A MULTI-INPUT MULTI-OUTPUT SELF TUNING REGULATOR FOR NONLINEAR HIGH PERFORMANCE AIRCRAFT CONTROL

A Thesis
Presented
to the Faculty of
California State University, Chico

In Partial Fulfillment
of the Requirement for the Degree
Master of Science
in
Electrical and Computer Engineering
Electronic Engineering Option

by
Wilfred L. Ghonsalves
Summer 2010
A MULTI-INPUT MULTI-OUTPUT SELF TUNING REGULATOR FOR NONLINEAR HIGH PERFORMANCE AIRCRAFT CONTROL

A Thesis

by

Wilfred L. Ghonsalves

Summer 2010

APPROVED BY THE DEAN OF GRADUATE STUDIES AND VICE PROVOST FOR RESEARCH:

---------------------------------------------
Katie Milo, Ed.D.

APPROVED BY THE GRADUATE ADVISORY COMMITTEE:

---------------------------------------------
Adel A. Ghandakly, Ph.D.
Graduate Coordinator

---------------------------------------------
Adel A. Ghandakly, Ph.D., Chair

---------------------------------------------
Dale Word, M.S.
DEDICATION

I would like to dedicate this thesis to my parents, Lawrence and Hilda Ghonsalves, whose unconditional love and sacrifice has instilled in me the drive and passion to follow my dreams and pursue my goals.

I would also like to acknowledge my brother, James Ghonsalves and my sisters, Ruby Pereira and Leena Mendes, for their constant support and encouragement, without which I would not have the strength to overcome life’s challenges.
ACKNOWLEDGMENTS

First and foremost, I would like to thank God for the manifold blessings that He has showered in my life. I truly believe that everything I have is only because of His grace and mercy. I sincerely pray that my work may be acceptable to Him and continue to glorify His name.

I would like to acknowledge and thank many people who have helped me successfully complete this thesis. Most notably, Dr. Adel Ghandakly, chair of the EECE Department, who has been my professor, advisor, and mentor throughout my graduate career. My thesis would not have seen the light of completion had it not been for his vital technical guidance and consistent encouragement.

I would like to extend my sincere thanks to my thesis committee member, Professor Dale Word, for taking time out to review the technical aspects of this research and for providing valuable suggestions. I also want to thank all my professors for being a true source of inspiration and for embracing me with their wisdom and experience. I am extremely grateful to all my wonderful friends, especially Rostan, Tian, Jowel and Pradeep, who have made graduate school so memorable and rewarding. I would also like to thank my special friend, Lourdes for her love and support.

Finally, I would like to thank my mom, Hilda and my brother, James for all their hard work and sacrifices. My opportunities in life are made possible because of them.
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>TABLE OF CONTENTS</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dedication .........................................................................................................</td>
<td>iii</td>
</tr>
<tr>
<td>Acknowledgments ...............................................................................................</td>
<td>iv</td>
</tr>
<tr>
<td>List of Tables ..................................................................................................</td>
<td>vii</td>
</tr>
<tr>
<td>List of Figures ..................................................................................................</td>
<td>viii</td>
</tr>
<tr>
<td>Abstract ............................................................................................................</td>
<td>xi</td>
</tr>
</tbody>
</table>

## CHAPTER

I. Introduction ........................................................................................................ 1
   - Background ...................................................................................................... 1
   - Problem Statement .......................................................................................... 4
   - Purpose of Study ............................................................................................ 5
   - Thesis Outline ............................................................................................... 6

II. Background ........................................................................................................ 8
   - Control of Nonlinear Systems ........................................................................ 8
   - System Identification ..................................................................................... 15
   - Multivariable Control - MIMO Systems ........................................................ 19
   - Adaptive and Optimal Control ........................................................................ 20
   - 6-DOF High Performance Aircraft Model ....................................................... 27
   - The Case for Aircraft Adaptive Control ......................................................... 35

III. Adaptive Controller Development ................................................................... 43
   - System Identification Algorithm ................................................................... 43
   - Adaptive and Optimal Control Implementation ............................................. 68
<table>
<thead>
<tr>
<th>CHAPTER</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>IV. Application to the 6-DOF Aircraft Model</td>
<td>83</td>
</tr>
<tr>
<td>Overall System Model</td>
<td>83</td>
</tr>
<tr>
<td>Aircraft Control Simulation Investigation</td>
<td>90</td>
</tr>
<tr>
<td>V. Conclusions</td>
<td>108</td>
</tr>
<tr>
<td>References</td>
<td>112</td>
</tr>
<tr>
<td>Appendix</td>
<td></td>
</tr>
<tr>
<td>A. List of Simulation Programs</td>
<td>116</td>
</tr>
</tbody>
</table>
# LIST OF TABLES

<table>
<thead>
<tr>
<th>TABLE</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Mathematical Symbols for Aircraft Motion Equations</td>
<td>30</td>
</tr>
<tr>
<td>2. Aircraft Control Inputs</td>
<td>32</td>
</tr>
<tr>
<td>3. Aircraft Test Cases 1-4 Trim States</td>
<td>37</td>
</tr>
<tr>
<td>4. Relative Forgetting Factor ($\lambda$) Values</td>
<td>56</td>
</tr>
<tr>
<td>5. Reference Commands for Aircraft</td>
<td>89</td>
</tr>
<tr>
<td>6. Aircraft Operating Points Trim Conditions</td>
<td>93</td>
</tr>
<tr>
<td>7. Reference Commands for Coordinated Maneuvers</td>
<td>97</td>
</tr>
<tr>
<td>8. Reference Commands for Complex Maneuver</td>
<td>105</td>
</tr>
</tbody>
</table>
# LIST OF FIGURES

<table>
<thead>
<tr>
<th>FIGURE</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Illustration of System Identification</td>
<td>16</td>
</tr>
<tr>
<td>2. MIMO System Architecture</td>
<td>19</td>
</tr>
<tr>
<td>3. Adaptive System Architecture</td>
<td>21</td>
</tr>
<tr>
<td>4. Self Tuning Regulator (STR) Block Diagram</td>
<td>24</td>
</tr>
<tr>
<td>5. Aircraft Motion Axis</td>
<td>28</td>
</tr>
<tr>
<td>6. Positive Control Deflection for Control Surfaces</td>
<td>32</td>
</tr>
<tr>
<td>7. Aircraft Simulation Model Structure</td>
<td>34</td>
</tr>
<tr>
<td>8. Aircraft Model Simulation Results</td>
<td>35</td>
</tr>
<tr>
<td>9. 3-D Trajectory of Aircraft Motion</td>
<td>36</td>
</tr>
<tr>
<td>10. Aircraft Process Linearization and Controller Design</td>
<td>39</td>
</tr>
<tr>
<td>11. Servo Mode Optimal Controller for Aircraft</td>
<td>39</td>
</tr>
<tr>
<td>12. Case-4 Controller Response to Ref. Command</td>
<td>40</td>
</tr>
<tr>
<td>13. Ref. Command Response with and Without Controller Parameters Adjustment</td>
<td>41</td>
</tr>
<tr>
<td>14. System Identification Procedure</td>
<td>45</td>
</tr>
<tr>
<td>15. MIMO System Identification Implementation</td>
<td>62</td>
</tr>
<tr>
<td>16. Process Input for Identification</td>
<td>63</td>
</tr>
<tr>
<td>17. Y1 - Identification Results</td>
<td>64</td>
</tr>
<tr>
<td>FIGURE</td>
<td>PAGE</td>
</tr>
<tr>
<td>--------</td>
<td>------</td>
</tr>
<tr>
<td>18. Y2 - Identification Results</td>
<td>65</td>
</tr>
<tr>
<td>19. Y3 - Identification Results</td>
<td>66</td>
</tr>
<tr>
<td>20. Convergence of Parameter Estimates</td>
<td>67</td>
</tr>
<tr>
<td>21. Process Switching Framework for MIMO Identification</td>
<td>68</td>
</tr>
<tr>
<td>22. Results of Identification with Process Dynamics Variation</td>
<td>69</td>
</tr>
<tr>
<td>23. System Identification with Different $\lambda$ Values</td>
<td>70</td>
</tr>
<tr>
<td>24. Identification Errors with Different Values of Forgetting Factor: Transient Errors (Left), Steady State Errors (Right)</td>
<td>71</td>
</tr>
<tr>
<td>25. System Identification Results with VFF</td>
<td>72</td>
</tr>
<tr>
<td>26. Value of $\lambda$ During Identification with VFF</td>
<td>73</td>
</tr>
<tr>
<td>27. Identification Error Analysis with Constant and Variable $\lambda$: Transient Errors (Left), Steady State Errors (Right)</td>
<td>74</td>
</tr>
<tr>
<td>28. Optimal Regulator Structure</td>
<td>75</td>
</tr>
<tr>
<td>29. Block Diagram of Servo Control Configuration</td>
<td>75</td>
</tr>
<tr>
<td>30. Self Tuning Regulator Implementation</td>
<td>78</td>
</tr>
<tr>
<td>31. Regulator Response to Disturbance</td>
<td>80</td>
</tr>
<tr>
<td>32. Regulator Performance with Variation in Process Dynamics</td>
<td>81</td>
</tr>
<tr>
<td>33. Servo Control Performance of Designed STR</td>
<td>82</td>
</tr>
<tr>
<td>34. MIMO System Identification of Aircraft</td>
<td>84</td>
</tr>
<tr>
<td>35. Results for MIMO System Identification of Aircraft</td>
<td>86</td>
</tr>
<tr>
<td>36. MIMO STR in Regulator Mode for Aircraft Control Application</td>
<td>87</td>
</tr>
<tr>
<td>37. Regulator Mode Results for Aircraft with Disturbance</td>
<td>88</td>
</tr>
<tr>
<td>FIGURE</td>
<td>PAGE</td>
</tr>
<tr>
<td>--------</td>
<td>------</td>
</tr>
<tr>
<td>38. Servo Control Configuration of MIMO STR for Aircraft Control</td>
<td>89</td>
</tr>
<tr>
<td>39. Servo Control Results for Aircraft Control Problem</td>
<td>90</td>
</tr>
<tr>
<td>40. 3-D Trajectory of Aircraft Motion During Servo Control Evaluation</td>
<td>91</td>
</tr>
<tr>
<td>41. Aircraft Output Response Using MIMO STR for Case-4</td>
<td>94</td>
</tr>
<tr>
<td>42. Performance for Multiple Operating Points Scenario</td>
<td>95</td>
</tr>
<tr>
<td>43. Case-5 and 6 Multiple Reference Commands and Operating Points</td>
<td>96</td>
</tr>
<tr>
<td>44. Case 4 and 10 Multiple Reference Commands and Operating Points</td>
<td>98</td>
</tr>
<tr>
<td>45. Case 8 and 2 Multiple Reference Commands and Operating Points</td>
<td>99</td>
</tr>
<tr>
<td>46. Flight-1 Maneuver with Case-2 Flight Conditions</td>
<td>100</td>
</tr>
<tr>
<td>47. Flight-2 Maneuver with Case-2 Flight Conditions</td>
<td>101</td>
</tr>
<tr>
<td>48. Flight-3 Maneuver with Case-4 Flight Conditions</td>
<td>102</td>
</tr>
<tr>
<td>49. Flight-4 Maneuver with Case-4 Flight Conditions</td>
<td>103</td>
</tr>
<tr>
<td>50. MIMO STR Performance for Coordinated Maneuver in Damaged Aircraft Case</td>
<td>104</td>
</tr>
<tr>
<td>51. 3-D Trajectory of Aircraft during Coordinated Maneuver</td>
<td>105</td>
</tr>
<tr>
<td>52. Results of Complex Maneuver with Reconfigured Aircraft Model</td>
<td>106</td>
</tr>
<tr>
<td>53. 3-D Trajectory of Reconfigured Aircraft Motion</td>
<td>107</td>
</tr>
</tbody>
</table>
ABSTRACT

A MULTI-INPUT MULTI-OUTPUT SELF TUNING REGULATOR FOR NONLINEAR HIGH PERFORMANCE AIRCRAFT CONTROL

by

Wilfred L. Ghonsalves

Master of Science in Electrical and Computer Engineering

Electronic Engineering Option

California State University, Chico

Summer 2010

This thesis presents the design and development of an adaptive control scheme for an application of high performance aircraft control. Traditional flight control design is based on linearization of an aircraft’s equations of motion around a set of operating points, and scheduling gains of linear controllers around each of these points to meet performance specifications. For the complex and aggressive maneuvers carried out by fighter aircraft, fixed gain or gain scheduled controllers do not guarantee robust performance and stability throughout the complete flight envelope. Therefore, there is a need for new design approaches to expand the flight envelope beyond the one achievable by linear controllers, attenuate failures, and to attain damage adaptation and recovery. Additionally, the variation in an aircraft dynamics during such aggressive maneuvers is poorly modeled.
Hence, an adaptive control scheme is an ideal solution to overcome such aerodynamics uncertainties.

In this thesis, a Multi Input Multi Output (MIMO) Self Tuning Regulator (STR) is proposed for the flight control problem. The approach uses MIMO system identification to capture process dynamics in the form of a linear model structure. Further, online design calculations are performed on an identified process model to design an optimal controller. This automated sequence of system identification and controller design is performed on each update interval using real-time recursive computations in order to adapt the controller parameters.

The proposed control design methodology is applied on a nonlinear 6-DOF high performance aircraft simulation model to evaluate its performance. Simulation results show that MIMO STR provides improved performance over linear controllers.
CHAPTER I

INTRODUCTION

Background

High performance aircraft control poses a unique challenge to a control designer due to stringent performance requirements, system complexities and large variations in dynamics over the complete flight envelope. The performance requirements may vary as functions of flight conditions such as aircraft's speed and altitude and/or pilot commands. Additionally, for an extremely maneuverable aircraft, meeting performance specifications can be particularly challenging because of difficulties involved in modeling or estimating the nonlinearities associated with such an aircraft.

A common method of dealing with variations in dynamics and performance objectives is to schedule the controller using relevant parameters. A linearized model of an aircraft operated at a specific flight condition is usually used to design a controller. Linear controllers designed using this method provide good performance near the design point. In order to guarantee the desired performance over a wide range of flight envelope, the gain scheduling technique is used on multiple linear controllers designed for different operating points [1], [2].

However, designing such multiple point controllers is a tedious and time consuming process also; the interpolation between the point controllers does not guarantee robust stability and good performance for the full flight envelope [3], [4].
Additionally, this method does not account for variations in process dynamics over the flight envelope and aircraft modeling errors.

Another approach deals with aircraft control problem by decoupling longitudinal and lateral aircraft dynamics in order to simplify control law design. In case of current advanced fighter aircrafts, the assumption that longitudinal and lateral dynamics can be decoupled may not hold true. Because of strong longitudinal-lateral cross-coupling, this approach may lead to poor flying and handling characteristics of the aircraft. To overcome the shortfalls of these traditional flight control approaches, alternate robust nonlinear controllers have been presented in control literature.

The research done in the area of flight control has been vigorous these past years and there are various control design techniques being used by the industry to design aircraft controllers. These techniques include Proportional-Integral (PI) control, Optimal Linear Quadratic Regulator / Linear Quadratic Gaussian (LQR/LQG) control, $\mu$ synthesis robust control, $H_\infty$ control, dynamic inversion, neural network, adaptive control and Linear Parameter Varying (LPV) control [5]. This clearly indicates that the industry has embraced advanced multivariable control techniques that are now the standard for designing flight control laws for advanced high performance aircraft.

The above mentioned techniques have been the subject of research these past few years and have provided vital contributions to the field of flight control performance. For instance, the Herbst like maneuver performed on a nonlinear fighter aircraft demonstrates the robustness of a radial basis neural network based controller [7]. On the other hand, a Model Reference Adaptive Control (MRAC) based adaptive controller is designed for aircraft pitch rate tracking using multiple fuzzy reference model technique
An alternate approach augments multiple modern control tools such as neural network, fuzzy logic, $\mu$ synthesis and H. control to perform Herbst like complex maneuver [9]. A combination of stochastic robustness and dynamic inversion is also proposed for high performance aircraft control application in research literature [10]. Sliding Mode Control of Pitch Rate for an F-16 is proposed in [11]. An adaptive controller design method based on back-stepping neural network is proposed for reconfigurable flight control systems in the presence of variations in aerodynamic coefficients or control effectiveness deficiencies caused by control surface damages [12].

This research is based on an adaptive control scheme known as Self Tuning Regulator (STR). A Multi Input Multi Output (MIMO) STR design technique is proposed for a class of multivariable linear/nonlinear systems. Any control system design activity passes through series of tasks such as process model building, selection of controller structure and parameterization of controller. The MIMO STR presented here is aimed at automating this multitask oriented control design activity through real time online recursive computations. This is a two stage activity. At the first stage, process input-output observations are used to build a process model through the system identification procedure. The identified process model is linear in nature and represents the finite dynamics of the process. In the second stage, the controller design procedures are performed on the identified model using optimal control theory. This sequence is performed using real-time online computations at each update interval in order to adapt controller parameters to deal with nonlinearity and variation in process dynamics.

In order to evaluate performance of MIMO STR algorithm, an adaptive controller is designed for high performance aircraft control application using the
proposed design methodology. Performance of the designed control scheme is compared with that of multiple linear controllers designed for particular flight conditions. Additionally, controller adaptation, robustness, recovery and survivability for aggressive maneuvers are studied with a case of damaged aircraft scenario for the full flight envelope.

Problem Statement

This thesis presents the synthesis and development of an adaptive control scheme for the class of nonlinear multivariable processes using MIMO STR. Consider a nonlinear system in the discrete time form as given in [13],

\[
x(k + 1) = f(x(k)) + g(x(k))U(k) \quad \ldots x(0) = 0
\]

\[Y(k) = h(x(k))\]  \hspace{1cm} (1)

where \(k\) is discrete-time index, \(x(k) \in \mathbb{R}^n\) is the internal system state vector, \(U(k) \in \mathbb{R}^m\) is control input and \(Y(k) \in \mathbb{R}^p\) is measured system output. \(f(x(k)) \in \mathbb{R}^n, g(x)k \in \mathbb{R}^{n \times m}\) and \(h(x)k \in \mathbb{R}^{m \times n}\) are real valued nonlinear functions that map system states, control inputs and system outputs.

The control objective is to determine a control input vector \(U(k)\), such that process output \(Y(k)\) tracks a given set of bounded smooth reference inputs \(R = [r_1, \ldots, r_m]\), while all other error signals remain bounded. The proposed control scheme will be applied on a 6-DOF high performance aircraft model to evaluate its effectiveness.
Purpose of Study

In the current scenario, for commercial and military aircraft, along with flying and handling qualities, performance qualities like agility, maneuverability, controllability, stability and safety have received increased emphasis. The aircraft must exhibit acceptable handling qualities while performing maneuvers [4]. To attain the desired performance and flying qualities, there is a need to develop robust, computationally and numerically efficient reconfigurable controllers to accommodate variations in process dynamics, modeling errors and other unidentified parameters. These will make it possible to expand the flight envelope, attenuate failures and attain damage adaptation and recovery.

Traditional approach in flight control makes use of gain scheduling scheme in the controller which has shortfalls in terms of good performance only within limited flight envelope and lack of mechanism to accommodate process dynamics uncertainty/variation during complex maneuvers. On the other hand, many modern nonlinear control laws propose solution of nonlinear partial differential equations (e.g. Hamiltonian-Jacobi-Bellman -HJB) using approximation methods to deliver optimal solution [4], [14]. Further, this results in a computation intensive implementation due to its iterative nature of computations.

In this research, a design methodology is investigated for a class of multivariable nonlinear systems. The proposed technique is based on computationally efficient real-time recursive online calculations. It incorporates controller parameter adaptation mechanism in order to reconfigure the controller while passing through multiple operating flight conditions throughout the entire flight envelope.
Thesis Outline

Chapter II presents an overview of control design tools and aspects associated with the design methodology proposed in this thesis. Fundamental understanding of nonlinear process control problem along with multivariable control design essentials is introduced. Further, dynamics of a 6-DOF high performance aircraft (F-16) is presented in the form of mathematical equations and its simulation framework is discussed. Development of simulation model of F-16 aircraft using MATLAB™ and Simulink™ environment is discussed in details. This developed model will be used in successive chapters to carry out various performance assessments. Further, the need for an adaptive controller for the chosen aircraft control application is demonstrated.

Chapter III describes synthesis and development of an adaptive controller in the form of a MIMO STR. Discussion in this chapter deals with Recursive Least Square Estimation (RLS) based MIMO system identification and design of an optimal controller in the regulator as well as in servo control configuration. The complete design procedure is based on real-time online computations. Initially, system identification yields a parametric model of process that covers finite dynamics of the process under control. Further, using optimal control theory, a controller design procedure is carried out on identified model to implement control law. Assessment of proposed design is presented after mathematical analysis.

Chapter IV presents an application of MIMO STR for high performance aircraft control problem. Complete simulation setup and results are presented with simulations in MATLAB™. Further, study of complex maneuvers are performed on aircraft model using the proposed control scheme. Robustness and survivability of the
aircraft under control is also verified through a damaged aircraft scenario. Performance of the designed MIMO STR is compared with that of the controller designed for particular operating conditions of aircraft which does not have parameter adaptation mechanism. The most prominent feature of proposed design technique is to accommodate any dynamic process without knowing its mathematical model. The same is demonstrated by reconfiguring the aircraft output vector.

Finally, Chapter V contains conclusions from the results of simulations performed to assess the performance of proposed design methodology.
CHAPTER II

BACKGROUND

This chapter presents an overview of control design tools and aspects associated with the design methodology proposed in this thesis. Fundamental understanding of nonlinear process control and multivariable control design essentials are introduced. A 6-DOF mathematical model for high performance aircraft and its simulation framework is discussed. In order to test performance of the proposed control design technique, this aircraft model will be used to frame nonlinear controls problem. Finally, need for an adaptive controller for chosen aircraft control application is demonstrated.

Control of Nonlinear Systems

Almost all physical systems exhibit nonlinearity. Potential sources of such nonlinearities can range from imperfectly made elements, energy limitations or special nonlinear elements which are added intentionally to improve the system's dynamic properties. Unlike nonessential nonlinearities, essential nonlinearities cannot be neglected as those directly influence system's dynamics. The amount of nonlinearities present in a system and desired performance calls for specific control design tools. Following discussion deals with methods and techniques used for nonlinear controls applications.
In many engineering, industrial and process-control applications where process nonlinearity is mild and system performance specifications are not demanding, the system can be approximated in a linear form and further, linear control or classical PID control can provide satisfactory performance. However, certain performance based critical applications such as aircraft control and complex robot manipulators which exhibit large nonlinearities, requires use of nonlinear control design techniques to achieve desired performance [13].

Dynamics of such systems are described by nonlinear mathematical equations which are difficult to solve. In highly nonlinear systems, simple change in one part of the system produces complex effects throughout. Known mathematical tools and methods do not give us powerful means to attack such nonlinear problem in order to get an exact solution. Therefore, contrary to linear systems, a general theory for all nonlinear systems does not exist. Instead, much more complex mathematical tools such as differential geometry, functional analysis and theory of nonlinear differential equations are used. A general solution for nonlinear system can be found only for Bernouilli and Riccati equations [15]. Excluding these special cases, for most nonlinear systems, individual procedures are used which are often too complex and improper for engineering practice. This fact is the main reason for existence of many approximate procedures in nonlinear control theory. Design of nonlinear control system presents following two major tasks: System Analysis: Theoretical and experimental research is carried out to find out the properties of the system or mathematical model of the system under investigation. System identification is one tool which is generally used to find out behavior of the system in terms of mathematical equations.
Control Synthesis: Synthesis deals with determining control system elements, structure of the controller and parameterization of selected controller structure in order to achieve desired performance. There is no unique solution for this problem in general for all nonlinear systems. Technical realization of the selected controller structure is an important aspect of this problem and many a times it leads to compromise between implementation, computational complexity and performance.

Use of nonlinear control design techniques comes with twofold difficulties. These methods are not so easily understood nor are they easy to structure as a routine implementation and require advanced mathematical tools. Considering these facts, there exist multiple ways to attack the problem of nonlinear control. Some of them indirectly use principle of linear controls, some try to get exact solution for nonlinear controls by approximation procedures and some use augmentation of both. Consider a discrete-time nonlinear dynamical system given in [13] in the form,

\[ x(k + 1) = f(x(k)) + g(x(k))U(k) \]
\[ Y(k) = h(x(k)) \] (2)

where \( k \) is discrete-time index, \( x(k) \in \mathbb{R}^n \) is the internal system state vector, \( U(k) \in \mathbb{R}^m \) is control input and \( Y(k) \in \mathbb{R}^p \) is measured system output. \( f(x(k)) \in \mathbb{R}^n, g(x) \in \mathbb{R}^{n \times m} \) and \( h(x) \in \mathbb{R}^{m \times n} \) are real valued functions which map system states, control inputs and system outputs and can be expressed as follows,

\[
\begin{bmatrix}
    f_1(x_1, x_2, \ldots, x_n) \\
    f_2(x_1, x_2, \ldots, x_n) \\
    \vdots \\
    f_n(x_1, x_2, \ldots, x_n)
\end{bmatrix}
\]
As stated earlier, general solution for all nonlinear systems does not exist. Depending upon nonlinearity involved in the process, one can represent nonlinear process in (2) by a linear mathematical representation. This method uses the process of linearization to find linear approximation of given nonlinear system about a single equilibrium state. Assuming feasibility of such representation for (2), it can be expressed in a linear state space model form as:

$$g(x(k)) = \begin{bmatrix}
g_{11}(x_1, x_2, ..., x_n) & g_{12}(x_1, x_2, ..., x_n) & \cdots & g_{1m}(x_1, x_2, ..., x_n) \\
g_{21}(x_1, x_2, ..., x_n) & g_{22}(x_1, x_2, ..., x_n) & \cdots & g_{2m}(x_1, x_2, ..., x_n) \\
\vdots & \vdots & \ddots & \vdots \\
g_{n1}(x_1, x_2, ..., x_n) & g_{n2}(x_1, x_2, ..., x_n) & \cdots & g_{nm}(x_1, x_2, ..., x_n)
\end{bmatrix}$$

$$h(x(k)) = \begin{bmatrix}
h_{11}(x_1, x_2, ..., x_n) & h_{12}(x_1, x_2, ..., x_n) & \cdots & h_{1n}(x_1, x_2, ..., x_n) \\
h_{21}(x_1, x_2, ..., x_n) & h_{22}(x_1, x_2, ..., x_n) & \cdots & h_{2n}(x_1, x_2, ..., x_n) \\
\vdots & \vdots & \ddots & \vdots \\
h_{m1}(x_1, x_2, ..., x_n) & h_{m2}(x_1, x_2, ..., x_n) & \cdots & h_{mn}(x_1, x_2, ..., x_n)
\end{bmatrix}$$

As stated earlier, general solution for all nonlinear systems does not exist. Depending upon nonlinearity involved in the process, one can represent nonlinear process in (2) by a linear mathematical representation. This method uses the process of linearization to find linear approximation of given nonlinear system about a single equilibrium state. Assuming feasibility of such representation for (2), it can be expressed in a linear state space model form as,

$$x(k + 1) = Ax(k) + BU(k)$$

$$Y(k) = Cx(k)$$

(3)

where $A$, $B$ and $C$ are constant matrices that define system dynamics. At this point, designer can use linear control theory to solve the control problem. It is important to note that the linear model given by process of linearization is not a true representation of nonlinear process but a close approximation. For highly nonlinear processes, one linear model may not suffice to cover complete range of dynamics of the real system. In such cases, multiple linear models can be defined around different operating points in order to apply linear control theory. One such representation is given in (4) where $A_i$, $B_i$ and $C_i$ are system dynamics matrices around operating point $p$. 
Once 'p' number of linear representations is available, linear control theory is applied for each representation to yield 'p' number of controllers, one for each operating point. From the above discussion it is evident that, it is always simple and convenient to study dynamics of the nonlinear system by using its linear representation. Considering all advantages of this method, caution must be taken since all nonlinear systems cannot be substituted by their linear models. Linear representation of nonlinear system will only yield valid result if linear effects are dominant in that system. In the case, where nonlinear effects are dominant, this representation may lead to a wrong conclusion which may results in poor performance of the system under control. As an alternative, considering multiple linear representations for such system will result in large number of operating points. This makes the design process more cumbersome due to an increase in design calculations. Also, any variation in process dynamics will not be accounted by this method. Representation of nonlinear system by a structure consisting of a linear part and a nonlinear part is an alternate way to tackle the nonlinear controls problem. Under such representation, the large proportion of dynamics of real time system gets covered leading to a more accurate representation.

Nonlinear Control Theory

The following discussion deals with nonlinear controls method that tries to get exact solution for nonlinear controls problem by approximation procedures. Nonlinear implementation of control law is a critical feature of such design methods. Therefore, neural network and structures which exhibit nonlinear behavior are ideal candidates for

\[
x(k + 1) = A_i x(k) + B_i U(k)
\]
\[
Y(k) = C_i x(k) \quad i = 0, 1, \ldots, p
\]
such control law implementation. Current research in the field uses this method independently or augmenting it with linear control techniques [6], [9], [16]. This method is based on classical viewpoint of optimal control theory in which major thrust is given to optimize performance of a system. Performance of any system can be captured by a suitably built cost function and optimization of that cost function leads to appropriate control law. In other words this can be expressed as, finding value of performance index (cost function) associated with the system which indicates some measure of deviation of the system performance from its ideal behavior and designing a control law based on this index to minimize present deviation and hence value of performance index itself.

Suppose a nonlinear system defined in (2) is drift free and without loss of generality, \( x(0) = 0 \) is an equilibrium state and \( f(0) = 0 \) and \( g(0) = 0 \). There is a need to find the control action \( U(x_k) \) which minimizes the infinite-horizon cost function given as,

\[
V(x_k) = \sum_{n=k}^{\infty} x_n^T Q_1 x_n + u_n^T Q_2 u_n
\]

\[
= x_n^T Q_1 x_n + u_n^T Q_2 u_n + \sum_{n=k+1}^{\infty} x_n^T Q_1 x_n + u_n^T Q_2 u_n
\]

\[
= x_n^T Q_1 x_n + u_n^T Q_2 u_n + V(x_{k+1})
\]  

(5)

where \( Q_1 > 0 \in \mathbb{R}^{n \times n} \) and \( Q_2 > 0 \in \mathbb{R}^{m \times m} \) are weighting matrices for relative emphasis on system states and control inputs respectively. From Bellman's optimality principle, Discrete Time Hamilton-Jacobi-Bellman (DT HJB) equation formulation is given in [14] in terms of performance value function \( V^*(x_k) \) as,
DT HJB equation in (6) develops backward in time. The optimal control \( u^* \) for this formulation is given by,

\[
V^*(x_k) = \min u_k \left( x_k^T Q_1 x_k + u_k^T Q_2 u_k + V^*(x_{k+1}) \right)
\]

\[
= x_k^T Q_1 x_k + \frac{1}{4} \frac{\partial V^*(x_{k+1})}{\partial x_{k+1}} g(x_k) Q_2^{-1} g(x_k)^T \frac{\partial V^*(x_{k+1})}{\partial x_{k+1}} + V^*(x_{k+1})
\]

This equation reduces to the Riccati equation in the Linear Quadratic Regulator (LQR) case, which can be efficiently solved. In the general nonlinear case, the Hamilton-Jacobi-Bellman (HJB) equation cannot be solved exactly since it involves solving either nonlinear partial differential or difference equations making it difficult to solve. Considering this fact, recursive methods are employed to obtain the solution of the HJB equation indirectly. In current nonlinear control literature, solution for such a difficult equation is computed based on close approximation properties. In order to do so, various levels of complex nonlinear structures are being used to represent performance cost function \( V(x_k) \) and then recursive methods are used to reduce this associated cost by finding close approximate control law value \( u(x_k) \). Most of the literature dealing with this method uses neural networks and exploit their "universal function approximation property" to compute tractable solutions to the nonlinear optimal control problems [14].

In conclusion, a general design methodology that is applicable to all nonlinear systems does not exist and also, substantial complex mathematical tools are required for
such control designs. But, an advantage which a nonlinear control design has over classical linear control is that the fundamental nonlinearities of the plant to be controlled are taken directly into account rather than ignored or neglected.

System Identification

System identification is the theory of building mathematical models of dynamical systems and signals based on observed inputs and outputs [17]. It is an important tool used extensively in the field of control engineering to estimate parameters of system models in order to understand its dynamics and to design controller by using appropriate control design tools. The beginning of all model-building is the observed data. For a dynamic system with the input at discrete time index ‘K’ denoted by U(k) and corresponding output denoted by Y(k), the data will be a finite collection of observations for a length of N.

$$Z^N = \{ \{U(0), Y(0)\}, \{U(1), Y(1)\}, \ldots, \{U(N), Y(N)\} \}$$

for a Multi Input Multi Output (MIMO) system, U(k) and Y(k) would be vectors. The system identification problem is then to find a mathematical relationship between the input sequence $U = \{U(0), U(1), \ldots, U(N)\}$ and the output sequence $Y = \{Y(0), Y(1), \ldots, Y(N)\}$, based on $Z^N$.

Simple illustration of system identification is shown in Figure 1 as a line fitting exercise. In this example, given X-Y points represents observed data. From given X-Y data, line fitting is performed on the basis of model for line equation ‘y = M x+C’. This equation can be correlated with system model for identification where M and C are
unknown parameters and the goal of identification process is to find their values. There are four key elements of system identification which are as follows:

1. Data set: The input-output data set is an entity on which relation between input and output of a system is modeled in mathematical form. The user may determine which signals to measure and when to measure them (sampling instance and time). Generally these parameters are selected by considering information contained in a data set. The designer has to take care that dataset measured has enough information to capture necessary dynamics of the system.

2. Selection of the model structure: This is critical and at the same time, the most difficult choice for system identification procedure. Choice of a specific model is an

Fig. 1. Illustration of system identification.
outcome of thorough analysis of the system under investigation as well as combination of prior knowledge and engineering insights.

3. Parameter estimation technique: It is a system identification method. Selected method manipulates the available dataset in order to parameterize the model structure. The assessment of model quality is typically judged by how the model performs when they attempt to reproduce the measured data.

4. Validation: Once the designer is settled on the preceding three choices, system identification process yields a parameterized model of the system i.e. the one in the set of parameters that best describes the data based on chosen criterion. An important question remains about it is whether this model is valid for its purpose. Model validation test justifies this answer using various procedures to assess how the model relates to observed data.

It is important to note that a model can never be a true description of the system. Instead, it can at best be considered as a good enough description of certain aspects that are of particular interest to user.

On identification Methods

There are various system identification methods such as Least Square Estimation (LSE), Extended least square, Maximum likelihood, Stochastic least squares and Subspace Identification. Each of them has different characteristics in terms of implementation complexity and computation time required for identification procedure. Control literature is dominated with LSE applications due to its simple implementation and computational advantage considering its recursive version i.e. Recursive Least Square (RLS). Current research in the field of controls has also shown an increased
interest in the subspace identification [18] because of its capability to accommodate MIMO system architecture without any further need of adaptation. In proposed control design technique, system model is continuously updated with each sample time. Additionally, emphasis is given to make control design process computationally efficient and therefore recursive version of LSE (RLS) is chosen for system identification. According to principle of least squares, the unknown parameters of a mathematical model should be chosen such that the sum of the squares of the differences between the actually observed and computed values, multiplied by the numbers that measure the degree of precision, is a minimum [17].

The model structure selected for system identification dictates the form of system representation i.e. linear or nonlinear. The identification process is simplified significantly if the model structure is linear in nature. An approximate representation of the nonlinear system can also be provided by finite numbers of linear models to capture finite range of nonlinear dynamics. In such cases if the model structure is maintained over multiple linear representations then, only model parameters needs to be adapted time to time to define nonlinear system behavior about an operating point or at a time instance. This principle is very useful while representing nonlinear systems with linear structure so as to use linear control theory on nonlinear control problems. However, it is important to check validity of such representation to ensure desired performance of the system under control. Also, it is important to note that multiple linearized model representation may not hold true for all class of nonlinear systems especially if systems have large nonlinearities. For such cases, system identification methods using nonlinear model representation exists in control literature [4] for which, the user has to apply nonlinear
control theory for controller design since the system is described in the form of a nonlinear model by system identification procedure.

**Multivariable Control - MIMO Systems**

*Multi Input Multi Output (MIMO)* or multivariable process with *m*-inputs (manipulated variables) and *p*-outputs (controlled variables) can be defined by basic transfer function model $Y = G(s)U$ where control input $U$ is a $m \times 1$ vector, process output $Y$ is a $p \times 1$ vector and $G(s)$ is a $m \times p$ matrix. Multivariable systems have more than one control loop and their input-output models may assume number of structural forms. Figure 2 shows one form of representation for MIMO system with $m=p=2$.

![MIMO system architecture.](image)

In the block diagram shown, $G_{11}(s)$ represent the forward path dynamics between $u_1$ and $y_1$, while $G_{22}(s)$ describes how $y_2$ responds after a change in $u_2$. The interaction effects are modeled using transfer functions $G_{21}(s)$ and $G_{12}(s)$. $G_{21}(s)$ describes how $y_2$ changes with respect to a change in $u_1$ while $G_{12}(s)$ describes how $y_1$
changes with respect to a change in $u_2$. Input output relation can be expressed in the matrix form as,

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} G_{11}(s) & G_{12}(s) \\ G_{21}(s) & G_{22}(s) \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}$$ \hspace{1cm} (9)

The interaction or coupling effect between manipulated variables ($U$) and controlled variables ($Y$) presents a challenge for control system design. There are multiple practices in controls to deal with this effect. One method is to choose a manipulated variable-controlled variable pair so that system interactions are minimized. Other alternative is to design a non-interacting or decoupling control scheme. In decoupling control scheme, multivariable process is decomposed into series of independent single-loop subsystems. If such decomposition can be achieved, multivariable process can be controlled using independent loop controllers. The amount of coupling present indicates the feasible method to solve the control problem for such systems [19]. In case of a large coupling effect where above mentioned methods cannot work, there is a need to design a controller with interactive control scheme using advanced techniques.

Adaptive and Optimal Control

An overview of adaptive and optimal control theory will be presented in this section. Optimal control based Self Tuning Regulator (STR) adaptive scheme is the heart of proposed thesis.

Adaptive Control

An adaptive control theory is used to design a control system that can modify its behavior in response to changes in the dynamics of the process. An adaptive controller
is defined as a controller with adjustable parameters and a mechanism for adjusting the parameter. Another definition defines adaptive controller as the combination of a parameter estimator, which generates parameter estimates online with a control law in order to control classes of plants whose parameters are completely unknown and/or could change with time in an unpredictable manner [17].

An adaptive controller is nonlinear in nature because of its parameters adjustment mechanism. A block diagram for adaptive control system is shown in Figure 3. As shown in Figure 3, an adaptive control system can be considered as a control system with two loops. One loop is normal feedback loop with plant and controller. The second loop is for parameter adaptation which automatically adjusts the controller parameters to accommodate changes in the process to be controlled or its environment.

![Fig. 3. A daptive system architecture.](image-url)
There are many factors which motivate use of adaptive control scheme for control of performance critical systems. Conventional linear feedback theory has a limitation to maintain desired performance of the system due to variation in process dynamics and disturbance characteristics. Additionally, linear control methods are sensitive to modeling errors and can degrade the performance of the system whenever there is a deviation between the mathematical model and the real system. Adaptive control research was initiated in 1950 while designing autopilots for high-performance aircraft. An aircraft operates in a wide range of altitudes and speeds. Additionally, their dynamics are highly nonlinear and time varying in nature. The following discussion elaborates case of an adaptive control using practical example of aircraft system.

As discussed earlier in the nonlinear controls overview section, an aircraft's nonlinear dynamics can be represented as a linear model by process of linearization about an operating point \( \mathbf{p} \) by using (4) and system dynamics matrices \( \mathbf{A}_p, \mathbf{B}_p, \) and \( \mathbf{C}_p \) as,

\[
x(k + 1) = \mathbf{A}_p x(k) + \mathbf{B}_p U(k) \\
Y(k) = \mathbf{C}_p x(k)
\]

(10)

As an aircraft goes through wide range of speed and altitude, its operating point changes which leads to change in values of matrices \( \mathbf{A}_p, \mathbf{B}_p, \) and \( \mathbf{C}_p \) indicating changes in process dynamics. Ideally, in this situation the controller needs to adjust its parameters to take care of variations in process dynamics. From automatic control perspective, under such circumstances there is a need of an adaptive system which detects changes in process dynamics and then adjust its controller parameters itself. There exist different methods of adaptive control system based on the choice of the parameter estimation technique, the choice of the control law and the way they are combined. Three of such adaptive systems
namely Gain Scheduling, Model Reference Adaptive Control (MRAC) and Self Tuning Regulator (STR) are discussed in following discussion. Proposed thesis is based on Self Tuning Regulator control scheme.

**Gain Scheduling**

In this method, change of controller parameters is accomplished by continuous monitoring of operating conditions of the process. Gain scheduling is a nonlinear control scheme which employs a linear controller and its parameters are changed as a function of process operating conditions in a preprogrammed manner. As an illustration, in an aircraft application discussed earlier, change of operating conditions could be detected by monitoring mach number (speed) and altitude of aircraft, and then use them as “scheduling variables” to accomplish the task. Whenever there is a significant variation in such monitored variables, gain scheduling control scheme will force changes in controller parameters based on precomputed values.

**Model Reference Adaptive Control (MRAC)**

As name suggests, this adaptive scheme uses a reference model which expresses desired response of the system under control. Output of the closed loop system is expected to match the output of the reference model in order to get desired response from the system. Any deviation of system output from expected value (model output) is considered as an error signal. Further, this error signal is used by parameter adjustment mechanism to adjust the parameters of the controller so as get desired performance from the real system irrespective of effects due to nonlinearities in the process and variation in process dynamics.
**Self Tuning Regulator (STR)**

Control system design procedure passes through series of tasks such as model building, selection of controller structure and parameterization of controller to get desired response from system under control. Self tuning regulator is an adaptive control scheme which is emphasized around online control design process. Once necessary framework is setup, STR automates this multitask oriented control design activity through real time online recursive computations.

The block diagram of STR scheme is shown in Figure 4. STR employs a two stage procedure for controller parameter adjustment. Initially process parameters are estimated which defines the updated process model and then this updated model is passed to an online control design stage which computes controller parameters to get desired response. Thus, unlike gain scheduling and model reference scheme, STR adapts
controller parameters indirectly by solving a control design problem online. STR also has two control loops. Ordinary feedback loop for closed loop system and second loop to take care of controller parameters adaptation by solving a control design problem. There are two critical tasks for designing a STR: 1) selection of an appropriate structure to represent system model in order to capture finite range of system dynamics, and 2) selection of a control design technique which can be automated through online computations. Once this framework is setup, STR continuously solves a control design problem and adjusts its controller parameters.

**Optimal Control**

The optimal control system can be defined as the system whose design optimizes the value of a function considered as performance index [20]. In designing an optimal control or regulator system, there is a need to find a law for determining the present control decision for process which is subject to the certain constraints so as to minimize some measure of deviation from the ideal behavior of process. In general, performance index is a function whose value is considered to be an indication of how well the system performance matches with desired performance and thus provides measure of such deviation. In order to minimize this performance index and hence deviation from desired response, the control input \(U(k)\) is chosen appropriately to optimize the system performance.

Optimal control of the linear system based on quadratic performance index is being used in practical systems as a measure of system performance. In the Linear Quadratic Regulator (LQR) control problem we desire to determine a law for the control vector \(U(k)\) such that a given quadratic performance index in (11) is minimized.
In (11), $X(k)$ is system state vector and $U(k)$ is control input. Matrix $Q_1$ is positive definite or positive semi definite Hermitian matrix and $Q_2$ is a positive definite Hermitian matrix. In the summation, the first term accounts for relative errors during control process and second term accounts for expenditure of energy of the control signals. $Q_1$ and $Q_2$ plays as weighting matrices which correlates measure of importance between process states and control energy.

The solution of an optimal control problem is to find control input vector $U(k)$ within allowable control vectors such that the performance index will be minimized to achieve optimized performance of the system under control. The designed optimization process needs to serve twofold purpose- first it should provide optimal control law and parameter configuration and second, it should also predict the degradation in system performance due to deviation of performance index from its minimum value. In general, optimality with respect to some criterion is not the only desirable property for a controller. One would also like stability of the closed-loop system, good gain and phase margins, robustness with respect to un-modeled dynamics etc. LQR is particularly interesting because the minimization procedure automatically produces controllers that are stable and somewhat robust. Moreover, this procedure is applicable to multiple-input/multiple-output processes for which classical designs are difficult to apply.

$$J = \frac{1}{2} \sum_{k=0}^{\infty} [X^T(k) Q_1 X(k) + U^T(k) Q_2 U(k)]$$

(11)
An adaptive control based design technique has been proposed in this thesis. The presented technique is applicable for class of multivariable processes which are linear or nonlinear in nature. An overview of multivariable processes and related control design tools has been discussed in earlier sections. In order to evaluate the performance of the proposed design technique, an aircraft control application is chosen as it presents a highly nonlinear multivariable process control problem. Aircraft control has been fascinating and challenging problem throughout control engineering community because of its large nonlinear dynamics. Mathematical model of a high performance fighter aircraft similar to an F-16 aircraft as reported in NASA technical report 1538 [21] is used to investigate performance of proposed design methodology.

A 6-DOF aircraft model is presented in [21] in the form of nonlinear differential equations along with various aerodynamics coefficients. The description of associated subsystem and its effect on aircraft characteristics is also presented in the report. Information about aircraft dynamics presented in [6], [21] and [22] is used to build simulation model of high performance aircraft using MATLAB™ and Simulink™. Aircraft motion axes are shown in Figure 5. From the control design perspective, current framework considers Roll, Pitch and Yaw angles of aircraft as outputs of multivariable process. In following discussion, mathematical model for aircraft will be introduced. Further, simulation model development process and simulation results will be presented. Considering rigid body assumption and fixed-body axis system, equations of motion for high performance aircraft are given as,
Fig. 5. Aircraft motion axis.


Force Equations:

\[ \dot{u} = rv - qw - g \sin \theta + \frac{\bar{q}S}{m} C_{x,t} + \frac{T}{m} \]  \hspace{1cm} (12)

\[ \dot{v} = pw - ru + g \cos \theta \cos \phi + \frac{\bar{q}S}{m} C_{y,t} \]  \hspace{1cm} (13)

\[ \dot{w} = qu - pv - g \cos \theta \sin \phi + \frac{\bar{q}S}{m} C_{z,t} \]  \hspace{1cm} (14)

Kinematic Equations:

\[ \dot{\phi} = p + \tan \theta \left( q \sin \phi + r \cos \phi \right) \]  \hspace{1cm} (15)

\[ \dot{\theta} = q \cos \phi - r \phi \]  \hspace{1cm} (16)

\[ \dot{\psi} = \left( q \sin \phi + r \cos \phi \right) / \cos \theta \]  \hspace{1cm} (17)
Moment Equations:

\begin{align*}
\dot{p} & = \frac{l_x - l_z}{l_x} qr + \frac{l_{xz}}{l_x} (\dot{r} + pq) + \frac{\bar{q} S b}{l_x} c_{l,t} + \frac{H_{e q}}{l_x} \\
\dot{q} & = \frac{l_x - l_z}{l_y} pr + \frac{l_{xz}}{l_y} (r^2 - p^2) + \frac{\bar{q} S c}{l_y} c_{m,t} - \frac{H_{e r}}{l_y} \\
\dot{r} & = \frac{l_x - l_y}{l_z} pq + \frac{l_{xz}}{l_z} (\dot{p} - qr) + \frac{\bar{q} S b}{l_z} c_{n,t} + \frac{H_{e q}}{l_z}
\end{align*}

(18) (19) (20)

Navigation Equations:

\begin{align*}
\dot{p}_N & = u \cos \theta \cos \psi + v (\cos \psi \sin \psi + \sin \phi \sin \theta \cos \psi) \\
& \quad + w (\sin \phi \sin \psi + \cos \phi \sin \theta \cos \psi) \\
\dot{p}_E & = u \cos \theta \sin \psi + v (\cos \phi \cos \psi + \sin \phi \sin \theta \sin \psi) \\
& \quad + w (\sin \phi \cos \psi + \sin \phi \sin \theta \sin \psi) \\
h & = u \sin \theta - v \sin \phi \cos \theta - w \cos \phi \cos \theta
\end{align*}

(21) (22) (23)

Auxiliary Equations:

\begin{align*}
\alpha & = \tan^{-1} \left( \frac{w}{u} \right) \\
\beta & = \sin^{-1} \left( \frac{v w}{V_T} \right) \\
V_T & = \sqrt{u^2 + v^2 + w^2}
\end{align*}

(24) (25) (26)

\begin{align*}
a_n & = \frac{qu - pv + g \cos \theta \cos \phi - \dot{w}}{g} \\
a_y & = \frac{-pw - ru + g \cos \theta \sin \phi + \dot{v}}{g}
\end{align*}

(27) (28)

Mathematical symbols describing the aircraft dynamics are shown in Table 1.
These aircraft dynamics equations are highly nonlinear and coupled. The aerodynamic coefficient $C_{X,t}, C_{Y,t}, C_{Z,t}, C_{l,t}, C_{m,t},$ and $C_{n,t}$ are expressed as equations which are functions of various other aerodynamics coefficients. These equations sum the various aerodynamic contributions to a particular force or moment. For example, the $X$ axis moment coefficient equation is given in (29).

Related aerodynamics coefficients with respect to aircraft surface angles, force and moments are presented in [21] in the form of tabular data. The aerodynamic data presented in report were derived from low-speed static and dynamic (force oscillation) wind-tunnel tests conducted with subscale models of the F-16 aircraft in wind-tunnel facilities at NASA, Ames and Langley Research Center. The static aerodynamics is presented in tabular form as function of both angle of attack (•) and
sideslip angle (•) over the ranges $-20^\circ \leq \alpha \leq 90^\circ$ and $-30^\circ \leq \beta \leq 30^\circ$. The dynamic data is presented in tabular form for $\bullet=0^\circ$ for $\bullet$ over the same range.

Simulation Model Development

The mathematical model described in (12)-(28) is implemented in MATLAB™ using modeling structures in [23]. The aircraft is powered by an afterburning turbofan jet engine. The model for the engine simulation is described as thrust responses to throttle inputs. The thrust values are given in tabular form for idle, military and maximum thrust values. There is no provision of thrust vectoring of control and hence the maneuver has to be restricted within the capabilities of the conventional control surfaces i.e. elevator, aileron and rudder. Also, for complex and critical maneuvers, the throttle is usually pushed to nearly full position to prevent significant loss of speed [6]. Therefore, in current simulation scenario, engine thrust control is not considered in the controller design and assumed to be constant.

Aircraft will be controlled over conventional control surfaces- elevator, aileron and rudder. In order to allow aircraft at higher angles of attack, an additional control surface known as “leading edge flap” has been incorporated in aircraft. Unlike conventional control surfaces; this control surface cannot be directly changed by the pilot. Instead, its deflection is managed by angle of attack and the static and dynamic

$$C_{t,t} = C_t(\alpha, \beta, \delta_n) + \Delta C_{t,\text{eff}} \left(1 - \frac{\delta_{\text{lef}}}{25}\right) + \left[ \Delta C_{t,\delta_n} + \Delta C_{t,\delta_n,\text{eff}} \left(1 - \frac{\delta_{\text{lef}}}{25}\right) \right] \frac{\delta_n}{20}$$

$$+ \Delta C_{t,\delta_r} \frac{\delta_r}{30}$$

$$+ \frac{b}{2V_t} \left[ C_t(\alpha) + \Delta C_{t,\text{eff}}(\alpha) \left(1 - \frac{\delta_{\text{lef}}}{25}\right) \right] r$$

$$+ \left[ C_{ip}(\alpha) + \Delta C_{ip,\text{eff}}(\alpha) \left(1 - \frac{\delta_{\text{lef}}}{25}\right) \right] p + \Delta C_{ip}(\alpha) \beta$$

(29)
pressure for which aircraft is flying. Maximum values and units for each control input are shown in Table 2 and positive orientation for each control surface is shown in Figure 6.

TABLE 2
AIRCRAFT CONTROL INPUTS

<table>
<thead>
<tr>
<th>Control</th>
<th>Unit</th>
<th>Min Value</th>
<th>Max Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Thrust</td>
<td>lbs</td>
<td>1000</td>
<td>19000 lbs</td>
</tr>
<tr>
<td>Elevator</td>
<td>deg</td>
<td>-25</td>
<td>25 deg</td>
</tr>
<tr>
<td>Aileron</td>
<td>deg</td>
<td>-21.5</td>
<td>21.5 deg</td>
</tr>
<tr>
<td>Rudder</td>
<td>deg</td>
<td>-30</td>
<td>30 deg</td>
</tr>
<tr>
<td>Leading Edge Flap</td>
<td>deg</td>
<td>0</td>
<td>25 deg</td>
</tr>
</tbody>
</table>

Fig. 6. Positive control deflection for control surfaces.

Complete simulation model was developed in a Simulink™ model file (aircraft.mdl). Architecture of implementation is shown in Figure 7. The input vector $U$, output vector $Y$ and state vector $X$ for implemented process model is given in (59). Along with symbols shown in Table 1, additional symbols in $X$ are normalized acceleration in x,y,z direction ($n_x, n_y, n_z$), mach number ($M$), dynamic and static pressure ($\bar{q}$ and $p_s$).

$$U = \begin{pmatrix} u_1 \\ u_2 \\ u_3 \end{pmatrix} \in \mathbb{R}^3 = \begin{pmatrix} \text{Elevator Surface} \\ \text{Aileron Surface} \\ \text{Rudder Surface} \end{pmatrix}$$

$$Y = \begin{pmatrix} y_1 \\ y_2 \\ y_3 \end{pmatrix} \in \mathbb{R}^3 = \begin{pmatrix} \text{Roll Angle} \\ \text{Pitch Angle} \\ \text{Yaw Angle} \end{pmatrix}$$

$$X = \begin{pmatrix} x_1(k) \\ x_2(k) \\ \vdots \\ x_{18}(k) \end{pmatrix} \in \mathbb{R}^{18} = \begin{pmatrix} p_N \\ p_E \\ h \\ \phi \\ \theta \\ \psi \\ V_T \\ \alpha \\ \beta \\ p \\ q \\ r \\ n_x \\ n_x \\ n_x \\ \bar{q} \\ p_s \end{pmatrix}$$

(30)

A MATLAB™ program (aircraftSimulation.m) was developed to initialize the simulation model, perform trimming based on initial conditions and to run the simulation.
The initial states of the aircraft were supplied by a trimming routine which was incorporated in this program. The trimming routine trims aircraft to a steady state flight for a given altitude and velocity. The aircraft can be initially trimmed in steady wing level, turning, pull up or roll flight. This simulation program was used to trim aircraft with turning rate = 2 deg/sec, altitude=10000 ft and velocity=600 ft/sec and then simulation was run for 5 sec. Figure 8 shows aircraft output angles, altitude, velocity and angle of attack. Figure 9 shows 3-D trajectory of aircraft during simulation.

An aircraft 6-DOF simulation model developed in this section will be used in the following chapters to test various control design frameworks. Finally, a MIMO STR based adaptive control scheme will be applied on this model to evaluate performance of proposed design technique.
The Case for Aircraft Adaptive Control

This section demonstrates need for an adaptive control scheme for chosen nonlinear controls application. Performance of high performance aircraft control problem will be evaluated using gain scheduling, a very basic form of an adaptive scheme against that of the constant gain controller. Further, results from this analysis will be presented in order to demonstrate need for an adaptive controller for the aircraft control problem.

As discussed in overview of adaptive control section, the controller with gain scheduling scheme demonstrates very basic form of adaptive behavior. It frames nonlinear control problem in linear framework and suggests use of linear control theory to solve the problem. In this approach, the nonlinear process is linearized around various
critical operating points to get linear models and then controller parameters are pre-computed for each linear model using manual design calculations.

For the aircraft control application, combination of aircraft speed and altitude can be considered as a good measure of operating point of an aircraft. The chosen aircraft application has large nonlinearities over its operating range. It has velocity range of 300 ft/Sec- 900 ft/Sec and altitude range of 5000 ft- 40000 ft. Nonlinear aircraft model was linearized around different altitude and velocity combinations referred to as operating points. Four different test cases were formed in order to cover finite dynamics and nonlinearity of aircraft. Prior to linearization, nonlinear aircraft process was trimmed at these selected operating points. Table 3 shows values of trim conditions for each operating point.

Fig. 9. 3-D trajectory of aircraft motion.
Linearization around different operating points yields linear models of the process in continuous time state space form given by (31).

\[
\dot{X}(t) = A_p X(t) + B_p U(t) \\
Y(t) = C_p X(t) + D_p U(t)
\]  

(31)

Matrices $A_p$, $B_p$, $C_p$, $D_p$ are system dynamics matrices related with operating point $p$ ($p = 1, 2, 3, 4$). Resulting linear aircraft model has 15 states i.e. $x(k) \in \mathbb{R}^{15}$. In order to frame control problem into discrete domain, sampling time of 20 mSec is considered and continuous state space model is transformed to discrete domain as,

\[
X(k + 1) = \phi_p X(k) + \gamma_p U(k) \\
Y(k) = G_p X(k) + H_p U(k)
\]  

(32)
Process output and control input is given by,

\[ Y(k) = \begin{pmatrix} y_1(k) \\ y_2(k) \\ y_3(k) \end{pmatrix} \in \mathbb{R}^3 = \begin{pmatrix} \text{Roll Angle} \\ \text{Pitch Angle} \\ \text{Yaw Angle} \end{pmatrix} \]

\[ U(k) = \begin{pmatrix} u_1(k) \\ u_2(k) \\ u_3(k) \end{pmatrix} \in \mathbb{R}^3 = \begin{pmatrix} \text{Elevator Surface} \\ \text{Aileron Surface} \\ \text{Rudder Surface} \end{pmatrix} \]

For discrete time linear process model expressed in (31), an optimal control framework was designed in servo mode configuration and for each operating point model set of controller parameters were found using optimal control theory. A MATLAB™ program (linearControllerDesignCase1to4.m) is developed to trim the aircraft at test operating points, obtain linearized representation of aircraft dynamics at each operating point and to compute controller parameters for servo control configuration using optimal control theory. Figure 10 shows process flow graph for entire procedure discussed.

The aircraft control problem is framed in servo mode configuration in order to accept reference commands by pilot. The structure of servo mode control scheme is shown in Figure 11. Linear control theory is applied on linearized aircraft model at each operating point to yield controller parameters. These parameters consists of state feedback gain (K) and integral gain (Ki). All four sets of computed controller parameters are arranged as lookup table entries in order to use them in gain scheduling scheme. As shown in control scheme, it is assumed that aircraft output vector Y(k) and all aircraft states (n=15) are measurable and available for control law implementation. A reference command test was performed using controller designed for operating point-4. Initially,
Fig. 10. Aircraft process linearization and controller design.

Fig. 11. Servo mode optimal controller for aircraft.
aircraft was trimmed for case-4, outputs were zeroed and then a reference command was applied on aircraft. Results are shown in Figure 12.

Fig. 12. Case-4 controller response to ref. command.

In order to test performance of case-4 controller for the variation in aircraft operating point, once the given reference command was achieved (as shown in Figure 12), aircraft altitude and speed were changed to match that of case-1 and then to case-3 to simulate change in aircraft operating conditions. The switching time of 5 sec was considered for this exercise. Another run was performed with similar sequence but using gain scheduling scheme. This time, along with change in altitude and velocity, controller designed for case-1 and case-3 was also introduced during switching time. A MATLAB™ program (linearControllerPerformance.m) is used to carry out complete simulation exercise discussed. The result of simulation is shown in Figure 13. It can be
Fig. 13. Ref. command response with and without controller parameters adjustment.

seen from the results that controller with lack of parameter adjustment shows degraded performance and unstable behavior of closed loop system. Whereas, gain scheduling scheme works very well for the given operating points provided there exists a gain switching mechanism based on monitored scheduled variables (i.e., aircraft velocity and altitude). Thus, it is evident that the controller with parameter adaptation is necessary to ensure stability of aircraft under control and to get desired performance.

An important point to note that we have only considered 4 operating points out of large operating range which represents a limited dynamics variation in the aircraft.
in motion. Thus, this analysis definitely indicates need of an adaptive control scheme to adapt controller parameters with respect to changes in system dynamics. For the chosen aircraft control problem, gain scheduling cannot be considered as an ideal solution since; there exist large number of operating points which need to be analyzed and respective gains needs to be computed beforehand which is a cumbersome task. The above analysis and discussion exhibits need of an adaptive control scheme to achieve desired performance from the system. Additionally, the employed adaptive scheme should ensure that system remains stable during complete duration of flight.

In summary, this chapter has presented overview of various controls tools associated with nonlinear controls problem. A mathematical model of high performance aircraft and its implementation in simulation framework is also discussed. Need of an adaptive control scheme for aircraft control problem is also demonstrated using gain scheduling test mechanism. Linearized models of aircraft presented in this section will be used in Chapter III to evaluate performance of designed control subsystems before applying them on nonlinear aircraft model.
CHAPTER III

ADAPTIVE CONTROLLER DEVELOPMENT

This chapter establishes complete design procedures for an adaptive controller in the form of a Multi Input Multi Output (MIMO) Self Tuning Regulator (STR). Discussion in this chapter deals with Least Square Estimation (LSE) based MIMO system identification and design of an optimal controller in regulator and servo control configuration. The complete design procedure is based on real-time online computations. Initially, system identification yields a parametric model that covers finite dynamics of the process under control. Further, using optimal control theory, a controller design procedure is carried out on identified model to implement control law. Assessment of proposed design procedures is presented after mathematical analysis.

System Identification Algorithm

As discussed in Chapter II, system identification provides parameter estimates in order to build the model of dynamic system. In the proposed design technique, the main idea lies in representing given linear/nonlinear process with a linear mathematical model that covers finite dynamics of the process and then to apply online control design procedures on an identified model using linear control theory. This operation is
continuously repeated at each sampling interval so as to cover large range of process
dynamics and nonlinearity.

Proposed design method is applicable for systems with process dynamics
variation and for most of the nonlinear control applications as well. Most of the nonlinear
systems can be represented by a linear mathematical model around an operating point by
process of linearization or by real time system identification procedure. This identified
model represents finite information about system dynamics at a given instant or around
an operating point and therefore cannot be considered as a true dynamic model of
complete system.

**LSE Identification for MIMO Systems**

The system identification overview has been presented in Chapter II. The real
time parameter identification concept for Single Input Single Output (SISO) systems
presented in [17] is extended and adapted for the case of Multi Input Multi Output
(MIMO) system.

The block diagram of MIMO system identification procedure is shown in
Figure 14. It is assumed that the system input and output vectors- U(k) and Y(k)
respectively are measurable and available for system identification and control law
implementation. A SISO system difference equation is extended to represent a linear
MIMO system with coupled inputs and outputs. Further, this difference equation is
expressed in a general parametric model structure which can be used by system
identification and control design procedures. The system identification technique
discussed in this section is aimed at estimation of parameters in model structure that
represents process under control at a given point of time.
Consider a Single Input Single Output (SISO) system of order \( n \) represented by its transfer function in the discrete time domain by,

\[
\frac{y(z)}{u(z)} = \frac{b_1 z^{-1} + b_2 z^{-2} + \ldots + b_N z^{-n}}{1 - a_1 z^{-1} - a_2 z^{-2} - \ldots - a_N z^{-n}}
\]  

(33)

Difference equation for system described in (33) is given by,

\[
y(k) = a_1 y(k-1) + a_2 y(k-2) + \ldots + a_n y(k-n) + b_1 u(k-1) + b_2 u(k-2) + \ldots + b_n u(k-n)
\]  

(34)

Similarly, an \( n^{th} \) order MIMO system with \( m \)-inputs \((u_1, u_2, \ldots, u_m)\) and \( m \)-outputs \((y_1, y_2, \ldots, y_m)\) can be represented in the form of difference equation as,
\( y_1(k) = a_{111}y_1(k - 1) + a_{112}y_2(k - 1) + \cdots + a_{11m}y_m(k - 1) + a_{211}y_1(k - 2) \\
+ a_{212}y_2(k - 2) + \cdots + a_{21m}y_m(k - 2) + \cdots + a_{n11}y_1(k - n) \\
+ a_{n12}y_2(k - n) + \cdots + a_{n1m}y_m(k - n) + \cdots + b_{111}u_1(k - 1) \\
+ b_{112}u_2(k - 1) + \cdots + b_{11m}u_m(k - 1) + b_{211}u_1(k - 2) \\
+ b_{212}u_2(k - 2) + \cdots + b_{21m}u_m(k - 2) + \cdots + b_{n11}u_1(k - n) \\
+ b_{n12}u_2(k - n) + \cdots + b_{n1m}u_m(k - n) \\
\)

\( y_2(k) = a_{121}y_1(k - 1) + a_{122}y_2(k - 1) + \cdots + a_{12m}y_m(k - 1) + a_{221}y_1(k - 2) \\
+ a_{222}y_2(k - 2) + \cdots + a_{22m}y_m(k - 2) + \cdots + a_{n21}y_1(k - n) \\
+ a_{n22}y_2(k - n) + \cdots + a_{n2m}y_m(k - n) + \cdots + b_{121}u_1(k - 1) \\
+ b_{122}u_2(k - 1) + \cdots + b_{12m}u_m(k - 1) + b_{221}u_1(k - 2) \\
+ b_{222}u_2(k - 2) + \cdots + b_{22m}u_m(k - 2) + \cdots + b_{n21}u_1(k - n) \\
+ b_{n22}u_2(k - n) + \cdots + b_{n2m}u_m(k - n) \\
\)

\[\vdots\]

\[\vdots\]

\[\vdots\]

\( y_m(k) = a_{1m1}y_1(k - 1) + a_{1m2}y_2(k - 1) + \cdots + a_{1mm}y_m(k - 1) \\
+ a_{2m1}y_1(k - 2) + a_{2m2}y_2(k - 2) + \cdots + a_{2mm}y_m(k - 2) \\
+ \cdots + a_{nm1}y_1(k - n) + a_{nm2}y_2(k - n) + \cdots + a_{nmm}y_m(k - n) + \cdots + b_{1m1}u_1(k - 1) + b_{1m2}u_2(k - 1) \\
+ \cdots + b_{1mm}u_m(k - 1) + b_{2m1}u_1(k - 2) + b_{2m2}u_2(k - 2) \\
+ \cdots + b_{2mm}u_m(k - 2) + \cdots + b_{nm1}u_1(k - n) \\
+ b_{nm2}u_2(k - n) + \cdots + b_{nmm}u_m(k - n) \]
Expressing (35) in general form as,

\[ Y(k) = A_1 Y(k-1) + A_2 Y(k-2) + \ldots + A_n Y(k-n) + B_1 U(k-1) + B_2 U(k-2) + \ldots + B_n U(k-n) \] (36)

In (36), \( U(k) \) and \( Y(k) \) are \( m \)-dimensional input and output vectors respectively and can be expressed as,

\[
Y(k) = \begin{pmatrix} y_1(k) \\ y_2(k) \\ \vdots \\ y_m(k) \end{pmatrix} \in \mathbb{R}^m \quad \text{and} \quad U(k) = \begin{pmatrix} u_1(k) \\ u_2(k) \\ \vdots \\ u_m(k) \end{pmatrix} \in \mathbb{R}^m
\]

and the system parameters matrices \( A_1, A_2, \ldots, A_n \) and \( B_1, B_2, \ldots, B_n \) are given by,

\[
A_i = \begin{bmatrix} a_{i11} & a_{i12} & \cdots & a_{i1m} \\ a_{i21} & a_{i22} & \cdots & a_{i2m} \\ \vdots & \vdots & \ddots & \vdots \\ a_{im1} & a_{im2} & \cdots & a_{imm} \end{bmatrix} \in \mathbb{R}^{m \times m} \quad i=1,2,\ldots,n
\]

\[
B_i = \begin{bmatrix} b_{i11} & b_{i12} & \cdots & b_{i1m} \\ b_{i21} & b_{i22} & \cdots & b_{i2m} \\ \vdots & \vdots & \ddots & \vdots \\ b_{im1} & b_{im2} & \cdots & b_{imm} \end{bmatrix} \in \mathbb{R}^{m \times m} \quad i=1,2,\ldots,n
\]

General model for \( n \)-th order linear MIMO system with coupled inputs and outputs is established in (36). This structure is used by system identification procedure that yields estimates of parameters in matrices \( A_i \) and \( B_i \). Least Square Estimation (LSE) technique is used for MIMO system identification procedure which is established in following discussion. Using (36), \( y_1(k) \) can be written as,
Equation (37) can be represented by a simplified mathematical model as,

\[ y_1(k) = \theta_{y1} \varphi_{y1}(k) + \theta_{y2} \varphi_{y2}(k) + \ldots + \theta_{yn} \varphi_{yn}(k) \]

\[ + \theta_{u1} \varphi_{u1}(k) + \theta_{u2} \varphi_{u2}(k) + \ldots + \theta_{un} \varphi_{un}(k) \]

\[ = \theta_1 \varphi^T(k) \]

Equation (37) can be represented by a simplified mathematical model as,

\[
y_1(k) = \begin{bmatrix} a_{111} & a_{112} & \ldots & a_{11m} \\ a_{211} & a_{212} & \ldots & a_{21m} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n11} & a_{n12} & \ldots & a_{n1m} \end{bmatrix} Y(k - 1) + \begin{bmatrix} b_{111} & b_{112} & \ldots & b_{11m} \\ b_{211} & b_{212} & \ldots & b_{21m} \\ \vdots & \vdots & \ddots & \vdots \\ b_{n11} & b_{n12} & \ldots & b_{n1m} \end{bmatrix} U(k - 1)
\]

\[ + \begin{bmatrix} a_{111} & a_{112} & \ldots & a_{11m} \\ a_{211} & a_{212} & \ldots & a_{21m} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n11} & a_{n12} & \ldots & a_{n1m} \end{bmatrix} Y(k - 2)
\]

\[ + \begin{bmatrix} b_{111} & b_{112} & \ldots & b_{11m} \\ b_{211} & b_{212} & \ldots & b_{21m} \\ \vdots & \vdots & \ddots & \vdots \\ b_{n11} & b_{n12} & \ldots & b_{n1m} \end{bmatrix} U(k - 2)
\]

\[ + \ldots + \begin{bmatrix} b_{111} & b_{112} & \ldots & b_{11m} \\ b_{211} & b_{212} & \ldots & b_{21m} \\ \vdots & \vdots & \ddots & \vdots \\ b_{n11} & b_{n12} & \ldots & b_{n1m} \end{bmatrix} U(k - n) \]

where,

\[
\theta_1 = \begin{bmatrix} \theta_{y1} \\ \theta_{y2} \\ \vdots \\ \theta_{yn} \\ \theta_{u1} \\ \theta_{u2} \\ \vdots \\ \theta_{un} \end{bmatrix} = \begin{bmatrix} \varphi_{y1}(k) \\ \varphi_{y2}(k) \\ \vdots \\ \varphi_{yn}(k) \\ \varphi_{u1}(k) \\ \varphi_{u2}(k) \\ \vdots \\ \varphi_{un}(k) \end{bmatrix} = \begin{bmatrix} Y(k - 1) \\ Y(k - 2) \\ \vdots \\ Y(k - n) \\ U(k - 1) \\ U(k - 2) \\ \vdots \\ U(k - n) \end{bmatrix} \in \mathbb{R}^{mn}
\]
Extending (38) to express multiple outputs,

\[ y_1(k) = \theta_1 \varphi^T(k) \]
\[ y_2(k) = \theta_2 \varphi^T(k) \]
\[ \vdots \]
\[ y_m(k) = \theta_m \varphi^T(k) \]

In (39), \( y_i(k) = i^{th} \) element of output vector \( Y(k) \) and \( \theta_i^T = i^{th} \) row vector of augmented matrix \([A_1, A_2, \ldots A_n, B_1, B_2, \ldots B_n]\) which represents set of parameters of the model structure to be determined. The variables within vector \( \varphi(k) \) are called regression variables or the regressors and the model in (39) is called regression model. Regression variables are known functions that depend on known and/or observed variables.

Pairs of observed variables and regressor \( \{ \{Y(k), \varphi(k)\}, k = 1, 2, \ldots, N \} \) are obtained from the running experiment for a sequence length of \( N \). The problem is to find out the parameters in such a way that the output calculated from the model in (39) agrees as closely as possible with the measured process output \( Y(k) \) in the sense of least square. This further implies that estimates of parameters will be used to minimize least square error between model and process output. In order to simplify the mathematical illustration, only first element of output vector \( Y(k) \) i.e., \( y_1(k) \) and related parameter vector \( \theta_1 \) is used in the mathematical analysis and the results are extended to describe system identification analysis for the complete process. Considering this fact, the parameters \( \theta_1 \) will be used to minimize following least square loss function for \( y_1(k) \)
By introducing notations,

\[ Y_N = (Y(1) \ Y(2) \ldots Y(N))^T \]

\[ E_N = (\varepsilon_1(1) \ \varepsilon_1(2) \ldots \varepsilon_1(N))^T \]

\[
\phi(N) = \begin{pmatrix}
\varphi^T(1) \\
\varphi^T(2) \\
\vdots \\
\varphi^T(N)
\end{pmatrix}
\]

\[ P(N) = (\phi^T(N)\phi(N))^{-1} = \left(\sum_{k=1}^{N} (\varphi(k) \ \varphi^T(k))\right)^{-1} \]

In (39), residual \( \varepsilon_1(k) \) is defined by \( \varepsilon_1(k) = y_1(k) - \hat{y}_1(k) = y_1(k) - \varphi^T(k)\theta_1 \). With these notations, loss function in (40) can be written as,

\[ V_1(\theta_1, k) = \frac{1}{2} \sum_{k=1}^{N} \varepsilon_1^2(k) \]

and \( E_N \) can be written as,

\[ E_N = Y_N - \hat{Y}_N = Y_N - \phi\theta^1 \]

Solution to this least square problem is given in [31] as,

\[ \theta_1 = \hat{\theta}_1 = (\phi^T\phi)^{-1}\phi^TY_N \]

Left side of (44) is vector of the model parameters associated with \( y_1(k) \) and the right side of the equation contains the measured data of the process i.e., input and
output over an observation window of 'N' samples. Using parameters estimates in (44),
estimated output \( \hat{y}_1(k) \) can be expressed as,
\[
\hat{y}_1(k) = \hat{\theta}_1^T \phi(k)
\] (45)

Recursive Least Square Estimation (R L S)

Identification of a system can be carried out through offline or online
computations depending on the type of application. Least square solution in (44) implies
that a significant size of input-output measured data set is always a part of system
identification calculations which gets reflected in terms of significant computation time.
In recursive LSE (RLS), computations are arranged in such a way that results (parameter
estimates) obtained at time instance 'k-1' can be used to get parameters estimates at time
instance 'k'. Therefore, recursive parameter estimation is preferred in the case of online
methods to reduce computation time and to make control algorithm more efficient. Using
recursive formulation in [17], results in (44) can be redefined as,
\[
\hat{\theta}_1(k) = \hat{\theta}_1(k - 1) + K(k) \varepsilon_1(k)
\] (46)

where,
\[
\varepsilon_1(k) = y_1(k) - \hat{\theta}_1(k - 1) \phi(k)
\]
\[
K(k) = \frac{P(k-1)\phi(k)}{1 + \phi^T(k)P(k-1)\phi(k)}
\]
\[
P(k) = [I_{(2m \times 2m)} - K(k) \phi^T(k)]P(k-1)
\]

It can be seen in RLS algorithm (46) that the current estimate \( \hat{\theta}_1(k) \) is
obtained by adding a correction to the previous estimate \( \hat{\theta}_1(k - 1) \) and correction is
proportional to estimation error \( y_1(k) - \phi^T\hat{\theta}_1(k - 1) \)) where, the term \( [\phi^T\hat{\theta}_1(k - 1)] \)
is value of \( y_1 \) at time instance 'k' predicted by identification model in (45).
For multiple process outputs scenario, one can define process outputs as a function of identified parameters by extending (43) as,

\[
\begin{align*}
\hat{y}_1(k) &= \hat{\theta}_1^T \phi(k) \\
\hat{y}_2(k) &= \hat{\theta}_2^T \phi(k) \\
&\vdots \\
\hat{y}_m(k) &= \hat{\theta}_m^T \phi(k)
\end{align*}
\]

In (47), \( \hat{y}_i(k) \) is the \( i \)-th element of estimated process output vector \( \hat{\Psi}(k) \) given by

\[
\hat{\Psi}(k) = \begin{pmatrix} \hat{y}_1(k) \\ \hat{y}_2(k) \\ \vdots \\ \hat{y}_m(k) \end{pmatrix} \in \mathbb{R}^m
\]

and using (46), parameter estimates can be expressed as,

\[
\begin{align*}
\hat{\theta}_1(k) &= \hat{\theta}_1(k-1) + K(k)\varepsilon_1(k) \\
\hat{\theta}_2(k) &= \hat{\theta}_2(k-1) + K(k)\varepsilon_2(k) \\
&\vdots \\
\hat{\theta}_m(k) &= \hat{\theta}_m(k-1) + K(k)\varepsilon_m(k)
\end{align*}
\]

From (46), (47) and (48) a recursive least square estimation process for MIMO system is expressed in (49).

\[
\theta_i(k) = \theta_i(k-1) + K(k)\varepsilon_i(k), \quad i = 1,2, \ldots, m
\]

where,

\[
\varepsilon_i(k) = y_i(k) - \theta_i^T(k-1) \phi(k), \quad i = 1,2, \ldots, m
\]

\[
K(k) = \frac{P(k-1)\phi(k)}{1 + \phi^T(k)P(k-1)\phi(k)}
\]

\[
P(k) = [I_{2mn \times 2mn} - K(k)\phi^T(k)]P(k-1)
\]
The parameter vector $\theta_i^T$ is estimates of $i^{th}$ row vector elements of augmented matrix-$\theta^T = [A_1, A_2, \ldots, A_n, B_1, B_2, \ldots, B_n]$ Elements of this augmented matrix represent process model parameters as expressed in (46) and are estimated at every sample interval using identification procedure in (49).

**Time Varying Parameters**

The RLS algorithm expressed in (49) yields estimation of system model parameters, $\theta^T = [A_1, A_2, \ldots, A_n, B_1, B_2, \ldots, B_n] \in \mathbb{R}^{m \times 2nm}$

These estimated parameters capture varying dynamics of the system at each update instant and present an equivalent mathematical model for corresponding time instant. Identified parameters keeps varying in more or less manner depending upon process nonlinearity, variation in process dynamics and update rate of system identification. It is always desirable to capture any change in process dynamics at the earliest in order to update control law to minimize undesired errors. This requirement dictates that there should be an efficient mechanism within system identification algorithm to get faster convergence of estimated parameters to deal with variation in system dynamics.

The presented recursive least square estimation technique in (49) can be further extended for two such cases. In first case, it is assumed that parameters are changing abruptly but infrequently. In the case of abrupt changes in process parameters, a working solution is obtained by resetting value of matrix $P$. Matrix $P$ is periodically reset to value $\cdot I$ where $\cdot$ is a large number. This solution implies that, by resetting value of $P$ matrix to $\cdot I$ the gain $K(k)$ in the estimator becomes large which results in large estimation steps and hence quicker estimation. The discussion on system identification [17] recommends a sophisticated solution of running $'r'$ number of estimators in parallel.
which gets reset periodically and then best estimates are selected based on some decision logic.

The second case is considered of a process in which there is continuous but slow variation in process parameters. One approach to deal with this case is to modify our general form of least square criterion expressed in (40) to

\[ V_i(\theta_i, k) = \frac{1}{2} \sum_{k=1}^{N} \lambda^{N-k}(y_i(k) - \varphi^T(k)\theta_i)^2, \quad i = 1,2,...,m \]  

(50)

Where, \( \lambda \) is called as forgetting factor or discounting factor and lies in the range of \( 0 < \lambda < 1 \). The modified loss function in (50) implies that time varying weighting of observation data is introduced. Unit weight is given for most recent data sample and data sample \( k \) time unit older is weighted by factor of \( \lambda^k \). The weighting method used here is called exponential forgetting or exponential discounting. After introducing concept of exponential forgetting to system identification procedure in (49), the recursive least square estimation leads to following expression:

\[
\theta_i(k) = \theta_i(k-1) + K(k)e_i(k) \\
e_i(k) = y_i(k) - \theta_i^T(k-1)\phi(k) \quad \text{and} \quad i = 1,2,...,m \\
K(k) = \frac{P(k-1)\phi(k)}{\lambda + \phi^T(k)P(k-1)\phi(k)} \\
P(k) = \frac{[I_{2nm} - K(k)\phi^T(k)]P(k-1)}{\lambda} 
\]  

(51)

For \( \lambda = 1 \), equal weighting occurs on all observations and value range of \( 0 < \lambda < 1 \) can be used to facilitate the tracking of varying process parameters.
Variable Forgetting Factor

The key property of an adaptive control scheme such as self-tuning regulator is its ability to track variations in process dynamics through estimated parameters. In order to achieve this critical task, employed parameter estimation technique must build up estimates of parameters based on emphasis on recent data samples. Introduction of forgetting factor in RLS estimation technique as discussed in earlier section is one of the ways to accomplish such a critical requirement of estimator. Considering RLS estimation in (51), with constant forgetting factor \( \lambda \), it can be noticed that data is discounted even if \( P(k)\phi(k) = 0 \). This condition implies that estimate of system output \( Y(k) \) does not contain any new information about the estimated process parameters. In such case, matrix \( P(k) \) increases exponentially since \( 0 < \lambda < 1 \).

Based on suggestion in [17], value for forgetting factor \( \lambda \) has to be chosen on the basis of sampling time period \( h \) and time constant for exponential forgetting \( T_f \) and their relation is given by,

\[
\lambda = e^{-(h/T_f)} 
\]  

(52)

Considering different values of ratio \( (T_f/h) \), value of corresponding \( \lambda \) is given in Table 4. Assuming constant sampling time \( h \), Table 4 indicates that, for longer exponential forgetting time constant (slower variation in process dynamics), \( \lambda \) should move towards unity and shall reduce for short value of exponential forgetting time constant (faster variation in process dynamics).

Recursive Least Square (RLS) estimation algorithm is well known for its good convergence property and small mean square error in stationary environments [24]. When applied to process with time varying dynamics, its performance in terms of tracking,
TABLE 4
Relative Forgetting Factor ($\lambda$) Values

<table>
<thead>
<tr>
<th>$T_f/h$</th>
<th>$\lambda$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.37</td>
</tr>
<tr>
<td>2</td>
<td>0.61</td>
</tr>
<tr>
<td>5</td>
<td>0.82</td>
</tr>
<tr>
<td>10</td>
<td>0.90</td>
</tr>
<tr>
<td>20</td>
<td>0.95</td>
</tr>
<tr>
<td>50</td>
<td>0.98</td>
</tr>
<tr>
<td>100</td>
<td>0.99</td>
</tr>
</tbody>
</table>

Convergence rate, stability and maladjustment depend on the forgetting factor [25]. The classical RLS algorithm discussed in earlier section uses a constant forgetting factor. By use of constant forgetting factor in RLS, it has to compromise on performance criteria mentioned earlier. When the forgetting factor is very close to one, the algorithm achieves good stability and low misadjustment but as a matter of fact its tracking capability is reduced. When forgetting factor approaches a smaller number towards zero, it improves tracking capability but at the same time it increases the misadjustment which could affect the stability of the algorithm.

Considering these limitations about constant forgetting factor, concept of Variable Forgetting Factor (VFF) is introduced in the MIMO system identification technique. Various VFF algorithms have been presented in the literature. The applicability and performance of these suggested methods for system identification depend on several factors like the

- ability to detect the changes in the system.
- character and level of noise that corrupts the output of the unknown system.
- related stability and complexity issues.
In order to make system identification more robust, gradient based variable forgetting factor technique presented in [24] is chosen. Chosen VFF algorithm controls the forgetting factor based on the dynamic equation of the gradient of the mean square error and provides fast tracking and small mean square model error. Additionally, its performance will not be degraded much even in the case of low signal-to-noise ratios. Complete mathematical analysis for chosen VFF algorithm is available in [24]. Control mechanism for chosen VFF algorithm is to adjust forgetting factor \( \lambda \) to minimize the mean square error (MSE). MSE is estimated by short exponential window which depends on factor \( \alpha \) and is expressed as,

\[ \sigma_e^2(n) = \alpha \sigma_e^2(n-1) + (1-\alpha)e^2(n) \]  

where \( e(n) \) is estimated model error and is expressed as \( e(n) = [y(n) - \hat{y}(n)] \). It is difficult to get exact value of variance of noise \( \sigma_n^2(n) \) thus, its value is calculated recursively from MSE \( \sigma_e^2(n) \) using a relatively long exponential window (depends on factor \( \beta \)) and is given by,

\[ \sigma_n^2(n) = \beta \sigma_n^2(n-1) + (1-\beta)\sigma_e^2(n) \]  

where \( \beta > \alpha \). Instead of computing noisy instantaneous estimate of gradient of forgetting factor \( \lambda \), it is calculated recursively using dynamic equation of MSE (51) and given by following equation,

\[ \nabla_\lambda(n+1) = \nabla_\lambda(n) + \frac{\partial A}{\partial \lambda} \sigma_e^2(n) + \frac{\partial B}{\partial \lambda} \sigma_n^2(n) \]  

where \( \frac{\partial A}{\partial \lambda} \) and \( \frac{\partial B}{\partial \lambda} \) are given by,

\[ \frac{\partial A}{\partial \lambda} = 2 + \frac{2(N+2)(1-\lambda)(N\lambda^2 - (N+3)\lambda - 1)}{[N(1-\lambda) + 2]^2} \]
Gradient based VFF is updated recursively using steepest descent method as follows,

\[ \frac{\partial B}{\partial \lambda} = -2 - \frac{4(1 - \lambda)[N\lambda^2 - (N + 3)\lambda - 1]}{[N(1 - \lambda) + 2]^2} \]  \hspace{1cm} (57)

Gradient based VFF is updated recursively using steepest descent method as follows,

\[ \lambda(n + 1) = [\lambda(n) - \mu \nabla_{\lambda}(n)]^{\lambda_{\text{max}}}_{\lambda_{\text{min}}} \]  \hspace{1cm} (58)

where \( \mu \) is step size, \( \lambda_{\text{max}} \) and \( \lambda_{\text{min}} \) are limits on VFF value and choice available to designer and need to be chosen based on performance requirements. Equation (52) can be used as a guideline to decide factors \( \lambda_{\text{max}} \) and \( \lambda_{\text{min}} \) based on \( (T_r/h) \) values.

VFF control mechanism works as follows. As derivative \( (\partial A/\partial \lambda) \) can be shown positive, whenever there is large model error \( e(n) \), \( \nabla_{\lambda}(n) \) increases which in turn reduces forgetting factor \( \lambda \) which improves tracking ability of system identification. On the other hand whenever \( \sigma_\epsilon^2(n) \) is reduced to a certain level, \( (\partial A/\partial \lambda) < (- \partial B/\partial \lambda) \), it drives \( \nabla_{\lambda}(n) \) to negative which leads \( \lambda \) to increase towards ceiling \( \lambda_{\text{max}} \). For design implementation purpose, following values are considered for various design factors based on suggestion in [24].

\[ \lambda_{\text{max}} = 0.999, \lambda_{\text{min}} = 0.7, \quad \alpha = 0.7, \beta = 0.95, \quad \mu = 0.2, N = 5, A = 0.7 \]

**Formulation of State Space Model Structure**

Structure of dynamic model used for identification process is a choice made by the designer and is mainly motivated by its compatibility with chosen control design technique. Present work is emphasized on keeping most control structures as general as possible to provide flexibility to the users to adapt to their choice of control design techniques. Considering this fact, state space model has been chosen as a suitable candidate to define dynamics of the system through system identification process.
This single structure has a capability to describe all linear systems and has been found to be the most suitable structure in all modern control techniques. State space structure exploits all advancements in the field of matrix linear algebra to enrich applied control and making control design methods more sophisticated and at the same time practically implementable. Additionally, in the proposed thesis, design method for a self tuning regulator for MIMO system is investigated and optimal control technique has been chosen for control design. Optimal control design method absorbs state space model of the system without any need of structure customization or reformulation.

The MIMO system model represented by (36) can be put in block observable canonical form as discussed in [26] by defining a set of state vectors as follows:

\[
\begin{align*}
X_1(k) &= A_n Y(k - 1) + B_n U(k - 1) \\
X_2(k) &= X_1(k - 1) + A_{n-1} Y(k - 1) + B_{n-1} U(k - 1) \\
&
\vdots \\
X_{n-1}(k) &= X_{n-2}(k - 1) + A_2 Y(k - 1) + B_2 U(k - 1) \\
X_n(k) &= X_{n-1}(k - 1) + A_1 Y(k - 1) + B_1 U(k - 1)
\end{align*}
\]

(59)

Where, \(X_1, X_2, \ldots, X_n \in \mathbb{R}^m\). The above set can be written in more compact form by using state space notion,

\[
\begin{align*}
X(k) &= A_M X(k - 1) + B_M U(k - 1) \\
Y(k) &= C_M X(k)
\end{align*}
\]

(60)
Where, 

**state transition matrix,** $A_M = 
\begin{bmatrix}
0 & 0 & 0 & \ldots & A_n \\
I_m & 0 & 0 & \ldots & A_{n-1} \\
0 & I_m & 0 & \ldots & \vdots \\
\vdots & \vdots & \ddots & \ddots & \vdots \\
0 & 0 & I_m & \ldots & A_2 \\
0 & 0 & 0 & \ldots & I_m & A_1 \\
\end{bmatrix} \in \mathbb{R}^{mn \times mn}$

**input transition matrix,** $B_M = 
\begin{bmatrix}
B_n \\
B_{n-1} \\
\vdots \\
B_2 \\
B_1 \\
\end{bmatrix} \in \mathbb{R}^{mn \times m}$

$C_M = [0, 0, \ldots, I_m] \in \mathbb{R}^{m \times mn}$

**state vector:** $X(k) = 
\begin{bmatrix}
X_1(k) \\
X_2(k) \\
\vdots \\
X_n(k) \\
\end{bmatrix} \in \mathbb{R}^{mn}$ as $X_i(k) \in \mathbb{R}^m$ $i = 1, 2, \ldots, n$

**output vector:** $Y(k) = 
\begin{bmatrix}
y_1(k) \\
y_2(k) \\
\vdots \\
y_m(k) \\
\end{bmatrix} \in \mathbb{R}^m$

**input vector:** $U(k) = 
\begin{bmatrix}
u_1(k) \\
u_2(k) \\
\vdots \\
u_m(k) \\
\end{bmatrix} \in \mathbb{R}^m$

State space model in (60) is used in the system identification procedure and RLS technique is used for estimation of parameters. System identification procedure yields system model parameter matrices $A_M$ and $B_M$ and these matrices are updated at each update instance or sample interval.
Assessment of MIMO System Identification Algorithm

The MIMO system identification algorithm in (51) along with state space formulation in (60) was applied on a linear MIMO process model. The linearized aircraft models presented in Chapter II as Case-1 to Case-4 were used for performance assessment of the developed MIMO system identification algorithm. By using Case-1 aircraft linearized model, a MATLAB™ simulation program (mimoSystemIdentification.m with user selection: 1. MIMO System Identification with constant forgetting factor) was developed to assess performance of MIMO system identification algorithm with constant forgetting factor $\lambda = 0.9$.

The process under identification has 3 inputs and 3 outputs ($m=3$) and order of system identification was chosen to be 3 ($n=3$). The process input $U(k)$, process model output $Y(k)$ and state vector $X(k)$ are expressed as,

$$Y_M(k) = \begin{pmatrix} y_1(k) \\
                      y_2(k) \\
                      y_3(k) \end{pmatrix} = \begin{pmatrix} \text{Roll Angle} \\
                      \text{Pitch Angle} \\
                      \text{Yaw Angle} \end{pmatrix} \in \mathbb{R}^3$$

$$= \begin{pmatrix} x_7(k) \\
                     x_8(k) \\
                     x_9(k) \end{pmatrix} \quad \therefore C_M = \begin{bmatrix} 0_{m \times (n-1)} & I_{m \times m} \end{bmatrix}$$

$$U(k) = \begin{pmatrix} u_1(k) \\
                      u_2(k) \\
                      u_3(k) \end{pmatrix} = \begin{pmatrix} \text{Elevator} \\
                      \text{Aileron} \\
                      \text{Rudder} \end{pmatrix} \in \mathbb{R}^3$$

$$X(k) = \begin{pmatrix} x_1(k) \\
                     x_2(k) \\
                     \vdots \\
                     x_9(k) \end{pmatrix} \in \mathbb{R}^9 \quad \ldots m = 3, n = 3 \Rightarrow mn = 9$$
MIMO System Identification

The system identification framework is illustrated in Figure 15. The MIMO system identification block continuously updates parameters of identified model i.e. matrix $A_M$ and $B_M$ values. The process output is denoted by $Y(k)$ and identified process output is shown as $Y_M(k)$. The error in system identification - $e(k)$ is expressed as difference between process output and process model output. Random varying process input vector - $U(k)$ as shown in Figure 16 was used for system identification. The corresponding identification results are shown in Figure 17, Figure 18, and Figure 19.

Variation in different parameters (values in matrix $A_M$) during process of identification is shown in Figure 20. As the process under identification is linear with
constant parameters, after initial variation the parameters are converged to a constant number and remain constant throughout the simulation indicating their convergence to a specific value.

**System Identification with Variation in Process Dynamics**

In order to test effectiveness of the presented MIMO system identification algorithm, large variation in process dynamics were simulated by changing process itself after specific time interval. For this analysis, linearized models of aircraft in Case-1, Case-2 and Case-3 were used to represent process under identification. Figure 21 shows...
the setup for this analysis. After every one second, process was replaced with different linearized model of an aircraft with the sequence of Case-1, 2 and 3.

A MATLAB™ simulation program (mimoSystemIdentification.m with user selection: 2. MIMO System Identification with variation in Process dynamics.) was developed to assess performance of the system identification algorithm under process dynamics variation scenario. Roll angle output i.e., $y_1(k)$ is chosen as a monitoring variable to measure performance merits for this analysis. It was assumed that remaining two process outputs namely Pitch and Yaw angle will also have similar performance that of Roll angle output. Result of this analysis is shown in Figure 22 which indicates effectiveness of the system identification technique. It is worth noting that this is a difficult test for the identification algorithm since in this simulation; process is changed.
abruptly within a sample interval. In reality, this is more or less unpractical and process dynamics varies at much slower rate.

The result from this analysis indicates effectiveness of the identification procedures as an identified process model is continuously updated with new parameters to capture variation in process dynamics from that of the earlier. In conclusion, results shown in Figure 22 are promising for the control applications for the processes that involve large variation in dynamics.

The performance of the system identification with constant forgetting factor in presence of process dynamics variation was evaluated using two different values of $\lambda$ (0.8 and 0.9) using a MATLAB simulation program (mimoSystemIdentification.m with user selection: 3. Performance Analysis with constant forgetting factor.). For this
analysis, once again process switching framework was used and the process was changed after every one second with sequence Case-1, 2 and 3. Results of identification are shown in Figure 23. Errors in identification during transient and steady state are also shown in Figure 24.

From results shown, it is very clear that lower values of \( \lambda \) yields quicker convergence of an identified model parameters but larger transient errors and vice versa. Figure 24 also shows expanded view to have a close look at steady state errors from time 1.5 to 2 Sec during which system identification parameters has been converged. These results clearly indicates that higher values of \( \lambda \) provides low transient errors but high steady state errors and vice versa.
Fig. 20. Convergence of parameter estimates.

Merits of Variable Forgetting Factor (VFF)

The gradient based VFF algorithm is introduced in system identification procedure. Performance of VFF algorithm was evaluated using a MATLAB simulation program (mimoSystemIdentification.m with user selection: 4. MIMO System Identification with Variable Forgetting Factor -VFF). The process was changed after every one second to introduce large variation in process dynamics. The improvements in the identification results because of VFF are shown in Figure 25. These results can be directly compared with results in Figure 22 (with constant $\lambda$). Figure 26 shows variation in forgetting factor due to VFF implementation. As the process is changed at each second, $\lambda$ takes lower value to get quicker parameter convergence.
Additionally, the performance of identification with V FF is compared with that of the constant values of 0.7 and 0.99 in terms of identification errors. This comparison is shown in Figure 27 from the transient and steady state errors perspective. From the analysis presented, it is evident that system identification with V FF works better than that with constant \( \lambda \) irrespective of the rate of change in process dynamics. This feature will yield performance improvement in system identification results.

Adaptive and Optimal Control Implementation

The system identification procedure discussed in the previous section yields a linear mathematical model that represents finite dynamics of the process under control. Further, this process model is used to carry out an online adaptive controller design using optimal control theory. The design of Regulator and Servo control configuration for the STR is discussed in this section.
Fig. 22. Results of identification with process dynamics variation.

**Regulator Problem**

As a part of the STR, an online control design procedure is carried out on the results obtained by the system identification. As discussed in the earlier section, MIMO system identification yields process model in the form given by (60). This process model is then used to design an adaptive controller for process.

\[
X(k + 1) = A_M X(k) + B_M U(k)
\]

\[
Y(k) = C_M X(k)
\]  \hspace{1cm} (61)

In the following discussion, LQR based optimal control technique is used for controller design procedure. For optimal control problem we desire to determine a control law such that a quadratic performance index \( J \) expressed in (62) will be minimized.
In (62), matrix $Q_1$ is positive definite or positive semi definite Hermitian matrix and $Q_2$ is a positive definite Hermitian matrix. In the summation, the first term accounts for relative errors during control process and the second term accounts for expenditure of energy of the control signals. $Q_1$ and $Q_2$ are weighting matrices which correlate measure of importance between process control errors and control energy. If optimal regulator system is stable, value of $J$ converges to a constant. This further indicates that $X(\cdot) = 0$. Thus, with this formulation, control problem dictates in finding control $U(k)$ such that

$$J = \frac{1}{2} \sum_{k=0}^{\infty} \left[ X^T(k) Q_1 X(k) + U^T(k) Q_2 U(k) \right]$$

(62)
As per mathematical analysis presented in [20], to design a steady state optimal controller for the process defined by (61), we require steady state solution of the Reccati equation given by,

$$ P(k+1) = Q_1 + A_M^T P(k) A_M $$

$$ - A_M^T P(k) B_M [Q_2 + B_M^T P(k) B_M]^{-1} B_M^T P(k) A_M $$

(63)

In order to compute the numerical solution for (63), it is important to note that matrix \( P \) is either Hermitian or a real symmetric matrix and is positive definite. In general practice,
solution for (63) begins with $P(0) = 0$ and iterating the equation until a steady value is obtained. Optimal control law under such formulation is given by,

$$U(k) = -K X(k)$$

(64)

where $K$ is state feedback gain matrix. Thus, design of optimal controller boils down to finding appropriate value of matrix $K$. Using [20], state feedback matrix $K$ can be expressed in terms of $P, A_M, B_M,$ and $Q_2$ as,

$$K = (Q_2 + B_M^T P B_M)^{-1} B_M^T P A_M$$

(65)

Figure 25 shows the structure of optimal regulator in current framework. It is assumed that for the controller design work, matrices $A_M$ and $B_M$ are available through the system identification procedures.
The matrices $Q_1$ and $Q_2$ are selected to weigh the relative importance of the performance measures caused by the state vector $X(k)$ and control vector $U(k)$ respectively. In general, certain amount of trial and error procedure is used through a computer program before a satisfactory design results. A control designer can select $Q_1$ to provide relative cost for system states based on critical outputs and selection of $Q_2$ can be made to reflect relative emphasis on control energy consumption.

**Servo Control**

In earlier section, the closed loop control of the system is discussed in regulator configuration in which control loop minimizes variation in process outputs because of disturbance or variations in process dynamics. This control was purely based on state feedback. In this section, introduction of reference command and servo control based on state feedback and integral control action will be discussed. In order to eliminate steady state errors in closed loop configuration, it is generally required to add

---

**Fig. 26. Value of $\lambda$ during identification with VFF.**
one or more integrators in control loop [20]. The control loop shown in Figure 29 demonstrates one method to introduce an integrator in closed loop system.

A new state vector that integrates the difference between the command vector ‘R’ and the output vector ‘Y’ is introduced in mathematical model of closed loop system. For each command input, an integrating element is added resulting in an integral controller with ‘m’ integrating elements. (Process inputs/outputs = m). Consider a linear MIMO process model as a result of system identification process given in (61) as,

\[ X(k + 1) = A_X X(k) + B_Y U(k) \]
\[ Y(k) = C_X X(k) \]

(66)
where, $X(k) \in \mathbb{R}^{mn}$ - process state vector,

$Y(k) \in \mathbb{R}^{m}$ - output vector

$U(k) \in \mathbb{R}^{m}$ - control input vector.

$A_M \in \mathbb{R}^{mn \times mn}$, $B_M \in \mathbb{R}^{mn \times m}$ and $C_M \in \mathbb{R}^{m \times mn}$ are system matrices.
As shown in Figure 29, after introduction of integral controller in closed loop, we have additional states given by,

\[ V(k) = V(k - 1) + (R(k) - Y(k)) \]  

(67)

\( V(k) \in \mathbb{R}^m \) is actuating error value expressed as integrated error between reference command \( R(k) \) and process output \( U(k) \). From Figure 29, control vector \( U(k) \) and other variables can be expressed as,

\[ U(k) = -K_x X(k) + K_i V(k) \]  

(68)

\[ V(k + 1) = V(k) + R(k + 1) - Y(k + 1) \]

\[ = V(k) + R(k + 1) - C[A_M X(k) + B_M u(k)] \]

\[ = -C_M A_M X(k) + V(k) - C_M B_M U(k) + R(k + 1) \]  

(69)

From (64) and (67) we have,

\[
\begin{bmatrix}
X(k + 1) \\
V(k + 1)
\end{bmatrix} =
\begin{bmatrix}
A_M & 0_{nm \times m} \\
-C_M A_M & I_{m \times m}
\end{bmatrix}
\begin{bmatrix}
X(k) \\
V(k)
\end{bmatrix} +
\begin{bmatrix}
B_M \\
-C_M B_M
\end{bmatrix} U(k) +
\begin{bmatrix}
0_{nm \times m} \\
I_{m \times m}
\end{bmatrix} R(k + 1) \tag{70}
\]

Considering the input \( R(k) \) as a step function or \( R(k) = R(k + 1) = R \), as \( k \) approaches infinity,

\[
\begin{bmatrix}
X(\infty) \\
V(\infty)
\end{bmatrix} =
\begin{bmatrix}
A_M & 0 \\
-C_M A_M & 1
\end{bmatrix}
\begin{bmatrix}
X(\infty) \\
V(\infty)
\end{bmatrix} +
\begin{bmatrix}
B_M \\
-C_M B_M
\end{bmatrix} U(\infty) +
\begin{bmatrix}
0 \\
1
\end{bmatrix} R(\infty) \tag{71}
\]

Define-

\[ X_e(k) = X(k) - X(\infty) \]

\[ V_e(k) = V(k) - V(\infty) \]  

(72)

then, the error becomes

\[
\begin{bmatrix}
X_e(k + 1) \\
V_e(k + 1)
\end{bmatrix} =
\begin{bmatrix}
A_M & 0 \\
-C_M A_M & 1
\end{bmatrix}
\begin{bmatrix}
X_e(k) \\
V_e(k)
\end{bmatrix} +
\begin{bmatrix}
B_M \\
-C_M B_M
\end{bmatrix} U_e(k) \tag{73}
\]
Note that,

\[ U_e(k) = -K_S X_e(k) + K_I V_e(k) \]

\[ U_e(k) = -[K_S \quad -K_I] \begin{bmatrix} X_e(k) \\ V_e(k) \end{bmatrix} \]  \hspace{1cm} (74)

Now define,

\[ G = \begin{bmatrix} A_M & 0 \\ -C_M A_M & 1 \end{bmatrix}, \quad H = \begin{bmatrix} B_M \\ -C_M B_M \end{bmatrix}, \quad K = -[K_S \quad -K_I], \quad w(k) = U_e(k) \]

\[ \xi(k) = \begin{bmatrix} X_e(k) \\ V_e(k) \end{bmatrix} = \begin{bmatrix} x_{1e}(k) \\ x_{2e}(k) \\ \vdots \\ x_{(mn)e}(k) \\ x_{(mn+1)e}(k) \\ x_{(mn+2)e}(k) \\ \vdots \\ x_{(mn+m)e}(k) \end{bmatrix} \]

\[ V_e(k) = \begin{bmatrix} x_{(mn+1)e}(k) \\ x_{(mn+2)e}(k) \\ \vdots \\ x_{(mn+m)e}(k) \end{bmatrix} \]

Thus, we have

\[ \xi(k + 1) = G \xi(k) + H \, w(k) \]

\[ w(k) = -K \, \xi(k) \]  \hspace{1cm} (75)

This formulation yields a state space model for closed loop servo control and thus, can be attacked by using the LQR optimal control design method discussed earlier by choosing appropriate weighting matrices \( Q_1 \) and \( Q_2 \). LQR control applied on
The formulated mathematical model yields optimal gain vector $K$ which is combination of $K_S$ and $K_I$.

**Assessment of Controller in Regulator and Servo Control Configuration**

The Self Tuning Regulator (STR) based on optimal control theory and MIMO system identification technique is designed for a process under control. The architecture of implementation is shown in Figure 30. Once identified process model is updated, it is passed to an online control design block to compute controller parameters.

![Diagram](image)

**Fig. 30.** Self tuning regulator implementation.

The process block in Figure 30 refers to plant under control. The MIMO identification block updates process model and passes it to the LQR controller design.
block and the state computation block. The optimal regulator theory discussed in earlier section is applied on process model to get state feedback gain of the controller block. The feedback gain $K$ and computed states of the process model $X(k)$ are then used to compute control vector using control law $U(k) = -K X(k)$. Complete control scheme is updated at each time sample which is of 10mSec period in current analysis and simulation. Using the aircraft linearized model in Case-1, a MATLAB™ simulation program (mimoRegulator.m with user selection: 1.MIMO Regulator Simulation.) was developed to assess performance of designed STR. Based on (63) and (65), computations of state feedback gain matrix $K$ is performed by MATLAB™ function: $K = \text{dlqr}(A_M,B_M,Q_1,Q_2)$. The weighting matrix $Q_1$ and $Q_2$ were selected so as express relative emphasis on states which reflects output of process and relative emphasis on control energy respectively.

An initial state disturbance of 2 degrees was introduced in the process state that represents Roll angle of aircraft. The STR response for an initial disturbance is shown in Figure 31. It behaves according to the expectation as it zeros all states ($X(\cdot) = 0$) within short time. The capability of designed regulator to handle the process with varying dynamics was tested using process switching framework. For this assessment, after every 15 seconds, the process under control was changed in sequence of Case-1, 2 and 3. The framework for this assessment was implemented in MATLAB™ simulation program (mimoRegulator.m with user selection: 2. MIMO Self Tuning Regulator with variation in Process dynamics).

The regulator performance in presence of large variation in the process dynamics is shown in Figure 32 as the result of simulation. Further, the designed STR
will be used in the Servo configuration in which, reference command will be added in the controller design procedures so that process output will track reference commands. The servo control formulation was applied on Case-1 linearized model of aircraft. The performance of servo configuration was evaluated using different step commands as reference inputs. The error in response to step command is an indication of command tracking capabilities of the designed closed loop servo control system. A MATLAB simulation program (mimoServoControl.m) was developed to evaluate servo control performance. The results of simulation in terms of reference input and process outputs are shown in Figure 33.

In summary, this chapter has established complete design procedures for MIMO STR as an adaptive control scheme. Performance assessments are presented at each stage like system identification, regulator mode control and servo mode control.
Effectiveness of proposed design technique is demonstrated using results of these assessments. Further, this design methodology will be applied on highly nonlinear 6-DOF fighter aircraft model and its performance will be evaluated. This will be presented in Chapter IV.
Fig. 33. Servo control performance of designed STR.
CHAPTER IV

APPLICATION TO THE 6-DOF AIRCRAFT MODEL

This chapter presents an application of MIMO STR for high performance aircraft control problem. The complete simulation setup and the results are presented in the following discussion. Further, complex maneuvers are performed on aircraft model by using the proposed control scheme. The robustness and survivability of the aircraft under control is also verified through a damaged aircraft scenario. The performance of the designed MIMO STR is compared with that of the controller designed for specific operating conditions of aircraft with no parameter adaptation mechanism. The most prominent feature of proposed design technique is to accommodate any dynamic process without knowing its mathematical model. This feature is demonstrated by reconfiguring aircraft output vector.

Overall System Model

The MIMO STR design methodology is applied on highly nonlinear 6-DOF aircraft model discussed in Chapter II. The performance of the proposed technique was assessed during development phase at various stages like MIMO system identification, regulator mode control and response of process in servo control configuration to various reference commands. The simulation setup for MIMO system identification
procedure is shown in Figure 34. Initially, the nonlinear aircraft model was trimmed with steady wing level flight at an altitude of 8000 ft. and velocity of 600 ft/sec. A sequence of random varying signal was used as an identification input $U(k)$. The performance of system identification is measured in terms of error signal $e(k)$ defined by difference between process output $Y(k)$ and identification model output $Y_M(k)$.

![Fig. 34. MIMO system identification of aircraft.](image)

In order to check robustness of system identification procedure, aircraft altitude and velocity states were changed to 20000 ft and 350 ft/sec after 5 seconds. This test causes change in aircraft dynamics because of aerodynamics coefficient variation and
explicitly brings out large nonlinearity in the aircraft model. The ability of system identification procedure to handle such variations in the process dynamics is an important feature and was verified through this test.

A MATLAB™ program was developed (mimoSTRforAircraft.m with user input- 1: MIMO System Identification for F-16 aircraft) to perform system identification procedure shown in Figure 34. A 4th order general MIMO process model (n=4) was considered for system identification procedure. The parameter estimates provided by the identification procedure were used to parameterize selected model. The results of simulation are shown in Figure 35. From the results, it is evident that proposed MIMO system identification procedure yields good performance irrespective of change in critical process states.

In order to evaluate regulator mode performance, the aircraft model and the MIMO STR were arranged as per structure shown in Figure 36. As shown, for the process model identified by system identification procedure, an online optimal controller design was carried out. The control vector \( U(k) \) was derived from the computed state vector \( X(k) \) and the state feedback gain matrix (\( K \)). The regulator mode framework presented in Figure 36 was implemented in a MATLAB™ program (mimoSTRforAircraft.m with user input- 2: MIMO STR in Regulator configuration.)

This simulation was meant to evaluate properties of the designed STR to regulate outputs of the aircraft through state feedback control without any reference command. Initially, aircraft was trimmed with steady wing level flight at an altitude of 8000 ft. and velocity of 600 ft/sec. Trim state output vector is denoted by \( Y_{trim} \) and has value as - [Roll = 0 deg, Pitch= 2 deg , Yaw = 0 deg] whereas current value of aircraft
output is denoted by $Y(k)$. With the state feedback controller in place, it was expected to maintain aircraft output vector fixed to the trim output vector irrespective of any change in process states. The output deviation signal $Y_e(k) = [Y(k) - Y_{\text{trim}}]$ was used as an output signal for identification purpose. The goal of the state feedback controller is to minimize deviation signal $Y_e(k)$ making aircraft output vector to remains at trim output values. In order to test the robustness of the designed regulator, a disturbance was introduced at time $t = 10$ sec. The simulation results for this evaluation are shown in Figure 37.
As shown in the results, aircraft outputs are regulated quickly to their trim values after 3 seconds. The time required for parameter convergence for system identification procedure was also included in this time interval. The disturbed process states at time \( t = 10 \) sec. are controlled and regulated by the designed STR to maintain aircraft outputs to their trim values.

The servo control configuration of the MIMO STR has been discussed in Chapter III. The proposed servo control configuration is applied for aircraft control problem as shown in Figure 38.
The servo control framework was implemented with a MATLAB™ program (mimoSTRforAircraft.m with User Input: 3: MIMO STR in Servo Control configuration). The reference command vector consists of desired Euler angles of aircraft (Roll, Pitch and Yaw). As the nonlinear aircraft inputs-outputs exhibit strong cross-coupling, it is important to consider valid combination of Euler angles as a reference command. In order to obtain such valid combinations of aircraft angles, initially aircraft model was trimmed with steady wing level flight at an altitude of 8000 ft and velocity of 600 ft/sec and allowed to run with uncontrolled inputs. The Euler angles of the aircraft were captured at various time instances to use them as a set of valid reference commands.
in the servo mode simulation exercise. The reference commands used for the servo control mode evaluation are shown in Table 5.

**TABLE 5**

Reference Commands for Aircraft

<table>
<thead>
<tr>
<th>Reference</th>
<th>Units</th>
<th>0 &lt; t • 10</th>
<th>10 &lt; t • 20</th>
<th>20 &lt; t • 30</th>
<th>30 &lt; t • 40</th>
<th>40 &lt; t • 50</th>
</tr>
</thead>
<tbody>
<tr>
<td>Roll</td>
<td>deg</td>
<td>0</td>
<td>4.6</td>
<td>8.8</td>
<td>0.16</td>
<td>0</td>
</tr>
<tr>
<td>Pitch</td>
<td>deg</td>
<td>1.87</td>
<td>7.8</td>
<td>1.6</td>
<td>-2.4</td>
<td>1.87</td>
</tr>
<tr>
<td>Yaw</td>
<td>deg</td>
<td>0</td>
<td>3.8</td>
<td>-2.2</td>
<td>2.3</td>
<td>0</td>
</tr>
</tbody>
</table>

Figure 39 shows result of aircraft control simulation in the servo control mode. As seen in result, reference commands were applied without any command filtering to evaluate transient and steady state response in servo control mode. The command filtering is used in following subsection in which large angle complex
Fig. 39. Servo control results for aircraft control problem.

maneuvers will be performed on an aircraft model. The 3-D trajectory of the aircraft motion for given reference command is shown in Figure 40.

Aircraft Control Simulation Investigation

The servo control simulation framework for high performance aircraft control problem is presented in the earlier section. The same framework is used in this section to
Fig. 40. 3-D trajectory of aircraft motion during servo control evaluation.

assess performance of the proposed MIMO STR. Multiple simulations are performed to test the performance of the proposed adaptive control scheme through various aspects. An assessment test presented in earlier section is extended to evaluate performance at multiple operating points. The objective of these test simulations is to bring out large nonlinearities in aircraft model in order to evaluate performance of the designed controller for process with large nonlinearities and variation in process dynamics.

The aircraft under control has large operating range in terms of altitude and velocity (Altitude: 5000 - 40000 ft, Velocity: 300-900 ft/sec). As discussed in Chapter II, combination of these two process states has dominant effect on different aerodynamic coefficients present in mathematical equations of aircraft model. Four test cases were formed in Chapter II with different altitude and velocity combinations. In this section,
additional 6 test cases are formed with different altitude and velocity combinations so as to cover large operating range of aircraft through 10 test cases altogether. A MATLAB™ program (aircraftTestCases.m) was developed to trim aircraft at these 10 operating points. The trim condition data is saved in a file “TrimData.txt” for user reference after the program run. Table 6 shows values of trim conditions for each operating point.

The first evaluation was carried out by simulating the scenario of aircraft passing through multiple operating points. This test is similar to one presented in Chapter II in which, the need of an adaptive control is demonstrated for aircraft control problem. A MATLAB™ program (mimoSTRandLinearController.m) was developed to carry out this simulation exercise. Initially, the aircraft was trimmed at Case-4 conditions, outputs were zeroed and then a reference command was applied on aircraft. The output response for given reference command is shown in Figure 41. After achieving reference command, the aircraft operating point was changed after every 5 seconds to match flight conditions in Case-1 and Case-3 respectively. Figure 42 shows response of the MIMO STR under such scenario along with that of the Case-4 controller and the gain scheduling scheme presented in Chapter II. It is observed from the results that the MIMO STR provides improved response over fixed gain controller by constantly adapting controller parameters. It also shows better performance over gain scheduling controller by eliminating transients presents in process outputs due to the controller switching.

In the following test simulations, multiple reference commands were provided along with multiple operating points. This test was used to evaluate performance of the designed control scheme to track the different reference commands along with the change in operating points (change in altitude and velocity). A MATLAB™ program
<table>
<thead>
<tr>
<th>Variables</th>
<th>Case-1</th>
<th>Case-2</th>
<th>Case-3</th>
<th>Case-4</th>
<th>Case-5</th>
<th>Case-6</th>
<th>Case-7</th>
<th>Case-8</th>
<th>Case-9</th>
<th>Case-10</th>
</tr>
</thead>
<tbody>
<tr>
<td>Altitude (ft)</td>
<td>8000</td>
<td>10000</td>
<td>13000</td>
<td>35000</td>
<td>15000</td>
<td>18000</td>
<td>25000</td>
<td>30000</td>
<td>15000</td>
<td>22000</td>
</tr>
<tr>
<td>Vt (ft/sec)</td>
<td>600</td>
<td>500</td>
<td>350</td>
<td>700</td>
<td>800</td>
<td>700</td>
<td>800</td>
<td>850</td>
<td>400</td>
<td>900</td>
</tr>
<tr>
<td>• (deg)</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>• (deg)</td>
<td>1.85</td>
<td>3.66</td>
<td>9.55</td>
<td>4.63</td>
<td>0.81</td>
<td>1.84</td>
<td>2.16</td>
<td>1.87</td>
<td>7.68</td>
<td>0.81</td>
</tr>
<tr>
<td>• (deg)</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>• (deg)</td>
<td>1.85</td>
<td>3.66</td>
<td>9.55</td>
<td>4.63</td>
<td>0.81</td>
<td>1.84</td>
<td>2.16</td>
<td>1.87</td>
<td>7.68</td>
<td>0.81</td>
</tr>
<tr>
<td>• (deg)</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>P (deg/sec)</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Q (deg/sec)</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>R (deg/sec)</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Thrust (lbs)</td>
<td>2416</td>
<td>2110</td>
<td>2676</td>
<td>2070</td>
<td>3021</td>
<td>2310</td>
<td>2147</td>
<td>2158</td>
<td>2408</td>
<td>3025</td>
</tr>
<tr>
<td>Elevator (deg)</td>
<td>-1.53</td>
<td>-2.02</td>
<td>-4.06</td>
<td>-2.02</td>
<td>-1.23</td>
<td>-1.42</td>
<td>-1.41</td>
<td>-1.28</td>
<td>-3.33</td>
<td>-1.22</td>
</tr>
<tr>
<td>Aileron (deg)</td>
<td>-0.08</td>
<td>-0.08</td>
<td>0.14</td>
<td>-0.08</td>
<td>-0.09</td>
<td>-0.08</td>
<td>-0.08</td>
<td>-0.09</td>
<td>0.06</td>
<td>-0.09</td>
</tr>
<tr>
<td>Rudder (deg)</td>
<td>0</td>
<td>0.07</td>
<td>-0.05</td>
<td>0.07</td>
<td>-0.05</td>
<td>-0.01</td>
<td>-0.01</td>
<td>-0.03</td>
<td>0.02</td>
<td>-0.04</td>
</tr>
<tr>
<td>dLEF (deg)</td>
<td>2.07</td>
<td>5.14</td>
<td>2.77</td>
<td>4.5</td>
<td>0</td>
<td>1.13</td>
<td>0.966</td>
<td>0</td>
<td>11.13</td>
<td>0</td>
</tr>
</tbody>
</table>
(mimoSTRforAircraftCases.m) was developed to simulate this framework. Each simulation run was carried out for 60 seconds. The operating conditions were changed after every 30 seconds and reference command was changed after every 10 seconds. Three simulation runs were performed with sequence as Case 5 & 6, Cases 4 & 10 and Case 9 & 2. The simulation results are shown in Figure 43, Figure 44, and Figure 45 respectively.

Fig. 41. Aircraft Output Response using MIMO STR for Case-4.

From the above analysis, it is evident that the designed MIMO STR based control scheme provides desired performance irrespective of presence of the large nonlinearities and variation in process dynamics.

The high performance aircraft always deal with complex maneuvers which imply large variations in aircraft angle within short duration. These maneuvers present
large nonlinearities to control scheme used. In order to test the performance of the proposed adaptive control scheme, multiple coordinated maneuvers are studied in following analysis. The performance of the MIMO STR is compared with that of the constant gain controllers designed for particular flight conditions. A MATLAB™ simulation program (mimoSTRandFixedGainComparison.m) was developed to simulate these coordinated maneuvers.

Fig. 42. Performance for multiple operating points scenario.
Fig. 43. Case-5 and 6 Multiple Reference Commands and Operating Points.

Flight conditions in Case-2 (altitude = 10000 ft and velocity = 500 ft/sec) and Case-4 (altitude = 35000 ft and velocity = 700 ft/sec) were considered for this exercise.

For each flight condition, two aggressive maneuvers were assigned as shown in Table 7. The results of the simulation are shown in Figure 46, Figure 47, Figure 48, and Figure 49.
TABLE 7

REFERENCE COMMANDS FOR COORDINATED MANEUVERS

<table>
<thead>
<tr>
<th>Reference Command</th>
<th>Units</th>
<th>Flight-1</th>
<th>Flight-2</th>
<th>Flight-3</th>
<th>Flight-4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Roll Angle (°)</td>
<td>deg</td>
<td>35</td>
<td>46</td>
<td>37</td>
<td>57</td>
</tr>
<tr>
<td>Pitch Angle (°)</td>
<td>deg</td>
<td>26</td>
<td>34</td>
<td>28</td>
<td>0</td>
</tr>
<tr>
<td>Yaw Angle (°)</td>
<td>deg</td>
<td>44</td>
<td>57</td>
<td>46</td>
<td>34</td>
</tr>
</tbody>
</table>

It was observed from the results in this exercise and Chapter II analysis that, linear controllers designed for the particular flight conditions works well around their design point with small reference commands. However, in the case of large angle complex maneuvers, their performance gets degraded and does not guarantee stability over entire flight duration.

An additional simulation exercise was carried out to analyze the adaptation, robustness, recovery and survivability of the designed controller. In order to test these performance qualities, aggressive maneuvers are studied on a damaged aircraft for the full flight envelope. In particular, the ability of the developed identification and control method to identify and accommodate possible damage is of particular interest. In this study, the control objective is to make an aircraft track the prescribed desired command even in a case of decrease in control surface effectiveness due to control surface damage. In order to verify the performance of the proposed control law under such scenario, control surface impairment case characterized by the percentage loss of the total effectiveness of the control surface is considered.

This scenario was simulated using a MATLAB™ program (aircraftRecoveryAnalysis.m) with 25% control surface damage for all of the three surfaces - elevator, aileron and rudder from time t=0 till complete flight duration. The
Fig. 44. Case 4 and 10 multiple reference commands and operating points.

Flight conditions in the Case-7 (altitude= 25000 ft and velocity=800 ft/sec) was used as the starting point of the flight. The response of the damaged aircraft is compared with that of a good aircraft and is shown in Figure 50. It is observed from the results that, proposed control law provides robust behavior by tracking the desired reference even in the case of damaged aircraft. The 3-D trajectory of the aircraft motion during the flight time is shown in Figure 51.
One of the prominent features of the proposed design technique is that, it does not require explicit process dynamics mathematical model to design a controller. The control scheme makes use of only process input-output observations and builds an equivalent representation of process dynamics in the form of a general model structure (state space model in this case). This feature is demonstrated by reconfiguring aircraft
Fig. 46. Flight -1 maneuver with case-2 flight conditions.

simulation model to provide the different output vector. For the analysis presented in the earlier sections, the aircraft Euler angles were considered as the process output and was expressed as $Y = [\text{Roll, Pitch, Yaw}]$. For this analysis, aircraft simulation model is reconfigured to provide output in terms of angle of attack ($\alpha$), roll angle ($\phi$) and sideslip angle ($\beta$) and output vector is expressed as $Y = [\alpha, \phi, \beta]$. The MIMO STR based control
scheme was applied on reconfigured process model and a complex maneuver presented in [27] is studied with the reference commands as shown in Table 8.

A MATLAB™ program (aircraftReconfiguration.m) was developed to study this maneuver. Initially, the aircraft was trimmed with steady wing level flight at an altitude of 10000 ft. and velocity of 500 ft/sec (Case-2). The simulation results are shown
In Figure 52, it is evident from the results that, the designed adaptive control scheme enables highly nonlinear aircraft to perform the desired maneuver by following the reference commands. The 3-D trajectory of aircraft motion during the entire flight is shown in Figure 53.
In summary, this chapter has presented the application of the MIMO STR based adaptive control scheme for high performance 6-DOF aircraft control. Various performance aspects of designed scheme are evaluated and verified by simulations in the MATLAB™. An ability of proposed controller to perform complex maneuvers is also demonstrated in presented analysis. The performance qualities like adaptation,
Fig. 50. MIMO STR performance for coordinated maneuver in damaged aircraft case.

robustness, recovery and survivability of the proposed control scheme are also verified with a damaged aircraft case.
Fig. 51. 3-D trajectory of aircraft during coordinated maneuver.

### TABLE 8
**REFERENCE COMMANDS FOR COMPLEX MANEUVER**

<table>
<thead>
<tr>
<th>Reference</th>
<th>Units</th>
<th>0 &lt; t • 2</th>
<th>2 &lt; t • 10</th>
<th>10 &lt; t • 15</th>
</tr>
</thead>
<tbody>
<tr>
<td>Angle of Attack (°)</td>
<td>deg</td>
<td>2.4</td>
<td>10</td>
<td>-2</td>
</tr>
<tr>
<td>Roll Angle (°)</td>
<td>deg</td>
<td>0</td>
<td>50</td>
<td>0</td>
</tr>
<tr>
<td>Sideslip Angle (°)</td>
<td>deg</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>
Fig. 52. Results of complex maneuver with reconfigured aircraft model.
Fig. 53. 3-D Trajectory of reconfigured aircraft motion.
CHAPTER V

CONCLUSIONS

An adaptive controller in the form of a Multi-Input Multi-Output Self Tuning Regulator (MIMO STR) was designed and developed to control a high performance fighter aircraft. It has been shown that the designed controller adapts its parameters in order to get the desired response irrespective of large nonlinearities in the process and/or variation in process dynamics. The performance results obtained through simulations prove that, the design methodology presented in this thesis is applicable for class of multivariable systems.

The research starts with a review of literature in the field of flight control that reveals various flight control methodologies in practice. The traditional approach in flight control incorporates fixed gain or gain-scheduled linear controller. This controller is designed by considering specific operating conditions and does not guarantee robustness and performance over complete flight envelope especially for complex maneuvers. Another common approach is, decoupling longitudinal and lateral dynamics to simplify control design problem. In the present day advanced aircrafts, there is a strong cross-coupling between longitudinal and lateral dynamics; hence decoupling approach may lead to poor performance qualities for such aircrafts.

The modern control laws propose multiple control tools like neural network, fuzzy control, dynamics inversion, H , µ-synthesis and robust adaptive control to name a
few. These tools are centered on an idea of a controller with parameters adaptation, reconfiguration and robust behavior. An implementation of the modern control laws show two control loops; one for close loop feedback and other to control process nonlinearities, parameter uncertainties and variations in process dynamics. These nonconventional design techniques expand the flight envelope beyond the capabilities of traditional linear controllers and enable pilots to perform complex maneuvers on fighter aircrafts.

A mathematical model of high performance aircraft (F-16) and simulation model development framework was discussed. The need of an adaptive control scheme for aircraft control problem was demonstrated by using gain scheduling and fixed gain controller on aircraft simulation model. It was observed that, fixed gain controller exhibits a poor performance when aircraft operates outside of controller design point. This observation demonstrates a need of controller parameters adjustments based on the flight conditions and hence, an adaptive control scheme.

The proposed design methodology deals with multivariable systems and is based on adaptive control, system identification and optimal control theory. An overview of these tools has been presented. Further, the Least Square Error (LSE) based system identification procedures used for the class of Single Input Single Output (SISO) systems were extended to accommodate multivariable systems. In order to improve computation efficiency, Recursive Least Square Error (RLS) estimation was incorporated in the developed MIMO system identification algorithm. The Variable Forgetting Factor (VFF) feature was introduced in system identification procedure to improve identification
results. A simulation assessment has presented to justify VFF superiority over constant forgetting factor implementation.

The MIMO system identification procedure yields representation of an aircraft dynamics in the form of a linear state space model. This representation was validated by comparing model output with that of the process and thus, the robustness of identification algorithm was demonstrated. An online optimal controller design procedures were implemented for identified system model. Performance of designed controller was assessed by using it on aircraft model in both configurations: regulator mode and servo control mode. The satisfactory performance of the MIMO STR has been demonstrated through various simulation exercises.

Various performance aspects of the designed scheme were evaluated and verified by simulations in MATLAB™. The ability of proposed controller to perform complex maneuvers was demonstrated. It was observed that, the proposed MIMO STR provides improved performance over multiple liner controllers designed for a specific operating conditions. The performance qualities like adaptation, robustness, recovery and survivability of proposed control scheme were also verified with a damaged aircraft case. The most prominent feature of proposed design technique is to accommodate any dynamic process without knowing its mathematical model. The same is demonstrated by reconfiguring aircraft output vector.

Thus, a control design methodology for a class of multivariable systems has been successfully applied to fighter aircraft control problem. The proposed adaptive controller shows superior performance compared with linear controllers. The real-time online system identification and controller design computations performed at each update
interval captures nonlinearity and/or varying dynamics of the process and adjusts the controller parameters to get desired response.
REFERENCES
REFERENCES


113


LIST OF SIMULATION PROGRAMS

All simulations were conducted in the simulation environment of MATLAB™. A complete listing of programs code can be obtained in the Department of Electrical and Computer Engineering at California State University, Chico. This listing reflects the organization of the compact disk located in the department office containing the programs.

Chapter II

aircraftSimulation.m

linearControllerDesignCase1to4.m

linearControllerPerformance.m

Chapter III

mimoSystemIdentification.m

mimoRegulator.m

mimoServoControl.m

Chapter IV

mimoSTRforAircraft.m

aircraftTestCases.m

mimoSTRandLinearController.m

mimoSTRforAircraftCases.m

mimoSTRandFixedGainComparison.m

aircraftRecoveryAnalysis.m

aircraftReconfiguration.m