman Filter to Nonlinear Systems”, AeroSense: 11th Int. Symposium Aerospace/Defense Sensing, Simulation and Controls, pp 182–193, Orlando, FL, 1997

[90] E. A. Wan, R. van der Merwe, and A. T. Nelson, "Dual Estimation and the Unscented Transformation", S.A. Solla, T. K. Leen, and K.-R. Muller, editors, Advances in Neural Information Processing Systems 12, pp. 666–672, MIT Press, November 2000

[91] R. van der Merwe and E. A. Wan, "Sigma-Point Kalman Filter for Nonlinear Estimation and Sensor Fusion - Applications to Integrated Navigation", AIAA Guidance Navigation and Control Conference and Exhibit, number AIAA-2004-5120, Providence, Rhode Island, August 16-19, 2004

[92] S. J. Julier and J. K. Uhlmann, "A General Method for Approximating Nonlinear Transformations of Probability Distributions", Technical report, The Robotics Research Group, Department of Engineering Science, The University of Oxford, November 1996

[93] R. Kandepu, B. Foss, and L. Imsland, "Applying the Unscented Kalman Filter for Nonlinear State Estimation", Journal of Process Control, vol. 18, pp 753-768, 2008

[94] D. Simon, "Optimal State Estimation: Kalman, H_∞, and Nonlinear Approach", Wiley-Interscience, 2006

[95] S. J. Julier, J. K. Uhlmann, and H. F. Durrant-Whyte, "A New Approach for Filtering Nonlinear Systems", American Control Conference, pp 1628-1632 Seattle, Washington, 1995

[96] S. J. Julier, "The Scaled Unscented Transformation", American Control Conference, vol. 6, pp 4555-4559 Anchorage, AK, 2002

[97] R. v. d. Merwe, "Sigma-Point Kalman Filters for Probabilistic Inference in Dynamic State-Space Models", Ph.D thesis, OGI School of Science & Engineering, Oregon Health & Science University, April 2004

[98] G. Chowdhary and R. Jategaonkar, "Aerodynamic Parameter Estimation from Flight Data Applying Extended and Unscented Kalman Filter", AIAA Atmospheric Flight Mechanics Conference and Exhibit, Keystone, Colorado, 2006

[99] D. Simon, "A Comparison of Filtering Approaches for Aircraft Engine Health Estimation", Aerospace Science and Technology, vol. 12, pp 276-284, 2008

[100] M. Athans, R. P. Wishner, and A. Bertolini, "Suboptimal State Estimation for Continuous-Time Nonlinear Systems from Discrete Noisy Measurements", IEEE Transactions on Automatic Control, vol. AC-13, No. 5, October 1968

[101] S. Julier, J. Uhlmann, and H. F. Durrant-Whyte, "A New Method for the Nonlinear Transformation of Means and Covariances in Filters and Estimators" IEEE Transactions on Automatic Control, vol. 45, no. 3, March 2000

[102] B. Mettler, "Identification, Modeling and Characteristic of Miniature Rotorcraft", Kluwer Academic Publishers, 2003

[103] C. L. Castillo, et al, "Small Unmanned Helicopter Simplified and Decentralized Optimization -Based Controller Design for Non-Aggressive Flights", International Transactions on Systems Science and Applications, vol. 1, no. 3, pp 303-315, 2006

[104] E. Aggelogiannaki and H. Sarimveis, "A Simulated Annealing Algorithm for Prioritized Multiobjective Optimization - Implementation in an Adaptive Model Predictive Control Configuration", IEEE Transactions on Systems, Man, and Cybernetics - Part B: Cybernetics, vol. 37, no. 4, pp 902-915, August 2007

[105] T. -H. Kim and T. Sugie, "Adaptive Receding Horizon Predictive Control for Constrained Discrete-Time Linear Systems with Parameter Uncertainties", International Journal of Control, vol. 81, no. 1, pp 62-73, January 2008

[106] D. Corona and B. De Schutter, "Adaptive Cruise Control for a SMART Car: A Comparison Benchmark for MPC-PWA Control Methods", IEEE Transactions on Control Systems Technology, vol. 16, no. 2, pp 365-372, March 2008

[107] C.-H. Lu and C.-C. Tsai, "Adaptive Predictive Control with Recurrent Neural Network for Industrial Processes: An Application to Temperature Control of a Variable-Frequency Oil-Cooling Machine", IEEE Transactions on Industrial Electronics, vol. 55, no. 3, pp 1366-1375, March 2008

[108] K. R. Muske, J. C. P. Jones, and E.M. Franceschi, "Adaptive Analytical Model-Based Control for SI Engine Air-Fuel Ratio", IEEE Transactions on Control Systems Technology, vol. 16, no. 4, pp 763-768, July 2008

[109] J. M. Maciejowski and C.N. Jones, "MPC Fault-Tolerant Flight Control Case Study: Flight 1862", IFAC SAFEPROCESS Conference, Washington, DC, 9-11 June 2003

[110] J. Qi, Z. Jiang, X. Zhao, and J. Han, "UKF-Based Rotorcraft UAV Fault Adaptive Control for Actuator Failure", IEEE International Conference on Robotics and Biomimetics, pp 1545-1550Sanya, China, December 15-18, 2007

[111] J. Qi and J. Han, “Fault Adaptive Control for RUAV Actuator Failure with Unscented Kalman Filter”, Third International Conference on Innovative Computing, Information and Control (ICICIC2008), Dalian, China, June 18-20, 2008

[112] T. Miksch, A. Gambier, and E. Badreddin, “Real-time Performance Comparison of Fault-Tolerant Controllers”, 17th IEEE International Conference on Control Applications Part of 2008 Multi-conference on Systems and Control, San Antonio, Texas,USA, September 3-5, 2008

[113] H. J. Kim and D. H. Shim, “A Flight Control System for Aerial Robots: Algorithms And Experiments”, Control Engineering Practice, vol. 11, pp 1389-1400, 2003

About the Author

Carlos Leobardo Castillo received his Bachelors of Science from the University of Los Andes, Merida, Venezuela. Carlos worked as an Electrical Engineer in the MARAVEN oil company. In 1992, Carlos joined the University of Los Andes as an Instructor and was promoted to Assistant Professor in 2000. Carlos joined the graduate program at the University of South Florida in August 2001. In December 2003, Carlos received the Master of Science degree in Electrical Engineering. His master's thesis focused on face recognition using neural networks. During his graduate studies Carlos was a Teacher Assistant in the Linear Control Lab in the Department of Electrical Engineering, as a Research Assistant for the Center of Robot Assisted Search and Rescue and for the Unmanned Systems Laboratory. Carlos Castillo has published and presented his work at conferences, in journals and in book chapter contributions to the field of Low-level Control of Unmanned Aerial Vehicles. The research interests of Carlos include Model Predictive Control, Adaptive Control, Fault-Tolerant Control, Estimation of Parameter/States and System Identification. Carlos is a member of the IEEE.