

Adaptive Envelope Protection Methods for Aircraft

A Thesis
Presented to
The Academic Faculty

by

Suraj Unnikrishnan

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy

School of Aerospace Engineering
Georgia Institute of Technology
August 2006

Copyright © 2006 by Suraj Unnikrishnan

Adaptive Envelope Protection Methods for Aircraft

Approved by:

Dr. J.V.R. Prasad
Committee Chair
School of Aerospace Engineering
Georgia Institute of Technology

Dr. Amy R. Pritchett
School of Aerospace Engineering
Georgia Institute of Technology

Dr. Eric N. Johnson
School of Aerospace Engineering
Georgia Institute of Technology

Dr. Eric Feron
School of Aerospace Engineering
Georgia Institute of Technology

Dr. Joseph F. Horn
Department of Aerospace Engineering
The Pennsylvania State University

Date Approved: 18 May 2006

DEDICATION

To my grandfather,

Alandath Balakrishnan Nair,

*who has always inspired me to improve myself and strive for
perfection, and*

To my grandmother,

Kallath Padmavathy Amma,

for her unconditional love and support.

ACKNOWLEDGEMENTS

I would like to thank my adviser Dr.J.V.R. Prasad for providing me with this unique opportunity to study and conduct research at one of the best engineering schools in the country. Without his patience, guidance, and continued support I couldn't have made it this far. I would also like to thank my committee members, Dr.Amy Pritchett, Dr.Eric Johnson, Dr.Eric Feron, and Dr.Joe Horn from Penn state for taking the time to review my work and providing valuable suggestions to improve my thesis. Special thanks to Dr.Anthony Calise for advising me during my first year at Tech and letting me continue as a member of his research lab. Many thanks to Dr.Naira Hovakimyan for inspiring and working tirelessly towards improving the theoretical aspects of our research. My deepest gratitude to all my teachers here at Georgia Tech for their dedication and effort in instilling deep and fundamental understanding of course material.

I would like to acknowledge fellow carefree researcher Dr.Ilkey Yavrucuk for trusting me with the implementation of his algorithms on various platforms and helping me gain more insight into envelope protection methods. My sincere thanks also goes to Dr.Geoffrey Jeram, a devout carefree lab member until recently, for his help in implementing and testing envelope protection algorithms within the RIPTIDE simulation environment. Without his efforts carefree lab and active sidestick workshop would not have existed.

Many thanks to Dr.Eric Johnson, Jeong Hur, Henrik Christophersen, Wayne Pickell and other UAV lab members for their help in implementing and flight evaluation of envelope protection algorithms on GTMax. I sincerely appreciate their effort and dedication in building such a unique, reliable and robust unmanned robotic flight

platform. Special thanks to Dr.Suresh Kannan for his significant contributions in the development of GTMax and for helping me personally understand GTMax flight architecture. I sincerely thank Dr.Manuj Dhingra for his friendship, time, and support in setting up as well as maintaining a functioning Linux operating system on my desktop. I thank my friend Dr.Sriram Rallabhandi for his help in improving my software implementations and for many insightful out of the box discussions on our research problems. His friendship and company has been pivotal during my stay at Georgia Tech.

I sincerely appreciate the support and friendship of my roommates Ramachandra Sattigeri and Dr.Venkatesh Madyastha. Venky's dedication towards his work has been truly inspirational. I am deeply indebted to him for always looking out for me and always willing to help. I have known Rama from my undergraduate days at IIT and cherish his friendship deeply. I enjoyed the time we spent together and cannot think of anyone else who would have tolerated me for so long. Special thanks to Matt Johnson for sparing his valuable research time in doing all those little but significant things for our lab that made such a big difference for all of us. Thanks to other past, present and alumni members of controls lab, Dr.Ali Kutay, Dr.Bon-Jun Yang, Dr.Manu Sharma, Dr.Nakwan Kim, Dr.Yoonghyun Shin, Adrian Koller, Allen Wu, Jonathan Muse, Konstantin Volyanskyy, Kilsoo Kim, Nimrod Rooz, and Yoko Watanabe for never shying away from a discussion and for being always ready to help. I wish them all the very best in all their future endeavours. I am also thankful for the opportunity to work and interact with Dr.Graham Drozeski, Phillip Jones, Cheng Chang and Jongki Moon.

I would also like to acknowledge the CERT at Georgia Tech and Software Enabled Control program for funding this research. My sincere appreciation and gratitude to the faculty at the school of aerospace engineering for conducting excellent research that preserves this institute's place among the nation's best.

I am thankful to all my family particularly my dear sister Surya, my brother-in-law Sajeesh and my parents, V.S.Unnikrishnan, and Indira Unnikrishnan, unni mama, and valiamma for their unconditional love and support. I am also thankful to my in-law's V.P.Manohar, and Nalini Manohar, amma, thatha, madras thatha, lily amma, sasi atha, mama, Arun and many others who have always prayed and wished for my success. Finally, I would like to thank my wife Anitha for her love, support and friendship. You have brought beauty and meaning into my life. You make everything worthwhile.

TABLE OF CONTENTS

DEDICATION	iii
ACKNOWLEDGEMENTS	iv
LIST OF TABLES	x
LIST OF FIGURES	xi
LIST OF SYMBOLS OR ABBREVIATIONS	xv
SUMMARY	xvii
I INTRODUCTION	1
1.1 Envelope protection system design- approach and challenges	2
1.1.1 Modular envelope protection system design	3
1.1.2 Design challenges	5
1.2 Envelope protection- existing methodologies and techniques	6
1.2.1 Limit cueing for aircraft envelope protection	6
1.2.2 Existing limit prediction and cueing algorithms for aircraft	8
1.3 Thesis objectives and outline	15
II ADAPTIVE ESTIMATE OF LIMIT PARAMETER DYNAMICS 20	
2.1 Problem formulation	21
2.2 Neural networks	22
2.3 Reduced order error observer	28
2.4 Single hidden layer weight adaptation	29
2.5 Simulation results	34
2.5.1 Linear plant	35
2.5.2 Van der Pol oscillator	40
III OPTIMAL CONTROL FORMULATION AND NONLINEAR TRAJECTORY GENERATION FOR ENVELOPE PROTECTION	44
3.1 Formulation of optimal control problem	45
3.2 Control limit as the area norm	49

3.3	Control limits from optimal solution- example	51
3.4	Introduction to B-splines	53
3.5	Nonlinear trajectory generation for real-time optimal solution	57
3.5.1	B-spline approximation of continuous functions	58
3.5.2	Approximation of smooth functions using B-splines	60
3.5.3	NTG real-time control limit estimation using true limit parameter dynamics	61
3.5.4	NTG real-time control limit estimation using adaptive estimate of limit parameter dynamics	65
3.6	Extension of envelope protection to multiple control limiting	68
3.7	Pilot-in-the-loop evaluations	71
3.7.1	Development and testing environment	71
3.7.2	Test maneuvers	72
3.7.3	Force-feedback tactile cueing for hub moment limit protection	75
3.7.4	Post-inceptor command shaping for high frequency limit protection	79
3.8	Safety and performance evaluation	87
3.8.1	Quantitative evaluation using sloop maneuver results	87
3.8.2	Qualitative evaluation of NTG and nonlinear function response method	93
3.8.3	Analysis of nonlinear function response method and NTG based limit protection method using linear spring-mass-damper	96

IV REACTIONARY APPROACH FOR AUTOMATIC ENVELOPE PROTECTION 97

4.1	Methodology	97
4.2	Framework	98
4.3	Step-1: Finite time horizon based prediction of envelope violation	98
4.4	Step-2: Safe response profile	100
4.4.1	Safe response profile for $r = 1$	103
4.4.2	Safe response profile for $r > 1$	104
4.5	Tracking safe response profile	105

4.6	Linear simulation results	107
4.7	GTMax integrated simulation and flight testing architecture	111
4.8	Software-in-the-loop evaluation of reactionary load factor limit protection system	114
4.9	Flight evaluation of reactionary load factor limit protection system .	123
4.10	R22 longitudinal flap angle limiting	128
4.11	Calculation of control margin using reactionary envelope protection method	132
4.11.1	Reactionary envelope protection method for hub moment limit protection	135
4.11.2	Example simulation- R-22 engine manifold pressure limit protection	137
V	THESIS CONTRIBUTIONS, CONCLUSIONS AND RECOMMENDED FUTURE WORK	143
5.1	Conclusions and contributions	143
5.2	Recommended future work	147
APPENDIX A	— COMPUTATION OF B-SPLINES AND THEIR APPROXIMATION PROPERTIES	150
APPENDIX B	— TRACE IDENTITIES	157
APPENDIX C	— ANALYSIS OF SPRING-MASS-DAMPER SYSTEM FOR APPLICATION OF NONLINEAR FUNCTION RESPONSE METHOD	158
REFERENCES	162
VITA	169

LIST OF TABLES

1	Modeling of flat outputs for the spring-mass-damper system by NTG	63
2	NTG linear example- neural network design and learning rate parameters	66
3	Swoop maneuver performance specification	74
4	Hub moment limit protection- NN design and learning rate parameters	75
5	Modeling of flat outputs for hub moment limit protection system . . .	76
6	Vehicle configurations used for evaluating hub moment limit protection system	76
7	Reactionary envelope protection linear example- neural network design and learning rate parameters	108
8	GTMax load factor limiting- neural network design and learning rate parameters	114
9	Maverick UAV flap angle limit protection- neural network design and learning rate parameters	129

LIST OF FIGURES

1	An open limit protection architecture and vehicle control/command system Source: Reference [23]	4
2	Automatic envelope protection for UAV- Control limiting architecture	7
3	Automatic envelope protection for UAV- Command limiting architecture	7
4	Control limit for force-feedback tactile cueing	9
5	Limit parameter step response	13
6	Generic structure of a single hidden layer neural network	23
7	Block diagram representation of the adaptive estimation architecture	25
8	Robust inner-loop compensator architecture	27
9	Error observer RIC architecture	29
10	Linear plant- Response comparison with adaptation OFF- <i>Approximate model-1</i>	36
11	Linear plant- Response comparison with adaptation ON- <i>Approximate model-1</i>	37
12	Linear plant- Variation in neural network weights during adaptation - <i>Approximate model-1</i>	37
13	Linear plant- Response comparison with adaptation OFF- <i>Approximate model-2</i>	39
14	Linear plant- Response comparison with adaptation ON- <i>Approximate model-2</i>	39
15	Linear plant- Variation in neural network weights during adaptation- <i>Approximate model-2</i>	40
16	Block diagram representation of the nonlinear controller Van der Pol oscillator system	41
17	Van der Pol oscillator limit estimation with adaptation OFF	42
18	Van der Pol oscillator limit estimation with adaptation ON	42
19	Spring-mass-damper- Optimal solution for $W=1.0$	52
20	Spring-mass-damper- Optimal solution for $W=5.0$	52
21	Spring-mass-damper response with envelope protection OFF, $W=0.5$	62

22	Spring-mass-damper response with envelope protection ON, W=0.5	62
23	Spring-mass-damper response with envelope protection ON, W=5.0	63
24	Spring-mass-damper limit parameter response variation with control weighting	64
25	Spring-mass-damper control limit estimation using adaptive estimate-Envelope protection OFF, W=5.0	66
26	Spring-mass-damper control limit estimation using adaptive estimate-Envelope protection ON, W=5.0	67
27	Spring-mass-damper control limit estimation using adaptive estimate-Envelope protection OFF, W=1.0	70
28	Spring-mass-damper control limit estimation using adaptive estimate-Envelope protection ON, W=1.0	70
29	Open platform for limit protection	72
30	Active sidestick inceptor- stirling dynamics model SA-S-2D-1	73
31	Carefree maneuver workshop setup at Georgia Tech	73
32	Graphic representation of sloop maneuver	74
33	Smoothing function for hub moment limit protection	77
34	Sloop maneuver without hub moment limit protection- Nominal configuration	77
35	Sloop maneuver with hub moment limit protection- Nominal configuration	78
36	Sloop maneuver with hub moment limit protection- Nominal configuration	80
37	Sloop maneuver with hub moment limit protection and DC compensation- Nominal configuration	81
38	Post-inceptor command shaping using dynamic overshoot compensation- Nominal configuration	82
39	Sloop maneuver without hub moment limit protection- Heavy configuration	83
40	Sloop maneuver with hub moment limit protection- Heavy configuration	84
41	Sloop maneuver with hub moment limit protection and DC compensation- Heavy configuration	84

42	Swoop maneuver with hub moment limit protection and FD compensation- Heavy configuration	85
43	Post-inceptor command shaping using frequency distribution scheme- Heavy configuration	86
44	Time when limit parameter response is within 10% of the boundary vs Integrated hub moment limit exceedance factor.	88
45	Time when limit parameter response is within 10% of the boundary vs maximum absolute peak.	89
46	Maneuver time vs Integrated hub moment limit exceedance factor . . .	91
47	Maneuver time vs maximum absolute peak	92
48	Swoop maneuver velocity and attitude responses with and without NTG cueing	92
49	Spring-mass-damper limit parameter response comparison with NTG and smoothing function ($W = 5.0$) and nonlinear function response method based envelope protection	96
50	Reactionary envelope protection method and obstacle avoidance . . .	99
51	Safe response profile for envelope protection near limit boundary . . .	101
52	Linear example- Limit parameter response with reactionary envelope protection OFF	109
53	Linear example- Limit parameter response with reactionary envelope protection ON	110
54	Georgia Tech unmanned aerial vehicle testbed- GTMax	111
55	GTMax simulation architecture	112
56	GTMax software-in-the-loop evaluation architecture	116
57	Graphic representation of the aggressive turn maneuver	117
58	SITL results- Load factor response with envelope protection OFF . . .	118
59	SITL results- Load factor response with envelope protection ON . . .	119
60	Pictorial representation of inertial and vehicle carried reference frames	120
61	SITL results- Load factor response comparison with and without en- velope protection	121
62	SITL results- Velocity response profiles with and without envelope pro- tection	122

63	SITL results: aggressive turn trajectory with and without load factor limit protection	122
64	Hardware-in-the-loop test architecture	124
65	Flight test results- GTMax load factor response with envelope protection OFF	125
66	Flight test results- GTMax load factor response with envelope protection ON	126
67	Flight test results- GTMax load factor response comparison with and without envelope protection	127
68	Flight test results- GTMax velocity response profiles with and without envelope protection	127
69	Flight test results- aggressive turn trajectory comparison with and without load factor limit protection	128
70	R22- Response with envelope protection OFF	130
71	R22- Response with envelope protection ON	131
72	R22- Safe response tracking for envelope protection	131
73	R22- Comparison of flap angle response with and without envelope protection	132
74	Control limits for hub moment limit protection- Reactionary vs non-linear function response method	136
75	Control limits for hub moment limit protection- Reactionary vs NTG method	137
76	Flight test architecture	139
77	Engine intake manifold pressure model validation	140
78	SITL validation of reactionary engine manifold pressure limit protection system	140
79	Engine manifold pressure response- Envelope protection OFF	141
80	Engine manifold pressure response- Envelope protection ON	142
81	Engine manifold pressure response comparison with and without envelope protection	142

LIST OF SYMBOLS OR ABBREVIATIONS

DC	Dynamic overshoot compensation.
ERITS	Equivalent Retreating Indicated Tip Speed.
FCS	Flight Control System.
FD	Frequency distribution.
GCS	Ground Control Station.
HITL	Hardware-in-the-loop.
IHMLEF	Integrated Hub Moment Limit Exceedance Factor.
NN	Neural Network.
NPSOL	Software package for solving constrained optimization problems.
NTG	Nonlinear Trajectory Generator.
OFE	Operational Flight Envelope.
OPLP	Open Platform for Limit Protection.
PNN	Polynomial Neural Network.
RIC	Robust internal-loop compensator.
RIPTIDE	Real-time Interactive Prototype Technology Integration/Development Environment.
SHL	Single Hidden Layer.
SITL	Software-in-the-loop.
UAV	Uninhabited Aerial Vehicle.
Δt_{crit}	Critical time of limit parameter response.
y_p	Limit parameter.
y_{lim}	Limit parameter upper or lower limit value.
Π_{n-1}	Linear space formed by polynomials of order n .
N_z	Load factor.
$\Delta - \nu_{ad}$	Modeling Error.

Δ	Modeling uncertainty.
Δt_{sim}	Simulation time-step.
r	relative degree of limit parameter.
u_{AN}^*	Area norm of the optimal control solution.
Δt_h	Threshold time.
$tr(\cdot)$	Matrix trace operator.

SUMMARY

Carefree handling refers to the ability of a pilot to operate an aircraft without the need to continuously monitor aircraft operating limits. At the heart of all care-free handling or maneuvering systems, also referred to as envelope protection systems, are algorithms and methods for predicting future limit violations. Recently, envelope protection methods that have gained more acceptance, translate limit proximity information to its equivalent in the control channel.

Envelope protection algorithms either use very small prediction horizon or are static methods with no capability to adapt to changes in system configurations. Adaptive approaches maximizing prediction horizon such as dynamic trim, are only applicable to steady-state-response critical limit parameters. In this thesis, a new adaptive envelope protection method is developed that is applicable to steady-state and transient response critical limit parameters. The approach is based upon devising the most aggressive optimal control profile to the limit boundary and using it to compute control limits. Pilot-in-the-loop evaluations of the proposed approach are conducted at the Georgia Tech Carefree Maneuver lab for transient longitudinal hub moment limit protection.

Carefree maneuvering is the dual of carefree handling in the realm of autonomous Uninhabited Aerial Vehicles (UAVs). Designing a flight control system to fully and effectively utilize the operational flight envelope is very difficult. With the increasing role and demands for extreme maneuverability there is a need for developing envelope protection methods for autonomous UAVs. In this thesis, a full-authority automatic envelope protection method is proposed for limit protection in UAVs. The

approach uses adaptive estimate of limit parameter dynamics and finite-time horizon predictions to detect impending limit boundary violations. Limit violations are prevented by treating the limit boundary as an obstacle and by correcting nominal control/command inputs to track a limit parameter safe-response profile near the limit boundary. The method is evaluated using software-in-the-loop and flight evaluations on the Georgia Tech unmanned rotorcraft platform- GTMax. The thesis also develops and evaluates an extension for calculating control margins based on restricting limit parameter response aggressiveness near the limit boundary.

CHAPTER I

INTRODUCTION

Aircraft, both manned and unmanned, are constrained by many operating limits. Envelope protection is the task of monitoring and maintaining vehicle operation within these limits. Traditionally, in piloted vehicles the task of envelope protection is vested with the pilot. Pilots are trained to monitor cockpit instruments and follow safety guidelines for envelope protection. Experienced pilots also rely on secondary vehicle cues such as structural vibration while operating close to the edges of the flight envelope. However, a pilot tasked with envelope protection invariably will have higher workload, particularly when operating the vehicle aggressively at the edges of the prescribed flight envelope [18]. Furthermore, safety guidelines are usually designed conservatively and in a way that is easy for the pilot to follow. Even though the pilot may be aware of this conservative nature, he/she will be reluctant to disregard them even in critical situations. A pilot flying the vehicle within the constraints of the safety guidelines is generally operating within a conservative region of the true Operational Flight Envelope (OFE) and hence this approach does not fully qualify for carefree handling.

With the advent of digital avionics and advanced active control technology, carefree handling is becoming a reality [1,62]. Carefree handling [36] represents the ability of a pilot to fly throughout an aircraft's OFE without concern for exceeding structural, aerodynamic or control limits [33]. The anticipated benefits resulting from carefree handling are the following:

- Improved or guaranteed safety during highly aggressive maneuvers and while operating near the edges of the flight envelope.

- Reduced task time by allowing the pilot to utilize maximum available performance.
- Reduced pilot workload and improved mission related situational awareness by relieving the pilot of the need to continuously monitor cockpit instruments for envelope protection.

Imposing conservative safety constraints within the Flight Control System (FCS) can prevent the aircraft from violating operating limits. However, such safety constraints do not fully qualify for carefree handling as they do not facilitate effective usage of the operational flight envelope and therefore restrict the performance of the vehicle.

Envelope protection is also significant for Uninhabited Aerial Vehicles (UAVs). UAVs are emerging as new frontiers replacing piloted vehicles in many operational roles. Advanced technologies developed previously for piloted vehicles are now being transitioned onto autonomous aircraft. Maneuverability constraints that were imposed on piloted vehicles for the safety of the pilot are no longer applicable to UAVs. Therefore, technology will enable modern UAVs to be more maneuverable and more aggressive than manned vehicles. However, the absence of a human operator means that the task of envelope protection must be done autonomously. Carefree maneuvering, i.e, the ability of the vehicle to operate within the flight envelope without external monitoring, will be a necessary feature in the UAV system architecture.

1.1 Envelope protection system design- approach and challenges

The main purpose of an envelope protection or limit protection system is to enable carefree handling/maneuvering while reducing the compromise between safety and performance. A conventional approach for achieving carefree handling in piloted vehicles and carefree maneuvering in UAVs is to design the flight control system incorporating the operating limits of the vehicle [52, 53]. The benefits gained from using

this design strategy will vary and depend upon the time and effort expended during the design phase. Reasonable performance improvements may not be achievable without devoting significant financial and human resources. However, the obvious disadvantage of such an integrated design is that it will further complicate the design of the flight control system. Furthermore, such an architecture will not facilitate introducing new operating limits which may invariably require a complete re-design and re-verification of the flight control system.

1.1.1 Modular envelope protection system design

In lieu of the complications and perhaps only marginal benefits gained from utilizing an integrated design strategy, researchers have pursued a modular design approach in which an envelope protection system is designed independently from the vehicle flight control system. The design of an envelope protection system can be divided into the following functional modules: limit cue modules, a limit arbitration module, and a control interface module. The envelope protection system designer may wish to pursue either an integrated or modular design approach for these functional modules depending on the available resources and design goals. However, a decoupled modular design will facilitate change and renewal of individual modules.

The design choices made during the development of an envelope protection system and its functional modules rely heavily on how well the operating limits of the vehicle are understood. Among the three functional modules, the limit cue modules are the most significant and pervasive across various envelope protection systems. In a modular design, each limit cue module within an envelope protection system pertains to a particular vehicle operating limit, example wing loading, stall etc. A limit cue module combines within itself both the limit prediction and the limit cueing algorithms. The purpose of the limit prediction algorithm is to predict or anticipate an impending limit boundary violation. The function of the limit cueing algorithm is

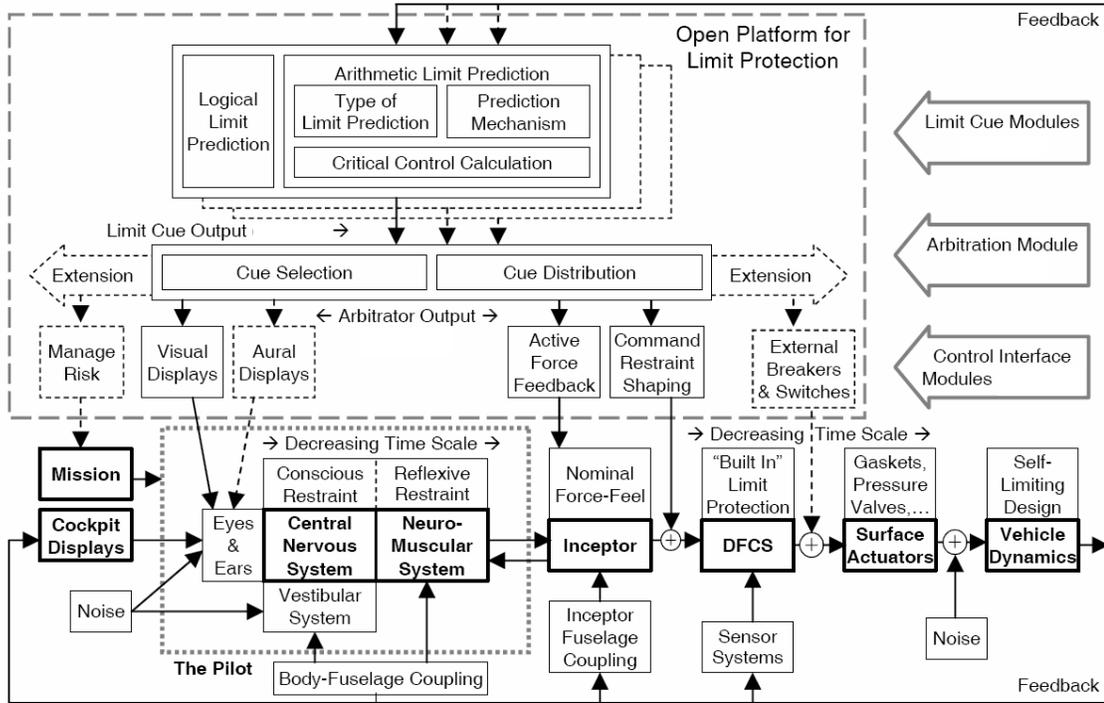


Figure 1: An open limit protection architecture and vehicle control/command system
Source: Reference [23]

to provide information to the limit arbitration module that will help prevent violation of the corresponding limit boundary, as shown in figure 1. The content and form of this limit cueing information will depend upon the limit cue design adopted for the operating limit and the vehicle platform. An envelope protection system may have multiple limit cue modules, each corresponding to an operating limit. The function of the limit arbitration module is to process and select among the multiple simultaneous limit cue information coming from the limit cue modules. This selection process within the limit arbitration module depends upon the design of the control interface module. The control interface module provides the points within the vehicle control/command path where limit cue information can be applied for limit protection. In the case of piloted vehicles, the operator is a part of the vehicle control/command path. Therefore, in addition to the interface points that are common to both manned

and unmanned aircraft, piloted vehicles have operator sensory channels such as vision, touch and hearing available as additional interface points, as shown in figure 1. The limit arbitration module must select, prioritize and distribute limit cue information to the various points across the vehicle control/command path using the control interface module [23].

1.1.2 Design challenges

Developing design methodologies and approaches for each functional module contained within a modular envelope protection system is a challenging problem. The focus of this thesis is the development of limit prediction and cueing algorithms that can be used in designing arithmetic limit cue modules. Arithmetic limit cue modules [23] are those that utilize analytical methods for limit prediction and limit cue determination. The alternative to arithmetic design approach is the logical limit cue design method. Logical limit cue modules are designed based on known or suspected cause and effect relationships between vehicle limits and inputs. Arithmetic limit cue design method is adopted for implementing limit cue modules for vehicle limits that have numerical values associated with them, such as stall, vertical load factor etc. The approach adopted for limit cue module design, logical or arithmetic, depends on available knowledge regarding the operating limit (limit parameter). For instance, logical limit cue design approach is adopted when the direction but not the magnitude of limit parameter variation with pilot control inputs are known. The next section reviews the various existing limit prediction and cueing methods for arithmetic limit cue module design. Also, the different approaches investigated for limit cueing in piloted and uninhabited aircraft are presented. The review will be used to converge upon the advantages as well as the drawbacks of existing methods. The methods proposed in this thesis will attempt to overcome these limitations while maintaining some of the desirable elements within the previous methodologies.

1.2 Envelope protection- existing methodologies and techniques

Researchers have adopted many different approaches for prediction and cueing of arithmetic limits. As mentioned previously, arithmetic limits have numerical values associated with them. Envelope protection in terms of arithmetic limits translates into maintaining limit parameter values either above and or below a certain specified value. Angle of attack, airspeed, bank angle and load factor are common examples of arithmetic limit parameters in fixed-wing aircraft. In comparison to fixed-wing aircraft, envelope protection for carefree handling/maneuvering in rotorcraft is a more challenging problem. This is because, in addition to conventional limits, rotorcraft have other unique limit parameters such as rotor flapping, hub moment etc. Furthermore, rotorcraft flight characteristics vary significantly between flight conditions. These issues prompted investigations into rotorcraft operating limits [61] and approaches for designing envelope protection system functional modules.

1.2.1 Limit cueing for aircraft envelope protection

In autonomous UAVs, limit cue information must be automatically incorporated into the vehicle control/command path. Therefore, the control interface module can inject limit cue information at two different points along the vehicle control/command path [46]. In the first architecture, shown in figure 2, the limit cue information is used to modify the commands from the flight control system to the vehicle actuators. The architecture shown in figure 2 is referred to as the control limiting architecture. The alternative to control limiting architecture is the command limiting architecture shown in figure 3. In this architecture, the limit cue information is used to modify higher level system commands issued to the flight control system. In most applications either one of the two architectures may be adopted for envelope protection.

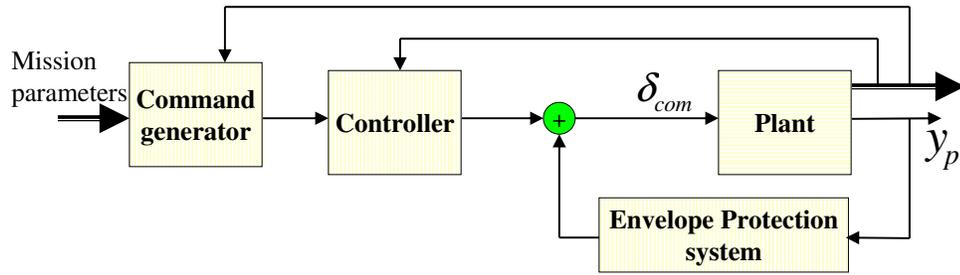


Figure 2: Automatic envelope protection for UAV- Control limiting architecture

However, in UAV systems utilizing high bandwidth or adaptive controllers, the command limiting architecture is preferable because modifications to low-level actuator commands may result in chattering or even instability. This chattering is similar to the oscillations around the limit boundary observed in manned envelope protection evaluations due to unexpected pilot response to force-feedback tactile cues.

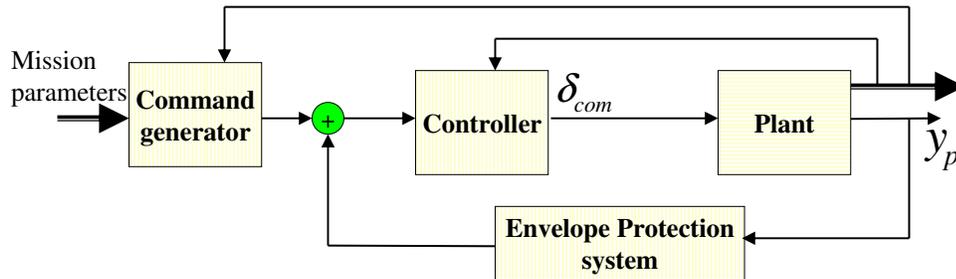


Figure 3: Automatic envelope protection for UAV- Command limiting architecture

1.2.1.1 Pilot cueing for limit protection

Automatic command/control limiting architectures can also be adopted for limit cueing and limit protection in piloted vehicles. However, in piloted vehicles human senses such as vision, touch and hearing are available as additional interface points for passing limit cue information. These interface points can be used for cueing the pilot regarding an impending or anticipated limit violation. Therefore, in manned vehicles

the envelope protection system designer can choose between voluntary pilot-in-the-loop limit protection and autonomous involuntary limit protection [5]. In both cases, the final decision authority for limit protection remains with the pilot. However, pilot-in-the-loop voluntary limit protection methodology emphasizes pilot judgment more than the safety of the vehicle and therefore, arguably leaves more decision authority with the pilot.

Researchers at NASA Ames investigated different aural, visual and tactile cueing methods for limit cueing [64, 41]. The study revealed that the discrete nature of the voice cues can cause the pilot to react suddenly which, in general, may not be desirable. The display cues were ineffective for limit parameters that change abruptly and therefore were difficult for the pilot to track. Tactile cueing using an active inceptor was found to be equally or more effective than other cueing methods. Tactile cues are perceived better by the pilot without actively looking for them. Also, specific forms of tactile cueing such as force-feedback tactile cueing can also provide the pilot with the corrective information necessary for preventing limit violation. In other words, force-feedback tactile cueing can be used to inform the pilot about the magnitude and direction of corrective response required for limit protection by guiding pilot input away from the control limit.

1.2.2 Existing limit prediction and cueing algorithms for aircraft

Implementing limit protection in manned vehicles using force-feedback tactile cueing requires the pilot to be able to perceive the limit boundary as an equivalent control limit in the active inceptor as shown in figure 4. The stick location on the active inceptor that corresponds to the control limit is referred to as the *critical control position*.

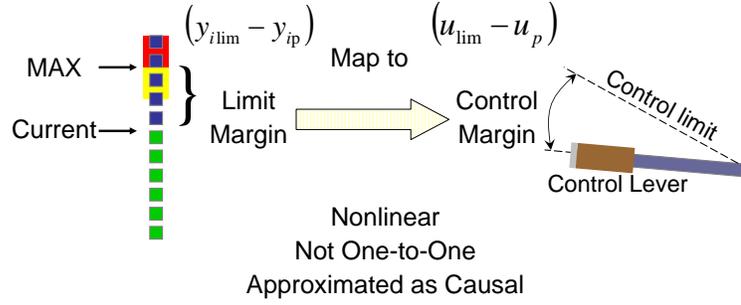


Figure 4: Control limit for force-feedback tactile cueing

1.2.2.1 Prediction horizon and control limit calculation

Control limit corresponding to the upper and lower limit boundaries are called as the upper and lower control limits, respectively. Control margin (shown in figure 4), is the difference between the control limit and the current control input. The control limit has to be determined from the nonlinear functional relationship that exists between the limit parameter response (y_p) and the control input (u_p). Since the true form of this nonlinear relationship is unknown, all envelope protection methods have to model or estimate this functional relationship. The control limit is defined as the control input that results in *future value* of the limit parameter response to be on the limit boundary. The prediction horizon used in calculating future limit parameter value is selected based on the accuracy of the limit parameter model. This prediction horizon is a critical variable that determines the effectiveness of an envelope protection method. When this prediction horizon is small or absent (instantaneous limit protection) the envelope protection system designer is forced to select additional safety margins to account for the dynamic nature of the limit parameter response [37]. Inappropriate selection of these safety margins will result in failure to fully utilize the operational flight envelope and, hence, the true performance of the vehicle. Therefore, the development of envelope protection methods that provide estimates of control limit, with timely prediction horizon, is a crucial requirement.

Researchers, particularly in the rotorcraft industry, have focused on developing

better methods for estimation and prediction of limit parameter dynamics. Some researchers such as Einthoven [7], have developed energy based methods for envelope protection. However, such approaches are restricted in application to a small class of limit parameters (torque). In the context of estimation and prediction of limit parameter dynamics, neural networks were increasingly being used for their ability to approximate continuous functions in a compact domain with any desired level of accuracy [8]. The following subsections present an overview of neural network based envelope protection algorithms proposed by researchers.

1.2.2.2 Neural networks for limit estimation and protection

Menon et.al [38] used artificial neural networks for both incorporating the rotorcraft manufacturer specified limits, and for adaptively establishing the relationship between various rotorcraft variables. Both off-line and on-line trained neural networks were used in the study to generate the limits on all rotorcraft variables using onboard measurements. The off-line neural networks were trained to represent rotorcraft manufacturer supplied limit data while the on-line trained networks (radial basis) were used to develop functional relationships between ill-defined limit variables that are too complex to be represented using compact expressions. The main limitation of the approach was in the use of simplified adaptive relationships for ill-defined limit variables, which can easily fail for complex limit parameters.

1.2.2.3 Polynomial neural networks for envelope protection

Bateman et.al [2] proposed an approach using off-line trained polynomial neural networks (PNN) for envelope protection. The PNNs were trained using limit parameter time-response data generated with multiple pilots flying the simulator. The time-response data was also generated for different flight regimes and vehicle conditions. The PNNs were trained to represent the input-output relationship between the current system variables and the future limit parameter response value. The prediction

horizon for the future limit parameter response was selected to yield the best closed loop performance, maximum linear correlation with current inceptor position and best pilot-in-the-loop performance during simulation evaluations.

In a similar investigation Whalley et.al [63] utilized PNN for predicting finite-time future response of Equivalent Retreating Indicated Tip Speed (ERITS) and torque. ERITS is a limit parameter corresponding to the main rotor blade stall. This PNN based architecture was also used by Jeram [21] for main rotor blade stall limit cueing using ERITS as the limit parameter. These studies clearly demonstrated the potential for using neural networks in estimation of limit parameter dynamics. Furthermore, the simulation evaluations showed that, for effective limit protection, reasonable amount of prediction horizon should be available.

1.2.2.4 Static neural network based dynamic trim method

Horn [14] proposed an envelope protection method based on *future* dynamic trim response of a limit parameter. Dynamic trim is a quasi-steady response condition of a limit parameter in which all the fast states affecting the limit parameter dynamics have evolved completely compared to the slow states that continue to evolve slowly.

Dynamic trim method based envelope protection system was successfully implemented for normal load factor and angle of attack limit protection within the V-22 tiltrotor aircraft simulation [9]. Dynamic trim data was generated off-line by sweeping through a range of influencing dimensionless variables and the envelope protection system was evaluated at the Boeing Flight Simulation Laboratory in Philadelphia. Dynamic trim method was also evaluated within the XV-15 simulation environment for angle-of-attack, load factor, airspeed and torque limiting using force-feedback tactile cueing in the longitudinal cyclic channel [10]. Dynamic trim method was also used to train artificial neural networks for predicting system parameters critical to flight envelope in the Helicopter Active Control Technology program [43].

The main drawback of the dynamic trim approach was that adequate training of neural networks requires generating large amounts of dynamic trim data. Generating training data in the entire flight regime and particularly near the envelope boundaries is extremely difficult. Also, the accuracy of these non-adaptive neural networks trained off-line to represent the relationship between future dynamic trim response of a limit parameter with the current slow states and control input, cannot be guaranteed for flight and vehicle configurations not represented in the neural network training data. Horn introduced limited adaptation into his approach by utilizing a complementary filter [2] to generate time-response data of the limit parameter from the neural network dynamic trim prediction [11,13]. The output of the complementary filter is then compared to the actual sensor measurement of the limit parameter response and the resulting error used to adapt the neural network weights. The problem with this approach is that the complementary filter has to be a realistic representation of the limit parameter dynamics in order to obtain an accurate error value.

1.2.2.5 Adaptive dynamic trim envelope protection method

Adaptive dynamic trim, proposed by Yavrucuk [71], is also based on estimating the future dynamic trim response of a limit parameter. In the adaptive dynamic trim method, instead of generating dynamic trim data to train a neural network off-line, an approximate first order linear model representing the limit parameter dynamics is augmented with an adaptive single hidden layer neural network to cancel the resulting modeling uncertainty. The dynamic trim response of the limit parameter is then estimated by setting the derivative of limit parameter to zero. The critical control position is estimated as the control input for which the future dynamic trim response of the limit parameter is on the limit boundary. The presence of an adaptive neural network within the limit parameter estimation architecture allows the limit prediction

algorithm to adapt to changing flight and or vehicle configurations.

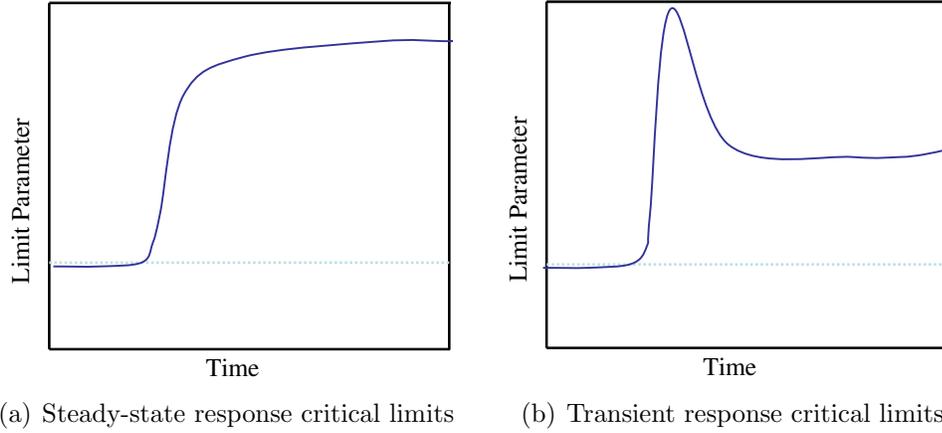


Figure 5: Limit parameter step response

The adaptive dynamic trim approach was shown to be an effective envelope protection algorithm using simulation evaluations within both manned [67, 69, 68, 66, 60] and unmanned system architectures [46, 70]. However, the adaptive dynamic trim approach carried over the limitations of the dynamic trim response condition. Such a quasi-steady response condition may not exist for all limit parameters. Also, the dynamic trim approach was essentially developed for maximizing the prediction horizon of the control limit calculation. However, the dynamic trim response is the *maximum* future limit parameter response only in the case of *steady-state-response critical limits* and not for *transient-response-critical limits*. A pictorial description is shown in figure 5 distinguishing the response of a steady-state-response critical limit against that of a transient-response critical limit. As seen in figure 5, steady-state-response critical limits are those limit parameters that reach their maximum step-response value in steady-state (load factor, angle of attack etc.). On the other hand, transient-response critical limits are limit parameters that reach their maximum step-response value in the transient phase (hub moment, flapping etc.).

1.2.2.6 Peak-response estimation method and nonlinear function response method

Envelope protection methods such as the peak-response estimation method and the nonlinear response function method were developed mainly for transient response critical limit parameters. The peak-response estimation method relied upon identifying a linear model for the transient-response critical limit parameter [15, 51]. Such a model may be difficult if not impossible to obtain. Furthermore, it may be valid for a very restrictive domain of flight and vehicle configurations.

The nonlinear function response method [49] utilized simulation or flight test data (if available) to represent limit parameter response in terms of nonlinear response functions. These functions are identified off-line using least-squares regression or other competent system identification schemes. Alternatively, the nonlinear response functions can also be approximated using neural networks. Even though the limit parameter response is represented in a similar functional form to that for peak-response estimation method, their approach for calculating control limits differ significantly. In the peak-response estimation algorithm the functional form of the limit parameter response is very simple because of the choice of linear model. Therefore, in the peak-response estimation algorithm the control limit is defined to be the input that results in the maximum step response of the limit parameter to be at the limit boundary. Estimating the control limit using the nonlinear response function method is more involved and complicated than the peak-estimation method because of the nonlinear representation of limit parameter response. The nonlinear response function method may require additional neural networks to approximate the maximum and minimum values of non-forced time response of limit parameter. The maximum and minimum values are then utilized to estimate the proximity of the limit parameter response to the upper and lower limit boundaries respectively. Additionally, these proximity values referred to as upper and lower limit margins, are then translated to the corresponding upper and lower control margin values by dividing the corresponding

limit margin with the maximum control sensitivity value (obtained from the nonlinear functional representation). Note that, this approach of computing control limit results in a conservative estimate of the actual control limit because control sensitivity will not take its maximum value at all times.

1.3 Thesis objectives and outline

The following conclusions can be drawn based upon the studies conducted by researchers into the various aspects of envelope protection system design, particularly envelope protection methods for design of arithmetic limit cue modules:

- Envelope protection methods that estimate control limits for limit protection are preferred for piloted vehicles and UAVs. This is because, in the case of UAVs, control limits can be used to automatically modify commands/control inputs to prevent limit violation. In piloted vehicles, control limits can be used to implement force-feedback tactile cueing for limit protection.
- The control limits are computed by identifying the control input that results in the future limit parameter response at the limit boundary. The prediction time horizon used in calculating the future response must be sufficient to account for the dynamic nature of limit parameter response. Also, in the case of manned vehicles, the prediction horizon should also account for additional delays associated with the dynamics of the active inceptor and pilot response to limit cues.
- Almost all the recent envelope protection methods proposed by researchers employ neural networks in some form or another to model limit parameter dynamics. Approaches using on-line adaptive neural networks, such as adaptive dynamic trim, are preferred over methods using off-line trained non-adaptive neural networks.

- Recent envelope protection methods such as adaptive dynamic trim and non-linear function response method have focused on identifying the functional relationship between the current control input and maximum future limit parameter response. This emphasis was based on the assumption that the limit cue should be available as early as possible for effective limit protection. Though this assumption has some validity in manned vehicles, it is not required for unmanned vehicles where control/command modifications can be instantly incorporated into the vehicle control/command path.

The survey of existing envelope protection methods reveals that there is a need to develop adaptive envelope protection methods, providing adequate prediction horizon, for transient-response critical limit parameters. There is also a need to develop an alternative envelope protection method to the adaptive dynamic trim method. The dynamic trim methods rely on the existence of dynamic trim response condition which is difficult to verify in many situations. On the basis of previous studies and current requirements, the following are the objectives for this thesis:

- Develop new envelope protection methods for estimating arithmetic control limits that can be applied within manned and or uninhabited aerial vehicles.
- Develop envelope protection methods applicable to both steady-state as well as transient response critical limit parameters.
- Develop envelope protection methods that will be able to utilize the adaptive estimate of limit parameter dynamics. Hence, the application of these envelope protection methods for designing limit cue modules will not require detailed off-line analysis or data generation.
- Develop envelope protection methods the application of which does not rely upon identifying functional relationships for the maximum limit parameter response. Instead, there should be design variables that can be varied to affect

the lead time available to the pilot or FCS to take corrective action for envelope protection.

- Develop new envelope protection methods that will aid in the full and effective utilization of the available/prescribed operational envelope.
- Finally, demonstrate and validate the proposed envelope protection methods using simulation and/or flight test evaluations.

There are many issues related to the application of envelope protection methods that are important from the perspective of an envelope protection system designer but are beyond the scope of this thesis.

- Selection of the most appropriate pilot cueing technique.
- Concerns related to incorrect pilot adaptation and adverse aircraft pilot coupling [42] when force-feedback tactile cueing is used for limit protection.
- An envelope protection system alerts the pilot regarding impending limit violations and according to this role is an alerting system [47]. This classification brings up issues regarding false alarms/alerts which maybe due to sensor noise and or incorrect measurements. However, in this research it is assumed that the designer has sufficient confidence in limit parameter measurements and the adaptive estimation process. Therefore, the effect of sensor noise and incorrect or insufficient adaptation resulting in false alarms is not investigated in this research.

This thesis is organized into five chapters. The first chapter presents an introduction to the concept of envelope protection and why it is significant. A brief overview of research related to envelope protection systems and methods is also presented. The methodology for adaptive estimation of limit parameter dynamics is presented in the

second chapter. This methodology for developing adaptive estimate of limit parameter dynamics is an extension derived from reference [71]. The approach for generating adaptive estimate of limit parameter dynamics is validated using linear and nonlinear simulation examples in chapter 2.

In chapter 3 a new envelope protection method is developed mainly for limit protection, using force-feedback tactile cueing, in manned systems. The approach is based on finding an optimal control profile that takes the limit parameter response, from its present value, to the limit boundary while minimizing an objective function with both time and control effort. A detailed description of the underlying methodology is presented along with the formulation for computing control limit using the optimal control profile. The proposed envelope protection method is then implemented within the Open Platform for Limit Protection (OPLP) as a limit cue module for longitudinal hub moment limit protection. This module is used to place softstops on an active inceptor for providing force-feedback tactile cues in the longitudinal cyclic channel of an active inceptor. The results from the pilot-in-the-loop simulation evaluations within the Real-time Interactive Prototype Technology Integration/Development Environment (RIPTIDE) are presented along with detailed analysis of the performance and effectiveness of the approach. The results from the Nonlinear Trajectory Generator (NTG) based limit cue module is also compared against a limit cue module designed using the nonlinear function response method for hub moment limit protection.

Chapter 4 presents a new automatic envelope protection method developed mainly for application in uninhabited aerial vehicles. This approach, referred to as reactionary envelope protection method, does not rely on translating the limit boundary information into its equivalent limit in the control channel. Instead, the method uses adaptive estimate of limit parameter dynamics and finite-time horizon predictions to predict or detect envelope violations. The limit boundary is treated as an obstacle

against which collision must be avoided. Limit boundary violations are prevented by correcting nominal system command/control inputs [see figures 2 and 3] so that the true limit parameter response is forced to track a prescribed safe-response profile near the boundaries. The proposed approach is demonstrated first using simple linear examples. A reactionary load factor limit protection system is implemented within the GTMax integrated simulation and flight test architecture. The results from the Software-in-the-loop (SITL) and flight test evaluations of this load factor limit protection system are presented. Reactionary envelope protection scheme is also used to implement and successfully evaluate a flap angle limit protection system for R22 using SITL tests. Furthermore, an extension to the reactionary envelope protection scheme is proposed that allows the approach to be used for estimating control limits for force-feedback tactile cueing. The control limits computed using this proposed extension scheme are compared against control limits estimated by the NTG based and nonlinear function response method for hub moment limit protection.

Finally, chapter 5 states the conclusions and presents suggested future work based upon this research.

CHAPTER II

ADAPTIVE ESTIMATE OF LIMIT PARAMETER DYNAMICS

The envelope protection methods proposed in this thesis, like many other competing approaches, requires a limit parameter model to be available. The main drawback of the existing model construction/estimation schemes used in envelope protection methods has been the lack of adequate adaptability or flexibility. This means that during any operation a vehicle is made to go through various flight and system configurations that influence the limit parameter dynamics. The most common and simple examples are weight and CG location changes. The limit parameter model used for envelope protection must be capable of adapting or modifying itself with changes in system configuration. However, most of the existing approaches either use static models based on off-line analysis of the system [15, 43] or use dynamic models with limited adaptation capability [50, 54]. A limit parameter estimation model that performs well for a wide range of vehicle/flight configurations and designed based on detailed model analysis would result in higher design costs and model complexity.

Neural networks have powerful function approximation capabilities and have recently emerged as significant components in adaptive control system design architectures [28, 65]. In this chapter, motivated from the use of neural networks in control system design, an adaptive architecture for estimation of limit parameter dynamics is presented. A single hidden layer neural network is trained on-line using gradient based weight update laws to cancel modeling uncertainties arising from approximate

modeling of limit parameter dynamics and/or system changes affecting limit parameter dynamics. Lyapunov analysis is used to show that under certain assumptions the error in neural network weights and limit parameter estimation will be ultimately bounded.

2.1 Problem formulation

The nonlinear system for which envelope protection system is being developed can be represented in the following generic state-space form:

$$\dot{\mathbf{x}} = f(\mathbf{x}, \mathbf{u}) \quad (1)$$

where $\mathbf{x} \in \mathfrak{R}^n$ and $\mathbf{u} \in \mathfrak{R}^p$ denotes the system states and control inputs respectively. Consider $y_p \in \mathfrak{R}$ to be a limit parameter which in general will be a nonlinear function of the system states as given in equation 2.

$$y_p = h(\mathbf{x}) \quad (2)$$

The relative degree of an output is defined as the minimum number of differentiations of the output required for the control variable to appear explicitly in the dynamic relationship. Therefore, if r is the relative degree of the limit parameter y_p then according to the definition of relative degree

$$y_p^{(r)} = h_r(\mathbf{x}, y_p, y_p^{(1)}, \dots, y_p^{(r-1)}, u_p) \quad (3)$$

where u_p is an element in the control vector \mathbf{u} . The following assumptions are required to develop the methodology for adaptive estimation of limit parameter dynamics.

Assumption 2.1.1. *The limit parameter has a well-defined and known relative degree.*

Assumption 2.1.2. *The limit parameter value is available or can be calculated from the available sensor measurements.*

Assumption 2.1.3. *The sign of the limit parameter control sensitivity ($\frac{\partial h_r}{\partial u_p}$) is known and the magnitude is also known to a reasonable upper bound.*

An approximate linear model is chosen for the limit parameter dynamics based on the relative degree and available system information.

$$\hat{y}_p^{(r)} = \hat{h}_r(\hat{y}_p, \hat{y}_p^{(1)}, \dots, \hat{y}_p^{(r-1)}, u_p) = \sum_{i=0}^{r-1} a_i \hat{y}_p^{(i)} + b u_p \quad (4)$$

where $\hat{h}_r = \sum_{i=0}^{r-1} a_i \hat{y}_p^{(i)} + b u_p$ represents the approximate linear model chosen for the limit parameter dynamics. Rewriting equation 4 in state-space form results in the following relation:

$$\dot{\hat{Y}}_p = \mathbf{A} \hat{Y}_p + \mathbf{B} u_p \quad (5)$$

where $\hat{Y}_p = \begin{bmatrix} \hat{y}_p & \hat{y}_p^{(1)} & \dots & \hat{y}_p^{(r-1)} \end{bmatrix}^T$ and

$$\mathbf{A} = \begin{bmatrix} 0 & 1 & 0 & 0 & \dots & 0 \\ 0 & 0 & 1 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ a_0 & a_1 & a_2 & a_3 & \dots & a_{r-1} \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ b \end{bmatrix} \quad (6)$$

Assumption 2.1.4. *The linear approximate model for the limit parameter dynamics [see equation 4] is stable with the matrix \mathbf{A} being Hurwitz.*

2.2 Neural networks

The response of the linear approximate model in equation 4, in general, will not match the true nonlinear limit parameter dynamics in equation 3. Therefore, it is necessary to augment the approximate model with an additional adaptive element that can capture the modeling error given in equation 7.

$$\Delta(\mathbf{x}, y_p, y_p^{(1)}, \dots, y_p^{(r-1)}) \triangleq h_r(\mathbf{x}, y_p, y_p^{(1)}, \dots, y_p^{(r-1)}, u_p) - \hat{h}_r(y_p, y_p^{(1)}, \dots, y_p^{(r-1)}, u_p) \quad (7)$$

A NN has been used as an adaptive element in many recent control architectures [17,24]. A single hidden layer neural network with sufficient number of neurons and the appropriate interconnection weights can approximate any continuous function in a compact domain to desired level of accuracy [8]. For obtaining an adaptive estimate of limit parameter dynamics the output of a single hidden layer neural network (SHL-NN), shown in figure 6, is used to augment the linear model in equation 5.

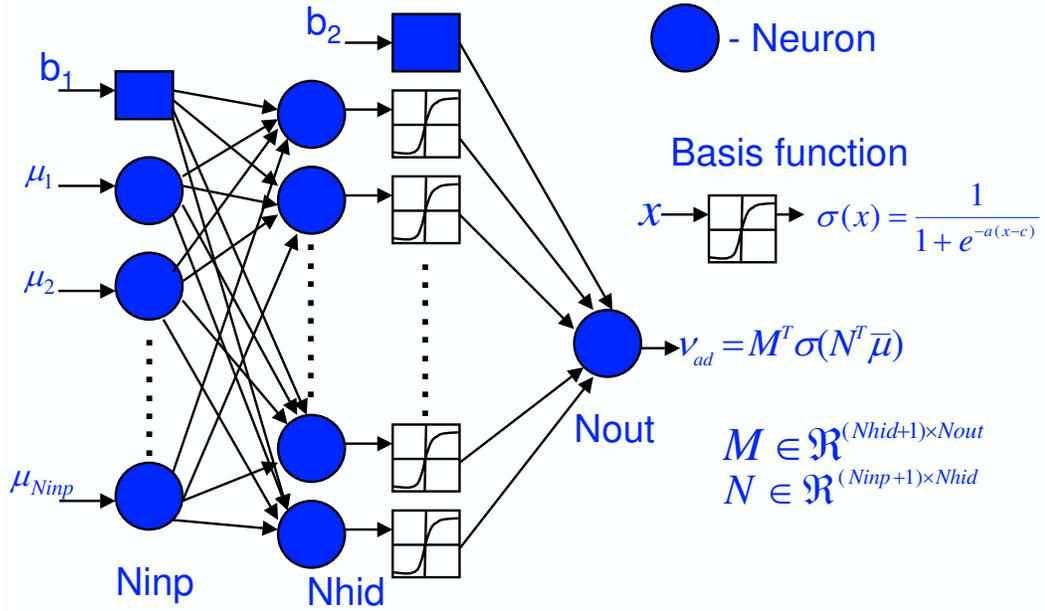


Figure 6: Generic structure of a single hidden layer neural network

A SHL-NN, as shown in figure 6, consists of three layers namely- input, hidden and output layer. Each layer contains a fixed number of neurons and each neuron represents a summing junction adding up all the incoming signals. Besides these fixed number of neurons, the input layer and hidden layer each contain a bias neuron. Every layer in this structure is connected to its adjacent layers and therefore the hidden layer is connected to both the input layer and output layer. The output from each neuron in the hidden layer, except for the bias neuron, passes through a function block called as the network basis function.

Each bias and non-bias neuron in the hidden layer is connected with every other

non-bias neuron in the adjacent layers. These interconnections have certain values associated with them called *weights*. For example, the values of interconnections between the input layer and hidden layer can be represented in a matrix form and is referred to as the hidden layer weight matrix. Similarly, the output layer weight matrix represents the value of interconnections between the output of the basis function blocks and the output layer. The dimension of these weight matrices are based upon the number of neurons that are contained in the individual layers. Let N_{inp} , N_{hid} and N_{out} be the number of neurons in the input, hidden and output layers respectively. Then,

$$\begin{aligned} \text{The hidden layer weight matrix-} & \quad \mathbf{N} \in \mathfrak{R}^{(N_{inp}+1) \times N_{hid}} \quad \text{and} \\ \text{the output layer weight matrix-} & \quad \mathbf{M} \in \mathfrak{R}^{(N_{hid}+1) \times N_{out}} \end{aligned} \quad (8)$$

Notice that the dimensions of the weight matrices also incorporate the interconnections between the bias neuron, which is treated separately, and the neurons in the adjacent layer. The phrase *neural net training* refers to tuning of the hidden layer and output layer weight matrices using gradient based back-propagation or similar methods. The weights of a SHL-NN can be tuned to adequately approximate any continuous function of input variables ($\bar{\mu}$) in a compact domain. When this tuning is done in real-time, with the event of interest taking place simultaneously (for example control), the process is called on-line training. The process of tuning weights on-line is simply referred to as weight adaptation.

A SHL-NN output is utilized to augment the linear approximate model in equation 4. The NN input vector consists of a normalized set of system variables that determine the value of the modeling error Δ . For example, in case of limit parameters with $r > 1$, higher order derivatives of limit parameter are usually not available. Therefore, adequate number of delayed values of limit parameter measurement are used instead

in the NN input vector [32].

$$\bar{\mu} = \begin{bmatrix} y_p(t) & y_p(t - t_d) & \dots & y_p(t - nt_d) \end{bmatrix}^T \quad (9)$$

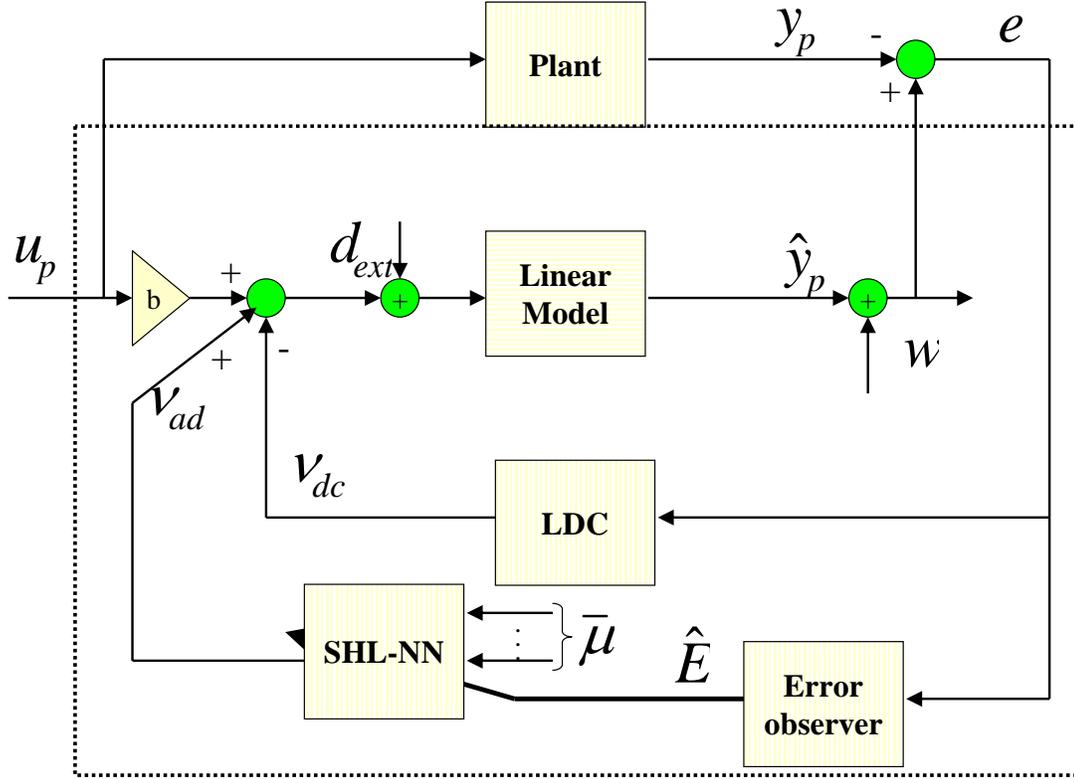


Figure 7: Block diagram representation of the adaptive estimation architecture

As the neural network weights are tuned on-line, the estimate of limit parameter dynamics obtained by augmenting the linear approximate model with the neural network is referred to as the adaptive estimate of limit parameter dynamics. A block diagram representation of the adaptive estimation architecture is presented in figure 7. The final differential equation form of the adaptive estimate is given in equation 10.

$$\begin{aligned} \hat{y}_p^{(r)} &= \hat{h}_r(\hat{y}_p, \hat{y}_p^{(1)}, \dots, \hat{y}_p^{(r-1)}, u_p) + \nu_{ad}(\bar{\mu}) - \nu_{dc} \\ \hat{y}_p^{(r)} &= \sum_{i=0}^{r-1} a_i \hat{y}_p^{(i)} + b u_p + \nu_{ad}(\bar{\mu}) - \nu_{dc} \end{aligned} \quad (10)$$

Notice that in addition to the neural network the linear approximate model is also augmented by a linear dynamic compensator. The linear dynamic compensator provides additional stability to the error dynamics by attenuating the effect of modeling error ($\Delta - \nu_{ad}$) and external disturbances (d_{ext}) on the limit parameter estimation error dynamics. The limit parameter estimate error (e) is defined as:

$$e \triangleq \hat{y}_p - y_p \quad (11)$$

The limit parameter estimation error dynamics is obtained by subtracting equation 3 from equation 10,

$$e^{(r)} = \sum_{i=0}^{r-1} a_i e^{(i)} + (\nu_{ad}(\bar{\mu}) - \Delta(\bar{\mu})) - \nu_{dc} \quad (12)$$

The error dynamics in equation 12 can be transformed into state-space representation of equation 13 using the following state vector $E \triangleq \begin{bmatrix} e & e^{(1)} & \dots & e^{(r-1)} \end{bmatrix}^T$

$$\dot{E} = \mathbf{A}E + \mathbf{B}_{(r,r)}(\nu_{ad} - \Delta) - \mathbf{B}_{(r,r)}\nu_{dc} \quad (13)$$

where $\mathbf{B}_{(j,r)} \in \mathbb{R}^r$ is a r -dimensional unit vector with its j^{th} element equal to one. The adaptive estimation architecture in figure 7 is structurally equivalent to a Robust internal-loop compensator (RIC) architecture shown in figure 8 [65].

Furthermore, the RIC architecture has been shown to be equivalent to a disturbance observer. If equation 14 presents the state-space form of the linear dynamic compensator then the matrices $\mathbf{A}_l, \mathbf{B}_l, \mathbf{C}_l, \mathbf{D}_l$ can be designed such that the closed loop system in figure 8 satisfies desired robustness criteria.

$$\begin{aligned} \dot{\eta} &= \mathbf{A}_l \eta + \mathbf{B}_l e \\ \nu_{dc} &= \mathbf{C}_l \eta + \mathbf{D}_l e \end{aligned} \quad (14)$$

Assumption 2.2.1. *There exist ideal output layer and input layer weight matrices \mathbf{M} and \mathbf{N} that can approximate the modeling error within a ball of radius ϵ .*

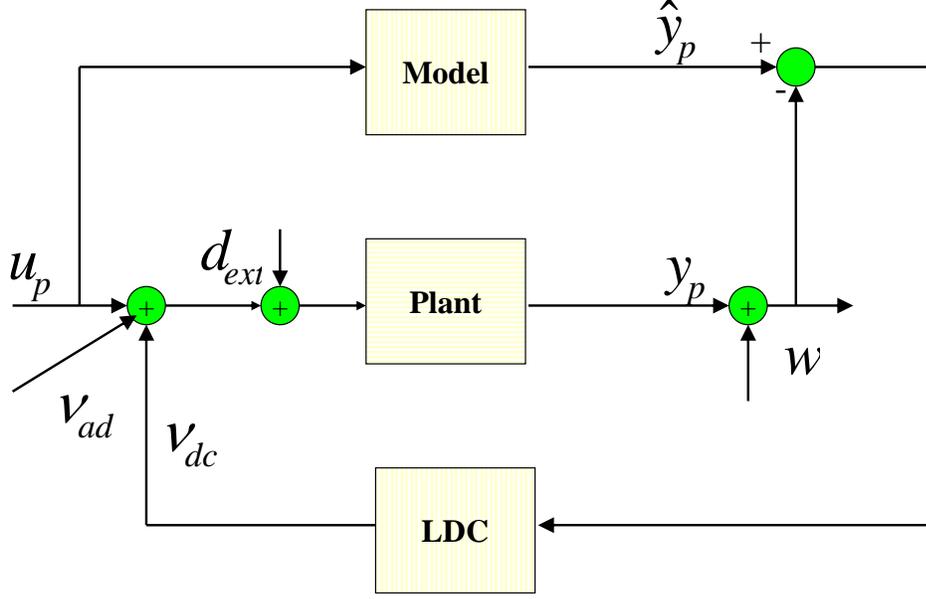


Figure 8: Robust inner-loop compensator architecture

According to assumption 2.2.1 the modeling error Δ can be computed using the SHL-NN if the ideal weights are known, i.e.,

$$\Delta = \mathbf{M}^T \sigma(\mathbf{N}^T \bar{\mu}) + \epsilon(\bar{\mu}) \quad (15)$$

where $\epsilon(\bar{\mu})$ is the function reconstruction error and $\|\epsilon(\bar{\mu})\| < \epsilon^*$. However, the ideal weights are not known and therefore NN adaptation laws must be formulated in such a way that the SHL-NN output tries to minimize the modeling uncertainty ($\nu_{ad} - \Delta$).

$$\nu_{ad} - \Delta = \hat{\mathbf{M}}^T \sigma(\hat{\mathbf{N}}^T \bar{\mu}) - \mathbf{M}^T \sigma(\mathbf{N}^T \bar{\mu}) - \epsilon(\bar{\mu}) \quad (16)$$

In equation 16 $\hat{\mathbf{M}}$, $\hat{\mathbf{N}}$ are the neural network output and input layer weight matrices, respectively. Using the Taylor series expansion formula

$$\begin{aligned} \mathbf{M}^T \sigma(\mathbf{N}^T \bar{\mu}) &= \mathbf{M}^T \sigma\left(\hat{\mathbf{N}}^T \bar{\mu} + (\mathbf{N} - \hat{\mathbf{N}})^T \bar{\mu}\right) \\ &= \mathbf{M}^T \left(\sigma(\hat{\mathbf{N}}^T \bar{\mu}) - \sigma' \tilde{\mathbf{N}}^T \bar{\mu} + O(\tilde{\mathbf{N}}^2) \right) \end{aligned} \quad (17)$$

where $\tilde{\mathbf{N}} \triangleq \hat{\mathbf{N}} - \mathbf{N}$ and $\tilde{\mathbf{M}} \triangleq \hat{\mathbf{M}} - \mathbf{M}$. Substituting equation 17 into equation 16 we get,

$$\begin{aligned}
\nu_{ad} - \Delta &= \hat{\mathbf{M}}^T \sigma(\hat{\mathbf{N}}^T \bar{\mu}) - \left(\mathbf{M}^T \left(\sigma(\hat{\mathbf{N}}^T \bar{\mu}) - \sigma' \tilde{\mathbf{N}}^T \bar{\mu} + O(\tilde{\mathbf{N}}^2) \right) \right) \\
&= (\hat{\mathbf{M}} - \mathbf{M})^T \sigma(\hat{\mathbf{N}}^T \bar{\mu}) + \mathbf{M}^T \sigma' \tilde{\mathbf{N}}^T \bar{\mu} - \mathbf{M}^T O(\tilde{\mathbf{N}}^2) \\
&= \tilde{\mathbf{M}}^T \sigma(\hat{\mathbf{N}}^T \bar{\mu}) + \hat{\mathbf{M}}^T \sigma' \tilde{\mathbf{N}}^T \bar{\mu} + w
\end{aligned} \tag{18}$$

where $w = -\tilde{\mathbf{M}} \sigma' \tilde{\mathbf{N}}^T \bar{\mu} - \mathbf{M}^T O(\tilde{\mathbf{N}}^2)$. The form of modeling uncertainty in equation 18 derived using Taylor series expansion of the modeling error will be used in the subsequent Lyapunov analysis of the error dynamics with the postulated weight update laws.

2.3 *Reduced order error observer*

The limit parameter estimate error vector E is required for adaptation of the SHL-NN weights. However, it maybe difficult to obtain accurate measurements for derivatives of the limit parameter. Therefore, an error observer [shown in figure 7] is used to construct an estimate of limit parameter estimate error vector. In the present formulation, a reduced order error observer [30, 31] is used for estimating limit parameter estimate error vector. A reduced order observer is of the form given in equation 19-

$$\begin{aligned}
\dot{\hat{E}} &= \mathbf{A} \hat{E} - \mathbf{K}(\hat{z} - z) \\
\text{where } \hat{z} &= \mathbf{C} \hat{E} \\
z &= \mathbf{C} E
\end{aligned} \tag{19}$$

Also, the matrix gain \mathbf{K} is chosen to place the eigenvalues of matrix $(\mathbf{A} - \mathbf{K}\mathbf{C})$ at desired locations. The error observer in equation 19 is referred to as a reduced order observer because alternate version exists that utilize a full-order error observer with both the error vector E and LDC states η used in NN adaptation [16].

2.4 Single hidden layer weight adaptation

The weight adaptation rules for the single hidden layer neural network are formulated from the Lyapunov analysis of the error dynamics. Lyapunov analysis is done over the limit parameter estimate error dynamics and the error dynamics of the error observer. The limit parameter estimate error dynamics is given in equation 13. The error dynamics for the error observer can be obtained by subtracting equation 13 from equation 19. From the perspective of a RIC architecture [see figure 9], the linear dynamic compensator can provide additionally stability and robustness to the error observer.

$$\dot{\hat{E}} - \dot{E} = \mathbf{A}\hat{E} - \mathbf{K}\mathbf{C}(\hat{E} - E) - \mathbf{A}E - \mathbf{B}_{(r,r)} \left((\nu_{ad}(\bar{\mu}) - \Delta(\bar{\mu})) - \nu_{dc} \right) \quad (20)$$

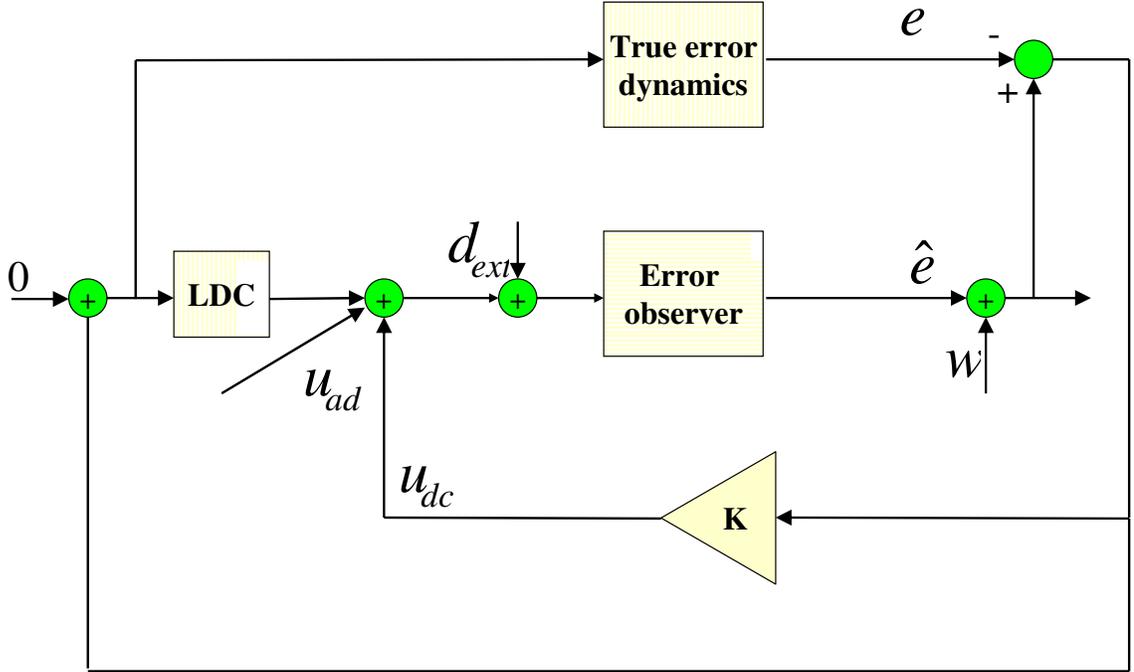


Figure 9: Error observer RIC architecture

Using the definition $\tilde{E} \triangleq \hat{E} - E$ equation 20 can be recast into the following form,

$$\dot{\tilde{E}} = (\mathbf{A} - \mathbf{K}\mathbf{C})\tilde{E} - \mathbf{B}_{(r,r)} \left((\nu_{ad}(\bar{\mu}) - \Delta(\bar{\mu})) - \nu_{dc} \right) \quad (21)$$

Let $tr(\cdot)$ denote the matrix trace operator then, the following Lyapunov function candidate is defined in the vector space $\zeta = \begin{bmatrix} E & \tilde{E} & \tilde{\mathbf{M}} & \tilde{\mathbf{N}} \end{bmatrix}^T$

$$L = E^T \mathbf{P} E + \tilde{E}^T \tilde{\mathbf{P}} \tilde{E} + tr(\tilde{\mathbf{N}}^T \Gamma_n^{-1} \tilde{\mathbf{N}}) + tr(\tilde{\mathbf{M}}^T \Gamma_m^{-1} \tilde{\mathbf{M}}) \quad (22)$$

In the Lyapunov candidate function L , the matrices $\mathbf{P}, \tilde{\mathbf{P}} \in \mathfrak{R}^r \times \mathfrak{R}^r$ are solutions to be the Lyapunov equations 23 and 24, respectively. The matrices $\Gamma_n \in \mathfrak{R}^{N_{hid}} \times \mathfrak{R}^{N_{hid}}$ and $\Gamma_m \in \mathfrak{R}^{N_{out}} \times \mathfrak{R}^{N_{out}}$ are positive definite and therefore non-singular matrices related to SHL-NN weight adaptation.

$$\mathbf{A}^T \mathbf{P} + \mathbf{P} \mathbf{A} + \mathbf{Q} = 0 \quad (23)$$

$$\tilde{\mathbf{A}}^T \tilde{\mathbf{P}} + \tilde{\mathbf{P}} \tilde{\mathbf{A}} + \tilde{\mathbf{Q}} = 0 \quad (24)$$

Also, in equations 23 and 24 $\mathbf{Q} \in \mathfrak{R}^r \times \mathfrak{R}^r$ and $\tilde{\mathbf{Q}} \in \mathfrak{R}^r \times \mathfrak{R}^r$ are arbitrary positive-definite design matrices. According to assumption 2.1.4 the matrices \mathbf{A} and $\tilde{\mathbf{A}}$ has all its eigenvalues in the left-half plane. Therefore, the matrix solutions \mathbf{P} and $\tilde{\mathbf{P}}$ of the Lyapunov equations will be unique as well as positive-definite [29]. Taking the derivative with respect to time of the Lyapunov function candidate we get,

$$\begin{aligned} \dot{L} = & \dot{E}^T \mathbf{P} E + E^T \mathbf{P} \dot{E} + \dot{\tilde{E}}^T \tilde{\mathbf{P}} \tilde{E} + \tilde{E}^T \tilde{\mathbf{P}} \dot{\tilde{E}} + tr(\dot{\tilde{\mathbf{N}}}^T \Gamma_n^{-1} \tilde{\mathbf{N}}) + tr(\tilde{\mathbf{N}}^T \Gamma_n^{-1} \dot{\tilde{\mathbf{N}}}) \\ & + tr(\dot{\tilde{\mathbf{M}}}^T \Gamma_m^{-1} \tilde{\mathbf{M}}) + tr(\tilde{\mathbf{M}}^T \Gamma_m^{-1} \dot{\tilde{\mathbf{M}}}) \end{aligned} \quad (25)$$

Substituting equation 13 and equation 21 into equation 25 we get,

$$\begin{aligned} \dot{L} = & \left(\mathbf{A} E + \mathbf{B}_{(r,r)} \left((\nu_{ad} - \Delta) - \nu_{dc} \right) \right)^T \mathbf{P} E + E^T \mathbf{P} \left(\mathbf{A} E + \mathbf{B}_{(r,r)} \left((\nu_{ad} - \Delta) - \nu_{dc} \right) \right) \\ & + \left(\tilde{\mathbf{A}} \tilde{E} - \mathbf{B}_{(r,r)} \left((\nu_{ad} - \Delta) - \nu_{dc} \right) \right)^T \tilde{\mathbf{P}} \tilde{E} + \tilde{E}^T \tilde{\mathbf{P}} \left(\tilde{\mathbf{A}} \tilde{E} - \mathbf{B}_{(r,r)} \left((\nu_{ad} - \Delta) - \nu_{dc} \right) \right) \\ & + tr(\dot{\tilde{\mathbf{N}}}^T \Gamma_n^{-1} \tilde{\mathbf{N}}) + tr(\tilde{\mathbf{N}}^T \Gamma_n^{-1} \dot{\tilde{\mathbf{N}}}) + tr(\dot{\tilde{\mathbf{M}}}^T \Gamma_m^{-1} \tilde{\mathbf{M}}) + tr(\tilde{\mathbf{M}}^T \Gamma_m^{-1} \dot{\tilde{\mathbf{M}}}) \end{aligned} \quad (26)$$

Grouping similar terms together, equation 26 can be re-cast into the following form-

$$\begin{aligned}\dot{L} &= E^T \left(\mathbf{A}^T \mathbf{P} + \mathbf{P} \mathbf{A} \right) E + \tilde{E}^T \left(\tilde{\mathbf{A}}^T \tilde{\mathbf{P}} + \tilde{\mathbf{P}} \tilde{\mathbf{A}} \right) \tilde{E} + 2E^T \mathbf{P} \mathbf{B}_{(r,r)} \left((\nu_{ad} - \Delta) - \nu_{dc} \right) \\ &\quad - 2\tilde{E}^T \tilde{\mathbf{P}} \mathbf{B}_{(r,r)} \left((\nu_{ad} - \Delta) - \nu_{dc} \right) + tr(\dot{\tilde{\mathbf{N}}}^T \Gamma_n^{-1} \tilde{\mathbf{N}}) + tr(\tilde{\mathbf{N}}^T \Gamma_n^{-1} \dot{\tilde{\mathbf{N}}}) \\ &\quad + tr(\dot{\tilde{\mathbf{M}}}^T \Gamma_m^{-1} \tilde{\mathbf{M}}) + tr(\tilde{\mathbf{M}}^T \Gamma_m^{-1} \dot{\tilde{\mathbf{M}}})\end{aligned}\quad (27)$$

The derivative of the Lyapunov function can be further simplified using equations 23 and 24. Also, the third term in equation 27 can be expanded using the relation $E = \hat{E} - \tilde{E}$,

$$\begin{aligned}\dot{L} &= -E^T \mathbf{Q} E - \tilde{E}^T \tilde{\mathbf{Q}} \tilde{E} + 2\hat{E}^T \mathbf{P} \mathbf{B}_{(r,r)} \left((\nu_{ad} - \Delta) - \nu_{dc} \right) - 4\tilde{E}^T \mathbf{P} \mathbf{B}_{(r,r)} \left((\nu_{ad} - \Delta) - \nu_{dc} \right) \\ &\quad + tr(\dot{\tilde{\mathbf{N}}}^T \Gamma_n^{-1} \tilde{\mathbf{N}}) + tr(\tilde{\mathbf{N}}^T \Gamma_n^{-1} \dot{\tilde{\mathbf{N}}}) + tr(\dot{\tilde{\mathbf{M}}}^T \Gamma_m^{-1} \tilde{\mathbf{M}}) + tr(\tilde{\mathbf{M}}^T \Gamma_m^{-1} \dot{\tilde{\mathbf{M}}})\end{aligned}\quad (28)$$

Proposition 2.4.1. *The SHL-NN are tuned on-line using the following gradient based adaptation laws to approximate/cancel the modeling uncertainty Δ that influences the estimate of limit parameter error dynamics.*

$$\begin{aligned}\dot{\tilde{\mathbf{N}}} &= -\Gamma_n \left[\bar{\mu} \hat{E}^T \mathbf{P} \mathbf{B}_{(r,r)} \hat{\mathbf{M}}^T \sigma' + \kappa_\epsilon \left\| \hat{E}^T \mathbf{P} \mathbf{B}_{(r,r)} \right\| \hat{\mathbf{N}} + \kappa_\sigma \hat{\mathbf{N}} \right] \\ \dot{\tilde{\mathbf{M}}} &= -\Gamma_m \left[\hat{\sigma} \hat{E}^T \mathbf{P} \mathbf{B}_{(r,r)} + \kappa_\epsilon \left\| \hat{E}^T \mathbf{P} \mathbf{B}_{(r,r)} \right\| \hat{\mathbf{M}} + \kappa_\sigma \hat{\mathbf{M}} \right]\end{aligned}\quad (29)$$

Using equation 18 and the weight update laws given in proposition 2.4.1, equation 28 can be re-written into the following form:

$$\begin{aligned}\dot{L} &= -E^T \mathbf{Q} E - \tilde{E}^T \tilde{\mathbf{Q}} \tilde{E} + 2\hat{E}^T \mathbf{P} \mathbf{B}_{(r,r)} \left(\tilde{\mathbf{M}}^T \hat{\sigma} + \hat{\mathbf{M}}^T \sigma' \tilde{\mathbf{N}}^T \bar{\mu} \right) + 2\hat{E}^T \mathbf{P} \mathbf{B}_{(r,r)} (w - \nu_{dc}) \\ &\quad - 4\tilde{E}^T \mathbf{P} \mathbf{B}_{(r,r)} \left(\nu_{ad} - \Delta - \nu_{dc} \right) - 2tr \left(\tilde{\mathbf{N}}^T \left[\bar{\mu} \hat{E}^T \mathbf{P} \mathbf{B}_{(r,r)} \hat{\mathbf{M}}^T \sigma' + \kappa_\epsilon \left\| \hat{E}^T \mathbf{P} \mathbf{B}_{(r,r)} \right\| \hat{\mathbf{N}} \right. \right. \\ &\quad \left. \left. + \kappa_\sigma \hat{\mathbf{N}} \right] \right) - 2tr \left(\tilde{\mathbf{M}}^T \left[\hat{\sigma} \hat{E}^T \mathbf{P} \mathbf{B}_{(r,r)} + \kappa_\epsilon \left\| \hat{E}^T \mathbf{P} \mathbf{B}_{(r,r)} \right\| \hat{\mathbf{M}} + \kappa_\sigma \hat{\mathbf{M}} \right] \right)\end{aligned}\quad (30)$$

In equation 30 the trace identity $tr(A) = tr(A^T)$ has been invoked to get the following relations

$$tr(\dot{\tilde{\mathbf{N}}}^T \Gamma_n^{-1} \tilde{\mathbf{N}}) = tr(\tilde{\mathbf{N}}^T \Gamma_n^{-1} \dot{\tilde{\mathbf{N}}}) \quad (31)$$

$$tr(\dot{\tilde{\mathbf{M}}}^T \Gamma_m^{-1} \tilde{\mathbf{M}}) = tr(\tilde{\mathbf{M}}^T \Gamma_m^{-1} \dot{\tilde{\mathbf{M}}}) \quad (32)$$

also if $a \in \mathfrak{R}$ then,

$$a = \text{tr}(a) \quad (33)$$

Using equation 33 along with the identity $\text{tr}(\mathbf{AB}) = \text{tr}(\mathbf{BA})$ the following trace relations can be derived,

$$\hat{E}^T \mathbf{PB}_{(r,r)} \tilde{\mathbf{M}}^T \hat{\sigma} = \text{tr} \left(\hat{E}^T \mathbf{PB}_{(r,r)} \tilde{\mathbf{M}}^T \hat{\sigma} \right) = \text{tr} \left(\tilde{\mathbf{M}}^T \hat{\sigma} \hat{E}^T \mathbf{PB}_{(r,r)} \right) \quad (34)$$

$$\hat{E}^T \mathbf{PB}_{(r,r)} \hat{\mathbf{M}}^T \sigma' \tilde{\mathbf{N}}^T \bar{\mu} = \text{tr} \left(\hat{E}^T \mathbf{PB}_{(r,r)} \hat{\mathbf{M}}^T \sigma' \tilde{\mathbf{N}}^T \bar{\mu} \right) = \text{tr} \left(\tilde{\mathbf{N}}^T \bar{\mu} \hat{E}^T \mathbf{PB}_{(r,r)} \hat{\mathbf{M}}^T \sigma' \right) \quad (35)$$

The trace relations in equations 34 and 35 can be used to further simplify, equation 30 into equation 36.

$$\begin{aligned} \dot{L} = & -E^T \mathbf{QE} - \tilde{E}^T \tilde{\mathbf{Q}} \tilde{E} + 2\hat{E}^T \mathbf{PB}_{(r,r)}(w - \nu_{dc}) - 4\tilde{E}^T \mathbf{PB}_{(r,r)} \left(\nu_{ad} - \Delta - \nu_{dc} \right) \\ & - 2 \left(\kappa_\epsilon \left\| \hat{E}^T \mathbf{PB}_{(r,r)} \right\| + \kappa_\sigma \right) \left(\text{tr}(\tilde{\mathbf{N}}^T \hat{\mathbf{N}}) + \text{tr}(\tilde{\mathbf{M}}^T \hat{\mathbf{M}}) \right) \end{aligned} \quad (36)$$

As mentioned earlier, the role of the linear dynamic compensator is to compensate for modeling uncertainty which includes correcting for higher order terms. Therefore,

$$\|\nu_{ad} - \Delta - \nu_{dc}\| \leq \alpha_{\hat{E}1} \left\| \tilde{\mathbf{Z}} \right\|_F + \alpha_{\hat{E}2} \quad (37)$$

$$\|\nu_{dc} - w\| \leq \alpha_{\hat{E}1} \left\| \tilde{\mathbf{Z}} \right\|_F + \alpha_{\hat{E}2} \quad (38)$$

where,

$$\mathbf{Z} \triangleq \begin{bmatrix} \mathbf{M} & 0 \\ 0 & \mathbf{N} \end{bmatrix}, \quad \hat{\mathbf{Z}} \triangleq \begin{bmatrix} \hat{\mathbf{M}} & 0 \\ 0 & \hat{\mathbf{N}} \end{bmatrix}, \quad \tilde{\mathbf{Z}} \triangleq \hat{\mathbf{Z}} - \mathbf{Z} \quad (39)$$

$$\|\nu_{ad} - \Delta\| \leq \alpha_1 \left\| \tilde{\mathbf{Z}} \right\|_F + \alpha_2 \quad (40)$$

Also,

$$\begin{aligned} -2\text{tr}(\tilde{\mathbf{N}}^T \hat{\mathbf{N}}) - 2\text{tr}(\tilde{\mathbf{M}}^T \hat{\mathbf{M}}) &= -2\text{tr} \left(\tilde{\mathbf{Z}}^T \hat{\mathbf{Z}} \right) \\ &= -2\text{tr}(\tilde{\mathbf{Z}}^T \tilde{\mathbf{Z}}) - 2\text{tr}(\tilde{\mathbf{Z}}^T \mathbf{Z}) \\ &\leq -2 \left\| \tilde{\mathbf{Z}} \right\|_F^2 + 2 \left\| \tilde{\mathbf{Z}} \right\|_F \|\mathbf{Z}\|_F \\ &\leq -2 \left\| \tilde{\mathbf{Z}} \right\|_F^2 + \left\| \tilde{\mathbf{Z}} \right\|_F^2 + \|\mathbf{Z}\|_F^2 \end{aligned} \quad (41)$$

$$\therefore -2tr(\tilde{\mathbf{N}}^T \hat{\mathbf{N}}) - 2tr(\tilde{\mathbf{M}}^T \hat{\mathbf{M}}) \leq -\left\|\tilde{\mathbf{Z}}\right\|_F^2 + Z^* \quad (42)$$

In deriving the trace relation in equation 42 the following identities were used,

$$|tr(\mathbf{A}^T \mathbf{B})| \leq \|\mathbf{A}\|_F \|\mathbf{B}\|_F \quad (43)$$

$$+2ab \leq a^2 + b^2 \quad (44)$$

$$\|\mathbf{Z}\|_F^2 = Z^* \quad (45)$$

Using equations 37,38 and 41 the derivative of Lyapunov function in equation 36 can be upper bounded as,

$$\begin{aligned} \dot{L} &\leq -\lambda_{min}(\mathbf{Q}) \|E\|^2 - \lambda_{min}(\tilde{\mathbf{Q}}) \|\tilde{E}\|^2 + 2 \|\hat{E}^T \mathbf{P} \mathbf{B}_{(r,r)}\| \left(\alpha_{\hat{E}1} \|\tilde{\mathbf{Z}}\|_F + \alpha_{\hat{E}2} \right) \\ &\quad + 4 \|\tilde{E}^T \mathbf{P} \mathbf{B}_{(r,r)}\| \left(\alpha_{\tilde{E}1} \|\tilde{\mathbf{Z}}\|_F + \alpha_{\tilde{E}2} \right) + \left(\kappa_\epsilon \|\hat{E}^T \mathbf{P} \mathbf{B}_{(r,r)}\| + \kappa_\sigma \right) \left(Z^* - \|\tilde{\mathbf{Z}}\|_F^2 \right) \end{aligned} \quad (46)$$

$$\begin{aligned} &\leq -\lambda_{min}(\mathbf{Q}) \|E\|^2 - \lambda_{min}(\tilde{\mathbf{Q}}) \|\tilde{E}\|^2 + 2 \|\hat{E}\| \left(\bar{\alpha}_{\hat{E}1} \|\tilde{\mathbf{Z}}\|_F + \bar{\alpha}_{\hat{E}2} \right) \\ &\quad + 4 \|\tilde{E}\| \left(\bar{\alpha}_{\tilde{E}1} \|\tilde{\mathbf{Z}}\|_F + \bar{\alpha}_{\tilde{E}2} \right) + \left(\bar{\kappa}_\epsilon \|\hat{E}\| + \kappa_\sigma \right) \left(Z^* - \|\tilde{\mathbf{Z}}\|_F^2 \right) \end{aligned} \quad (47)$$

where $\bar{\alpha}_{\hat{E}i} = \|\mathbf{P} \mathbf{B}_{(r,r)}\| \alpha_{\hat{E}i}$ and $\bar{\kappa}_\epsilon = \kappa_\epsilon \|\mathbf{P} \mathbf{B}_{(r,r)}\|$. Now using the property $2xy \leq x^2 + y^2$ equation 47 can be re-written into the following form-

$$\begin{aligned} \dot{L} &\leq -\lambda_{min}(\mathbf{Q}) \|E\|^2 - \lambda_{min}(\tilde{\mathbf{Q}}) \|\tilde{E}\|^2 + \|\hat{E}\| \left(1 + \bar{\alpha}_{\hat{E}1} \|\tilde{\mathbf{Z}}\|_F^2 \right) \\ &\quad + \|\tilde{E}\|^2 + \bar{\alpha}_{\tilde{E}2}^2 + 4 \|\tilde{E}\|^2 + 2\bar{\alpha}_{\tilde{E}1}^2 \|\tilde{\mathbf{Z}}\|_F^2 + 2\bar{\alpha}_{\tilde{E}2}^2 \\ &\quad + \bar{\kappa}_\epsilon \|\hat{E}\| Z^* - \bar{\kappa}_\epsilon \|\hat{E}\| \|\tilde{\mathbf{Z}}\|_F^2 - \kappa_\sigma \|\tilde{\mathbf{Z}}\|_F^2 + \kappa_\sigma Z^* \end{aligned} \quad (48)$$

Collecting common terms and re-arranging equation 48 results in the following equation:

$$\begin{aligned} \dot{L} &\leq -\lambda_{min}(\mathbf{Q}) \|E\|^2 - \left(\lambda_{min}(\tilde{\mathbf{Q}}) - 4 \right) \|\tilde{E}\|^2 + \|E\| + \|\tilde{E}\| \\ &\quad + \left(\|E\| + \|\tilde{E}\| \right)^2 + \bar{\alpha}_{\tilde{E}2}^2 + 2\bar{\alpha}_{\tilde{E}1}^2 + \bar{\kappa}_\epsilon \left(\|E\| + \|\tilde{E}\| \right) Z^* \\ &\quad - \left(\bar{\kappa}_\epsilon - \bar{\alpha}_{\hat{E}1} \right) \|\hat{E}\| \|\tilde{\mathbf{Z}}\|_F^2 - \left(\kappa_\sigma - 2\bar{\alpha}_{\tilde{E}1}^2 \right) \|\tilde{\mathbf{Z}}\|_F^2 + \kappa_\sigma Z^* \end{aligned} \quad (49)$$

Choose $\bar{\kappa}_\epsilon \geq \bar{\alpha}_{\tilde{E}1}$ and let $\bar{q} = \min\left\{\lambda_{\min}(\mathbf{Q}), \lambda_{\min}(\tilde{\mathbf{Q}}) - 4\right\}$ then the RHS of equation 49 can be upper bounded as follows:

$$\begin{aligned} \dot{L} \leq & -(\bar{q} - 1)\left(\|E\| + \|\tilde{E}\|\right)^2 + \left(1 + \bar{\kappa}_\epsilon Z^*\right)\left(\|E\| + \|\tilde{E}\|\right) \\ & + \bar{\alpha}_{\tilde{E}2}^2 + 2\bar{\alpha}_{\tilde{E}2}^2 - \left(\kappa_\sigma - 2\bar{\alpha}_{\tilde{E}1}^2\right)\|\tilde{\mathbf{Z}}\|_F^2 + \kappa_\sigma Z^* \end{aligned} \quad (50)$$

Let $\beta^2 = (\bar{q} - 1)$ and completing squares for $\|E\| + \|\tilde{E}\|$ we get,

$$\dot{L} \leq -\left[\beta\left(\|E\| + \|\tilde{E}\|\right) - \Gamma_1\right]^2 - \left(\kappa_\sigma - 2\bar{\alpha}_{\tilde{E}1}^2\right)\|\tilde{\mathbf{Z}}\|_F^2 + \Gamma_2 \quad (51)$$

where $\Gamma_1 = \frac{1 + \bar{\kappa}_\epsilon Z^*}{2\beta}$ and $\Gamma_2^2 = \bar{\alpha}_{\tilde{E}2}^2 + 2\bar{\alpha}_{\tilde{E}2}^2 + \kappa_\sigma Z^* + \left(\frac{1 + \bar{\kappa}_\epsilon Z^*}{2\beta}\right)^2$. Therefore, if $\beta > 0$ and $\kappa_\sigma > 2\bar{\alpha}_{\tilde{E}1}^2$ then the time derivative of the Lyapunov function will be negative definite whenever *either* one of the following conditions are satisfied-

$$\|E\| > \frac{\Gamma_1 + \Gamma_2}{\beta} \quad (52)$$

$$\|\tilde{E}\| > \frac{\Gamma_1 + \Gamma_2}{\beta} \quad (53)$$

$$\|\tilde{\mathbf{Z}}\|_F > \frac{\Gamma_2}{\kappa_\sigma - 2\bar{\alpha}_{\tilde{E}1}^2} \quad (54)$$

Equations 52, 53 and 54 together define a bounded set in the error vector space ζ .

2.5 Simulation results

To illustrate the application of adaptive estimate of limit parameter dynamics two different cases are presented. In the first example shown, an adaptive estimate, starting from two different approximate models, is constructed for a linear system with third order limit parameter dynamics. In the second example, the limit parameter is an output variable of a nonlinear plant controlled by a nonlinear adaptive controller. The plant is the classic van der Pol oscillator system coupled to a lightly damped mode [19].

2.5.1 Linear plant

The true limit parameter dynamics, as given in equation 55, is a third order linear system with eigenvalues -1,-2,-3.

$$\ddot{y} + 6\dot{y} + 11y + 6y = 6u. \quad (55)$$

2.5.1.1 Approximate model-1

The linear approximate model, given in equation 56 is chosen to have eigenvalues as $\{-1, -2, -2.5\}$ with uncertainty only in the fastest mode.

$$\ddot{y}_m + 5.5\dot{y}_m + 9.5y_m + 5y_m = 4u. \quad (56)$$

For the adaptive estimation process the LDC was designed using LQG design techniques. The dynamic compensator design matrices obtained using this method are:

$$A_c = \begin{bmatrix} -2.0334 & 1.0000 & 0 \\ -2.0674 & 0 & 1.0000 \\ -67.2142 & -50.3522 & -10.5809 \end{bmatrix}, \quad B_c = \begin{bmatrix} 2.0334 \\ 2.0674 \\ -3.6731 \end{bmatrix}$$

$$C_c = \begin{bmatrix} 65.8872 & 40.8522 & 5.0809 \end{bmatrix}, \quad D_c = \begin{bmatrix} 0 \end{bmatrix}$$

Neural network implemented for this case has 10 hidden neurons and 4 input neurons. It is assumed that no measurements of limit parameter time derivatives are available. To compensate for the lack of limit parameter derivative information, the neural network input vector is chosen to contain delayed values of limit parameter measurement.

$$\mu = \begin{bmatrix} y(t) & y(t - t_d) & y(t - 2t_d) & u(t) \end{bmatrix}$$

The adaptation gains are chosen to be $\Gamma_m = 5I$, $\Gamma_n = 2I$ and $\kappa_\epsilon = 0.7$. Also, the matrix \mathbf{K} used in the reduced order error observer is selected to be $K^T = \begin{bmatrix} 22 & 107 & -177.5 \end{bmatrix}$

Figures 10, 11 and 12 present the results for this case. Figure 10 compares the actual limit parameter response to the model response when adaptation is off. The network output is also plotted along with the modeling error in the same figure. Without adaptation the network output is zero, and hence, the approximate model is unable to match the true response.

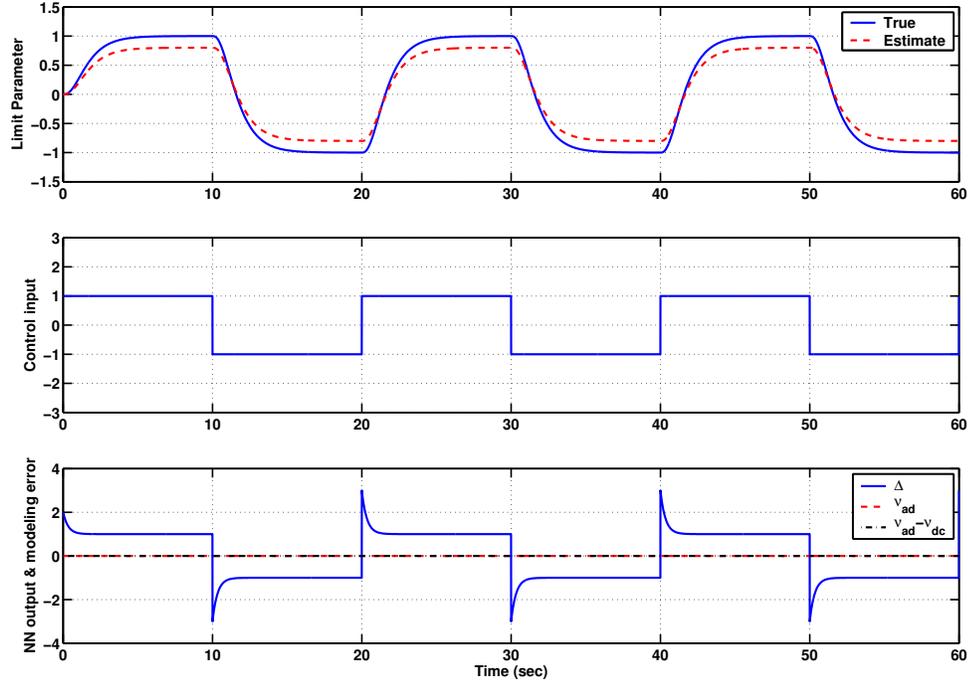


Figure 10: Linear plant- Response comparison with adaptation OFF-*Approximate model-1*

With adaptation on, the results are shown in figure 11. Figure 12 presents the variation in neural network weights during adaptation. A comparison of figures 10 and 11 shows a significant improvement in the estimated response when the adaptation is switched on. Due to the presence of LDC and the NN that is continuously adapting to the modeling error, the model response is seen to match the true limit parameter response quite well.

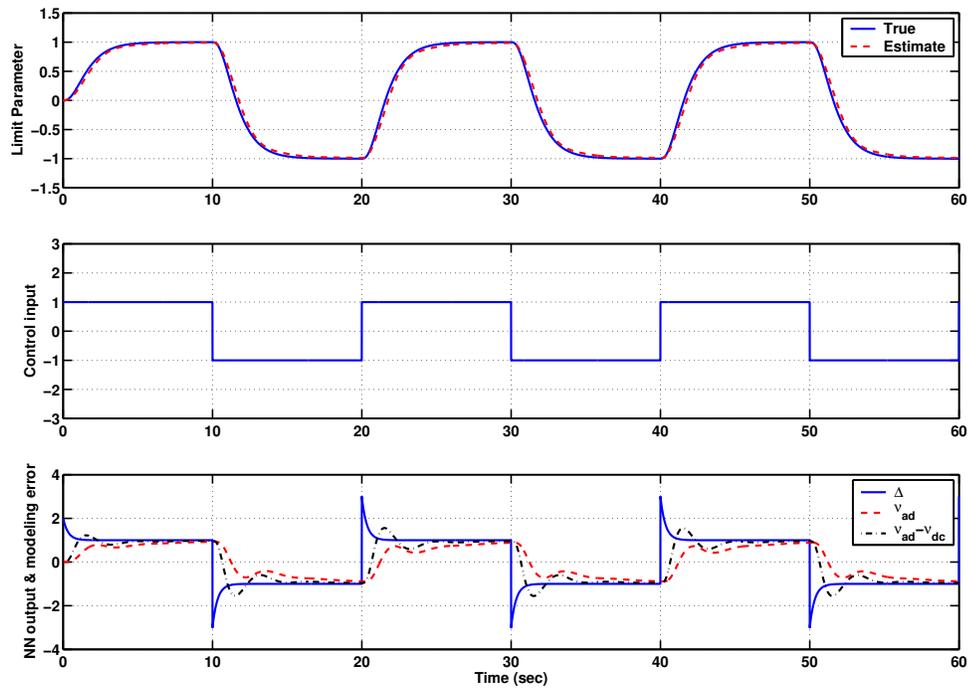


Figure 11: Linear plant- Response comparison with adaptation ON-*Approximate model-1*

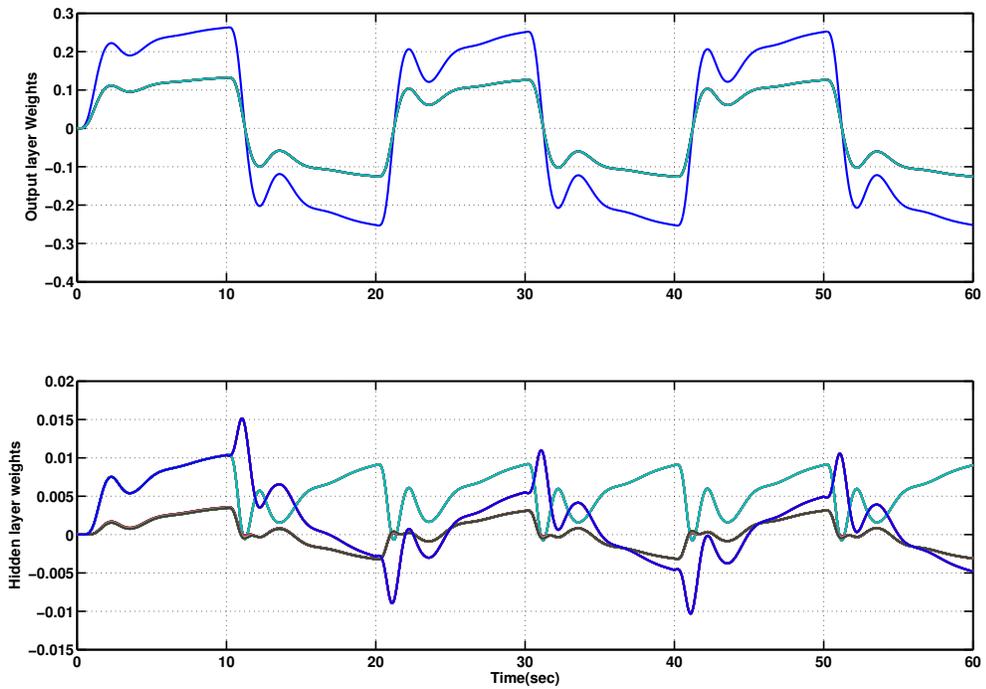


Figure 12: Linear plant- Variation in neural network weights during adaptation -*Approximate model-1*

2.5.1.2 Approximate model-2

The linear approximate model obtained from choosing eigenvalues as -1.2,-1.3,-3 is given in equation 57. Note that in this case the modeling uncertainty is in the first two modes.

$$\ddot{y}_m + 6.5\dot{y}_m + 13.26y_m + 8.28y_m = 4u \quad (57)$$

For the same control input as the previous case the model response with adaptation off is compared to the actual limit parameter response in figure 13. Notice that this choice of approximate linear model results in larger limit parameter estimate error than in the previous case. To correct for these modeling errors, the adaptation process begins with the design of the LDC. Again LQG design technique is used which produces the following matrices for the compensator:

$$A_c = \begin{bmatrix} -1.5270 & 1.0000 & 0 \\ -1.1658 & 0 & 1.0000 \\ -67.7776 & -51.1364 & -10.8629 \end{bmatrix}, \quad B_c = \begin{bmatrix} 1.5270 \\ 1.1658 \\ -3.4162 \end{bmatrix}$$

$$C_c = \begin{bmatrix} 62.9138 & 37.8764 & 4.3629 \end{bmatrix}, \quad D_c = \begin{bmatrix} 0 \end{bmatrix}$$

The NN input vector and structure is kept the same but the adaptation gains are chosen as, $\Gamma_m = 10I, \Gamma_n = 5I$ and $\kappa_\epsilon = 1$. The gain matrix for the reduced order error observer is chosen to be $\mathbf{K}^T = \begin{bmatrix} 26.0 & 149.24 & -288.1 \end{bmatrix}$. Figures 14 and 15 present results when adaptation is on. Notice from figure 14 that the model response shows significant improvement with adaptation. The NN output tries to approximate the modeling uncertainty, which is shown in figure 14. In computing the matrix \mathbf{P} , the matrix \mathbf{Q} is selected to be $5I$ in case of approximate model I and $6I$ in case of approximate model 2.

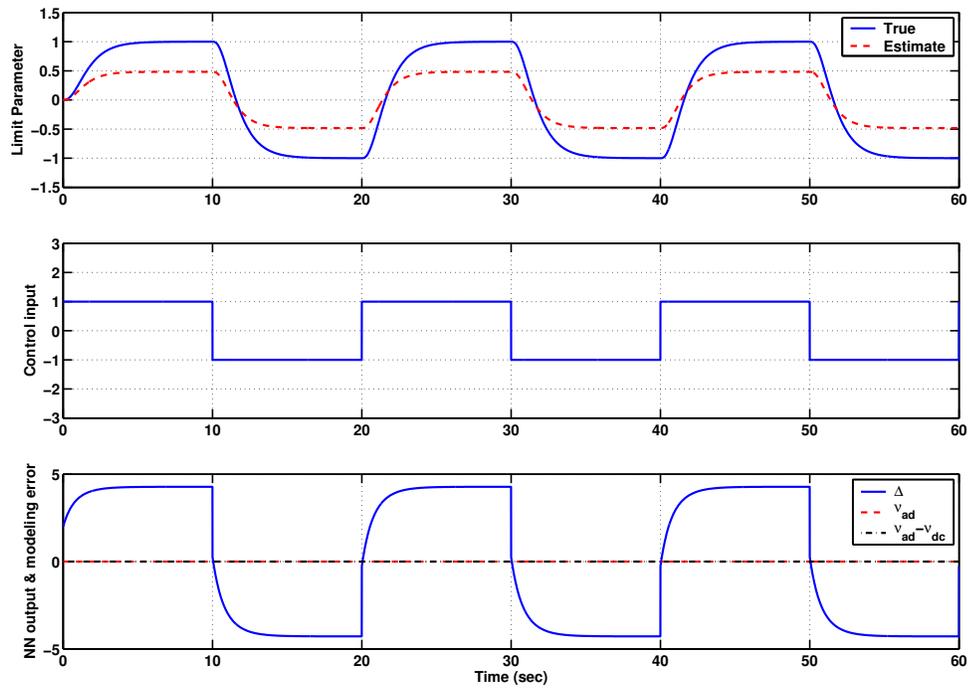


Figure 13: Linear plant- Response comparison with adaptation OFF-*Approximate model-2*

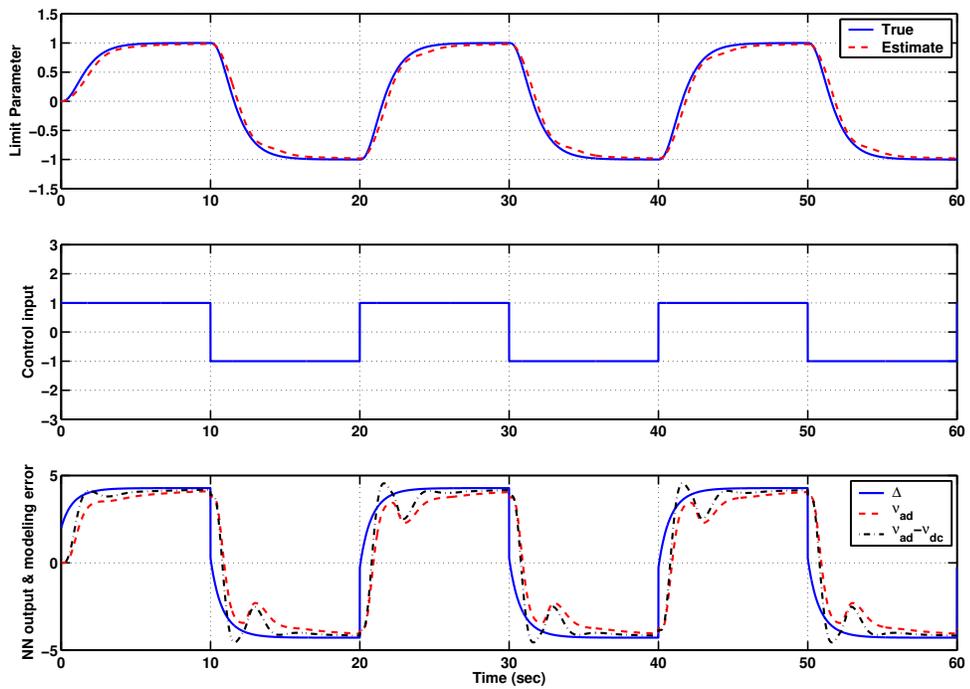


Figure 14: Linear plant- Response comparison with adaptation ON-*Approximate model-2*

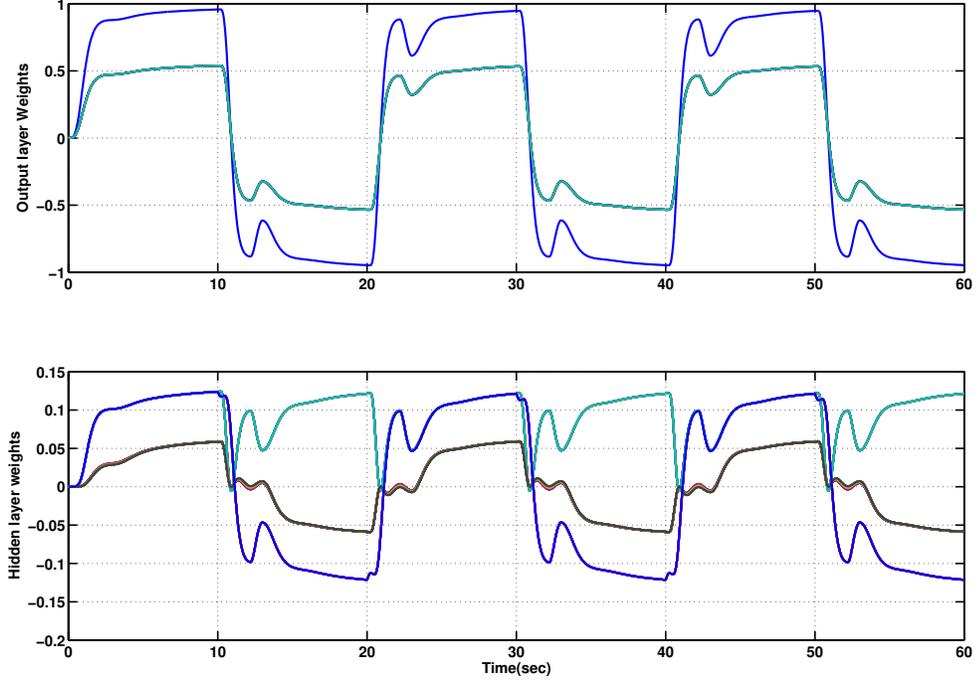


Figure 15: Linear plant- Variation in neural network weights during adaptation- *Approximate model-2*

2.5.2 Van der Pol oscillator

A block diagram representation of the nonlinear controller-plant system is shown in figure 16. The input to this system is the tracking command y_c . The controller used is a nonlinear adaptive NN based controller, details about which are provided in [19]. The plant, Van der Pol oscillator, is modeled by the following equations:

$$\begin{aligned}
 \dot{x}_1 &= x_2, \\
 \dot{x}_2 &= -\alpha(x_1^2 - 1)x_2 - x_1 + u, \\
 \dot{x}_3 &= x_4, \\
 \dot{x}_4 &= -x_3 - 0.2x_4 + x_1.
 \end{aligned} \tag{58}$$

The control input u is generated by a first order actuator with dynamics as given below.

$$\dot{u} = \frac{1}{\tau_a}(u_c - u), \tag{59}$$

with a time constant $\tau_a = 0.1$ seconds. The “damping” parameter α is set to 0.8. The output of the plant (y), given by equation 60, is coupled with the internal dynamic states of the system x_3 and x_4 .

$$y = x_1 + \varepsilon x_3, \quad (60)$$

These states are not directly influenced by the control input u but are driven by the Van der Pol oscillator state x_1 . The coupling parameter ε in equation 60 is set to 0.2.

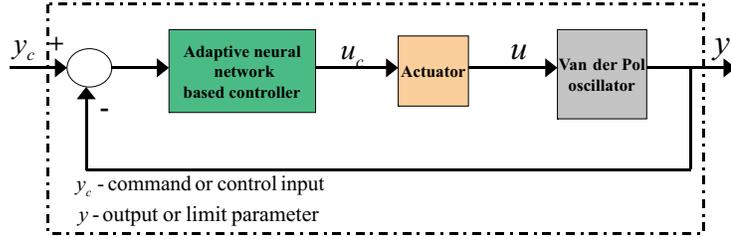


Figure 16: Block diagram representation of the nonlinear controller Van der Pol oscillator system

The approximate model is chosen to be

$$\ddot{y}_m + 10\dot{y}_m + 17y_m + 8y_m = 5y_c \quad (61)$$

with eigenvalues $\{-8, -1, -1\}$. Repeated eigenvalue -1 corresponds to approximation of the van der pol oscillator dynamics and eigenvalue -8 represents actuator dynamics.

In figure 17 this approximate linear model response is plotted along with the true response of the nonlinear system. Notice that without adaptation the responses do not match. The adaptation process begins with augmentation of the approximate model with a LDC and a SHL NN. The LDC is designed using LQG design technique and produces the following matrices:

$$A_c = \begin{bmatrix} -1.1603 & 1.0000 & 0 \\ -0.6731 & 0 & 1.0000 \\ -98.48 & -62.46 & -13.82 \end{bmatrix}, \quad B_c = \begin{bmatrix} 1.1603 \\ 0.6731 \\ -1.8367 \end{bmatrix}$$

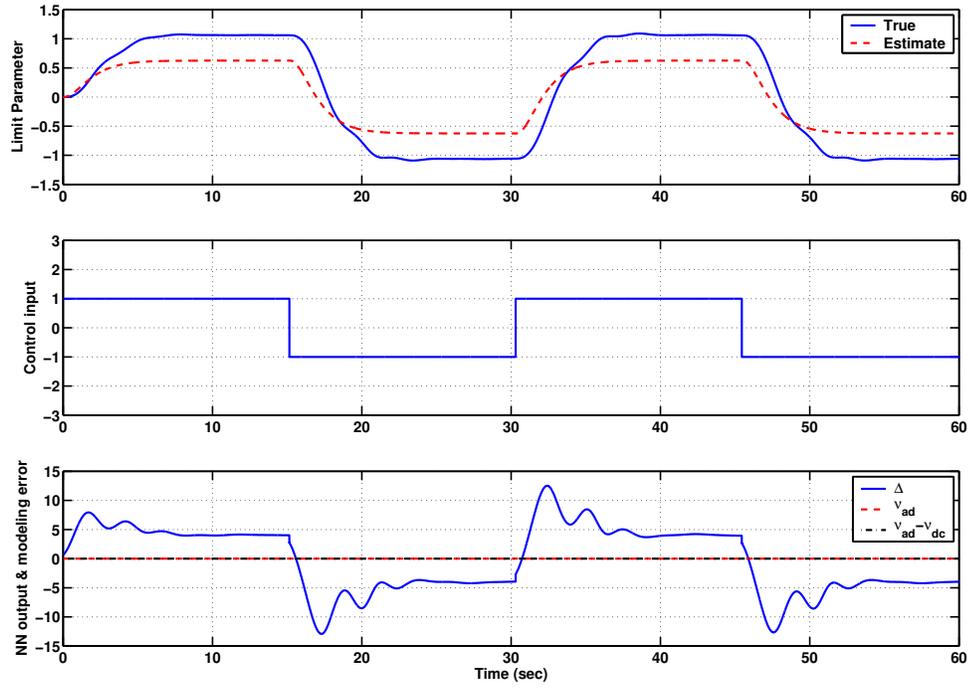


Figure 17: Van der Pol oscillator limit estimation with adaptation OFF

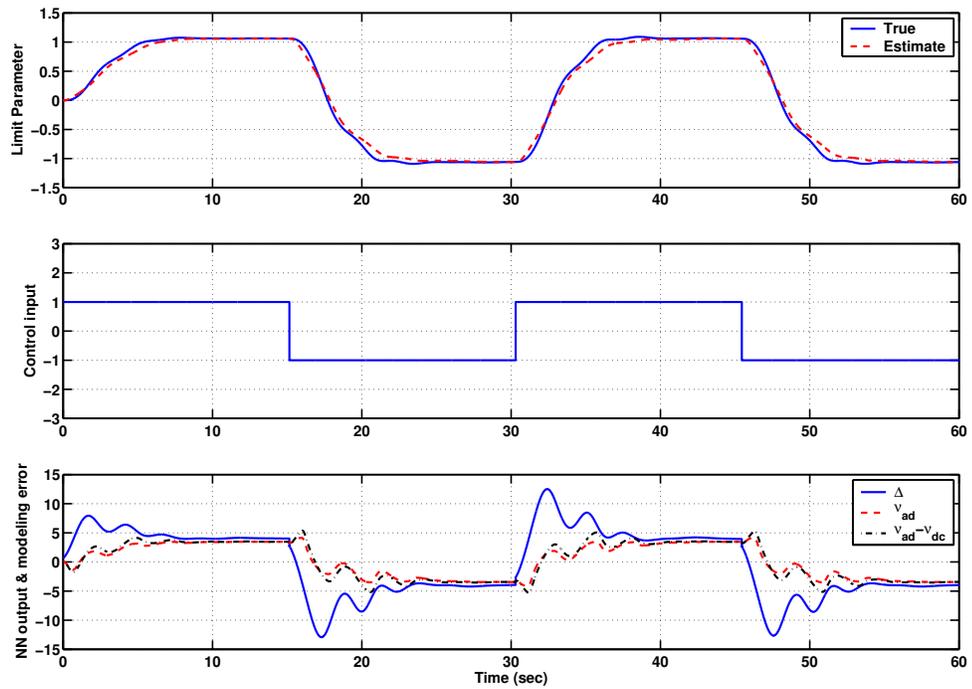


Figure 18: Van der Pol oscillator limit estimation with adaptation ON

$$C_c = \begin{bmatrix} 92.32 & 45.46 & 3.18 \end{bmatrix}, \quad D_c = \begin{bmatrix} 0 \end{bmatrix}$$

The neural network is once again chosen to have four input neurons and ten hidden neurons with adaptation gains $\mathbf{\Gamma}_m = 12I, \mathbf{\Gamma}_n = 5I$ and $\kappa_\epsilon = 1.0$.

The response of adaptive estimate of limit parameter dynamics is compared against the true response in figure 18. Observe that with adaptive augmentation the estimate of the limit parameter comes significantly closer to the true response. This improvement is a result of the adaptive neural network and LDC trying to minimize the modeling uncertainty present in the estimate of limit parameter dynamics.

CHAPTER III

OPTIMAL CONTROL FORMULATION AND NONLINEAR TRAJECTORY GENERATION FOR ENVELOPE PROTECTION

Envelope protection methods such as dynamic trim [12], nonlinear function response method [15] and peak-estimation method [54] are based on identifying the functional relationship to calculate the maximum future response of the limit parameter for a given control input. In case of steady-state response critical limit parameters (limit parameters that attain their maximum absolute step-response value in the steady-state) the future dynamic trim response is also the maximum value that the limit parameter response can attain. The functional form of the future dynamic trim response value is then used to estimate the control limit, identified as the input that results in the dynamic trim response of the limit parameter at the limit boundary. Adaptive dynamic trim envelope protection method [68] uses adaptive estimate of limit parameter dynamics and dynamic trim approach. Adaptive dynamic trim based envelope protection method has been successfully evaluated on both piloted [60] and uninhabited aerial vehicle platforms [70]. However, this functional form as well as the concept of dynamic trim are only applicable to steady-state response critical limit parameters and cannot be used to estimate control limits for transient-response critical limit parameters (limit parameters that attain their maximum absolute step-response value during the transient phase). Currently, no known extensions of the dynamic trim based method exist that make it applicable to transient response critical limit parameters.

Motivated by the success of adaptive NN based techniques for on-line estimation of limit parameter dynamics, a new approach is proposed for estimating control limits. This new envelope protection method is referred to as the Nonlinear Trajectory Generator (NTG) approach and is based on determining an optimal control profile that takes the limit parameter response to the limit boundary. The optimal control profile minimizes an objective function which is a sum of both time and control effort. This aggressive control profile is then used to estimate the control limits corresponding to the limit boundary. Unlike approaches such as dynamic trim, the NTG approach is not based on developing the functional form to represent maximum step response of the limit parameter and therefore can be used for both steady-state as well as transient response critical limit protection. Additionally, the approach is developed for application within piloted vehicles, however, the estimated control limits can also be used for automatic envelope protection [70] in unmanned autonomous systems.

3.1 Formulation of optimal control problem

Assuming the adaptive estimate of limit parameter dynamics is a good approximation of the true limit parameter dynamics, i.e,

$$\|e(t)\| = \|\hat{y}_p(t) - y_p(t)\| \leq \epsilon \quad (62)$$

It is proposed that the control limits can be computed by finding the optimal control that *minimizes* the objective function in equation 63 for the adaptive estimate of limit parameter dynamics.

$$J = \int_{t_0}^{t_f} (1 + 0.5Wu_p^2) \quad (63)$$

In equation 63, $W > 0$ is a design constant referred to as control weighting. *Terminal constraints* of the optimal control problem are:

$$\hat{y}_p(t_f) = y_{lim} \quad (64)$$

where the final time t_f is *free* and y_{lim} refers to the value of limit boundary (upper or lower). The true limit parameter dynamics, with known relative degree r can be represented by the following equation,

$$y_p^{(r)} = h_r(\mathbf{x}, y_p, y_p^{(1)}, \dots, y_p^{(r-1)}, u_p). \quad (65)$$

The general functional form of the adaptive estimate of limit parameter dynamics is given as:

$$\hat{y}_p^{(r)} = \sum_{i=0}^{r-1} a_i \hat{y}_p^{(i)} + \frac{\partial \hat{h}_r}{\partial u_p} u_p + \nu_{ad}(\bar{\mu}) - \nu_{dc} \quad (66)$$

The problem of estimating the control limits consist of two parts:

1. Solving for the optimal control in real-time.
2. Estimating the control limit from the optimal control solution.

The objective function in equation 63 for this optimal control problem is not chosen arbitrarily but is based on certain assumptions. Most limit boundary violations occur when pilot is executing the mission aggressively which translates into attempting to complete the task in shortest possible time. The first term in the objective function represents this aggressiveness factor (minimize t_f). If the objective function is only to minimize time, i.e, control weighting is zero, then the optimal solution will be infinite control. The second term however, weighs the control contribution and, therefore, the optimal control solution is guaranteed to take the limit parameter response from its present value to the limit boundary with an acceptable measure of aggressiveness. This acceptable measure is then imposed on the pilot by transforming the optimal control solution into an equivalent control limit value.

Let u_p^* denote the optimal control solution. In order to compute control limits from the optimal control solution, a new functional called as the *area norm* is defined. The area norm of a control signal $u_p(t)$ between the time interval t_0 to t_f can be computed

as:

$$u_{AN}\left(u_p(\cdot), [t_0 \quad t_f]\right) \triangleq \sqrt{\frac{\int_{t_0}^{t_f} u_p^2}{t_f - t_0}}. \quad (67)$$

The area norm of the optimal control solution u_p^* denoted as u_{AN}^* is then given by equation 68.

$$u_{AN}^*(t_0) \triangleq u_{AN}\left(u_p^*(\cdot), [t_0 \quad t_f^*]\right) = \sqrt{\frac{\int_{t_0}^{t_f^*} u_p^{*2}}{t_f^* - t_0}} \quad (68)$$

where t_f^* denotes the final time obtained from the optimal solution.

Proposition 3.1.1. *The area norm of the optimal control solution $u_{AN}^*(t_0)$ is an estimate of the control limit at time t_0 .*

Let \mathcal{U} be a set representing all step inputs at time t_0 . Then, any element $\bar{u}_p(t) \in \mathcal{U}$ will be of the following form-

$$\bar{u}_p(t) = \begin{cases} u_p(t) & t \leq t_0 \\ k & t > t_0 \end{cases} \quad (69)$$

where $k \in \mathfrak{R}$. The following analysis will prove that any element of this set that can take the system to the limit value in shorter time $\bar{t}^* < t_f^*$ will have area norm greater than u_{AN}^* . Assume $\bar{u}_p^*(t) \in \mathcal{U}$ to be such a control input that takes the adaptive estimate of the limit parameter dynamics from its present value $\hat{y}_p(t_0)$ to the limit boundary y_{lim} in time \bar{t}^* . Let \bar{u}_p^* be given as in equation 70.

$$\bar{u}_p^*(t) = \begin{cases} u_p(t) & t \leq t_0 \\ k^* & t > t_0 \end{cases} \quad (70)$$

Then the area norm of $\bar{u}_p^*(t)$ is k^* . According to the principle of optimality, the cost of the optimal solution is always the smallest. Hence,

$$\int_{t_0}^{\bar{t}^*} (1 + 0.5W\bar{u}_p^*.\bar{u}_p^*) \geq \int_{t_0}^{t_f^*} (1 + 0.5Wu_p^*.u_p^*) \quad (71)$$

Since $\bar{t}^* < t_f^*$, the following inequality can be deduced from equation 71,

$$\int_{t_0}^{t_f^*} (1 + 0.5W\bar{u}_p^*.\bar{u}_p^*) > \int_{t_0}^{t_f^*} (1 + 0.5Wu_p^*.u_p^*) \quad (72)$$

Using the definition of area norm in equation 67, equation 72 can be re-written as

$$\begin{aligned} (t_f^* - t_0) + 0.5W(t_f^* - t_0)\bar{k}^*.\bar{k}^* > \\ (t_f^* - t_0) + 0.5W(t_f^* - t_0)u_{AN}^*(t_0).u_{AN}^*(t_0) \end{aligned} \quad (73)$$

When $(t_f^* - t_0) > 0$, equation 73 can be reduced to the following identity,

$$\implies \|\bar{k}^*\| > \|u_{AN}^*(t_0)\| \quad (74)$$

The inequality in equation 74 is used to conclude the following proposition-

Proposition 3.1.2. *The absolute value of any step control input that takes the limit parameter response to the limit boundary in shorter time than the optimal final time, will be larger than the absolute value of the area norm of the optimal control solution (u_{AN}^*).*

Therefore, the area norm of the optimal control solution can be used as an estimate of the control limit for envelope protection. However, as the limit parameter response approaches the limit boundary t_f^* will approach t_0 , i.e.,

$$\Delta t_{crit} \triangleq t_f^* - t_0 \rightarrow 0 \quad \text{as} \quad \hat{y}_p \rightarrow y_{lim} \quad (75)$$

and therefore, the area norm calculation in equation 68 will become numerically ill-defined. This problem is avoided by choice of a threshold time value Δt_h and a smoothing function $S(\cdot)$ based on Δt_{crit} . The control limit is calculated using the area norm of the optimal control profile and the smoothing function as given in equation 76.

$$u_{lim}(t_0) = u_p(t_0) + \left(u_{AN}^*(t_0) - u_p(t_0) \right) S(\Delta t_{crit}) \quad (76)$$

The smoothing function is designed to be such that the control limit approaches the current control input value smoothly as $\Delta t_{crit} \rightarrow 0$.

$$S(\Delta t_{crit}) = \begin{cases} 1 & \Delta t_{crit} \geq \Delta t_h \\ \text{Any smooth function} & \Delta t_{crit} < \Delta t_h \end{cases} \quad (77)$$

3.2 Control limit as the area norm

In the previous section it is shown that the area norm of the control input that minimizes the objective function given in equation 63 will be smaller in value than any step control input that would take the limit parameter response to the limit boundary faster than the optimal control. The main reasoning behind comparing the area norm against all possible step-control inputs arises from the traditional definition of control limit. According to the traditional definition, control limit is defined as the value of step control input that results in maximum value of the limit parameter response to be at the limit boundary. Let us denote this control limit as $\bar{u}_{lim} \in \mathcal{U}$.

Proposition 3.2.1. *Among all other step control inputs that take the limit parameter response to the limit boundary the control limit (\bar{u}_{lim}) has the smallest area norm.*

Proposition 3.2.1 is a natural conclusion of the traditional definition of control limit. A control input smaller than the control limit will not cause the limit parameter response to reach the limit boundary. On the other hand any step-control input larger than the control limit value will cause the limit parameter response to reach the limit boundary (even in shorter time) but will have a larger area norm.

Proposition 3.2.2. *Among all step control inputs, \bar{u}_{lim} is the step control input that will take the limit parameter response from its present value to the limit boundary while minimizing the objective function given in equation 63.*

This proposition can be proved by combining proposition 3.1.2 and 3.2.1. According to proposition 3.1.2 the area norm of the optimal control solution is the

smallest among all other step-control inputs that takes the limit parameter response to the limit boundary while minimizing the objective function in equation 63. Now according to proposition 3.2.1 the control limit is this smallest step control input. Therefore, if only step-control inputs are allowed in the minimization problem then the area norm of the optimal solution will be the traditional control limit.

In the proposed approach however, a more general class of control inputs is treated and the definition of control limit is relaxed to consider not just step inputs but also other arbitrary control profiles. Since many of the existing envelope protection applications require a control limit value rather than control limiting profile, the area norm functional is utilized as a mapping tool to generate a control limit estimate from the optimal control solution. The following proposition states a special case under which the area norm of optimal control solution (arbitrary control profile) will be equal to the traditional control limit value, i.e, when $u_{AN}^*(t_0)$ will be equal to \bar{u}_{lim} .

Proposition 3.2.3. *When the optimal final time to reach the limit boundary (t_f^*) is equal to the peak time (time for step-response to reach maximum value) then the area norm of the optimal control solution will be equal to control limit in the traditional sense.*

Proposition 3.2.3 can be proved as follows- Let us assume that for some choice of control weighting,

$$\begin{aligned}
u_{AN}^*(t_0) &= \bar{u}_{lim} & (78) \\
\implies \frac{\int_{t_0}^{t_f^*} u^{*2} dt}{t_f^* - t_0} &= \frac{\int_{t_0}^{t_f^*} \bar{u}_{lim}^2 dt}{t_f^* - t_0} \\
\implies W \int_{t_0}^{t_f^*} u^{*2} dt &= W \int_{t_0}^{t_f^*} \bar{u}_{lim}^2 dt
\end{aligned}$$

$$\begin{aligned}
\therefore J_{min} &= t_f^* + W \int_{t_0}^{t_f^*} u^{*2} dt = t_f^* + W \int_{t_0}^{t_f^*} \bar{u}_{lim}^2 dt \\
&= t_p + W \int_{t_0}^{t_p} \bar{u}_{lim}^2 dt & (79)
\end{aligned}$$

when t_p , time to peak is equal to t_f^* . Therefore, the control limit estimated from the optimal control solution will be equal to \bar{u}_{lim} when $t_f^* = t_p$. This condition is rather difficult to guarantee during practical applications but can be approximately achieved by judicious choice of control weighting.

3.3 Control limits from optimal solution- example

In this section, a linear spring-mass-damper example is used to demonstrate that the area norm of the optimal control solution is indeed a reasonable measure of the control limit.

Consider a linear spring-mass-damper system with damping coefficient $\zeta = 0.7$ and frequency $\omega_n = 2.0$ rad/sec. This system can be mathematically represented in the following differential equation form:

$$\ddot{x} + 2\zeta\omega_n\dot{x} + \omega_n^2x = u \quad (80)$$

and can be re-written in linear state-space form as

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -4.0 & -2.8 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u \quad (81)$$

Let $y_p = x_1$ be the limit parameter with an upper limit of $y_{lim} = 5.0$ and the system is assumed to be completely determinate. The initial condition of the system states are set as $[0 \ 0]^T$. Non-real time collocation method [3] is used to find the solution of the proposed optimal control problem.

Figure 19 presents the optimal control solution for a control weighting of 1.0. Also, figure 19 presents the state-space trajectories of the system for the optimal control solution. The upper limit boundary appears as a horizontal line in figure 19. The optimal final time in this case is found to be 2.0 seconds. The peak time and rise

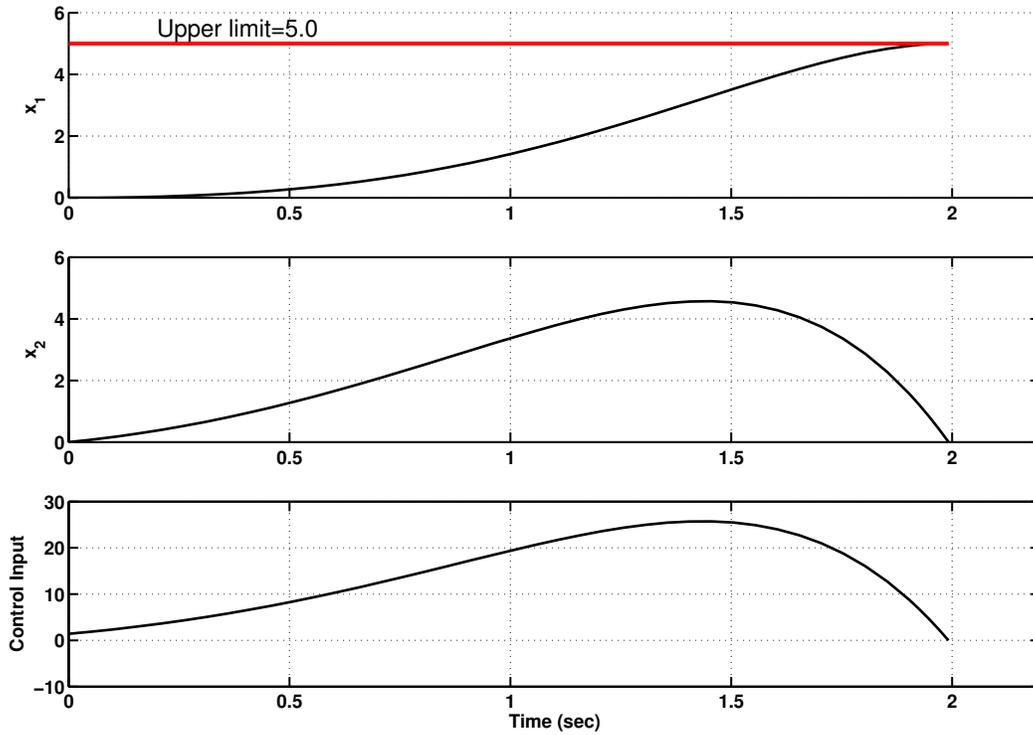


Figure 19: Spring-mass-damper- Optimal solution for $W=1.0$

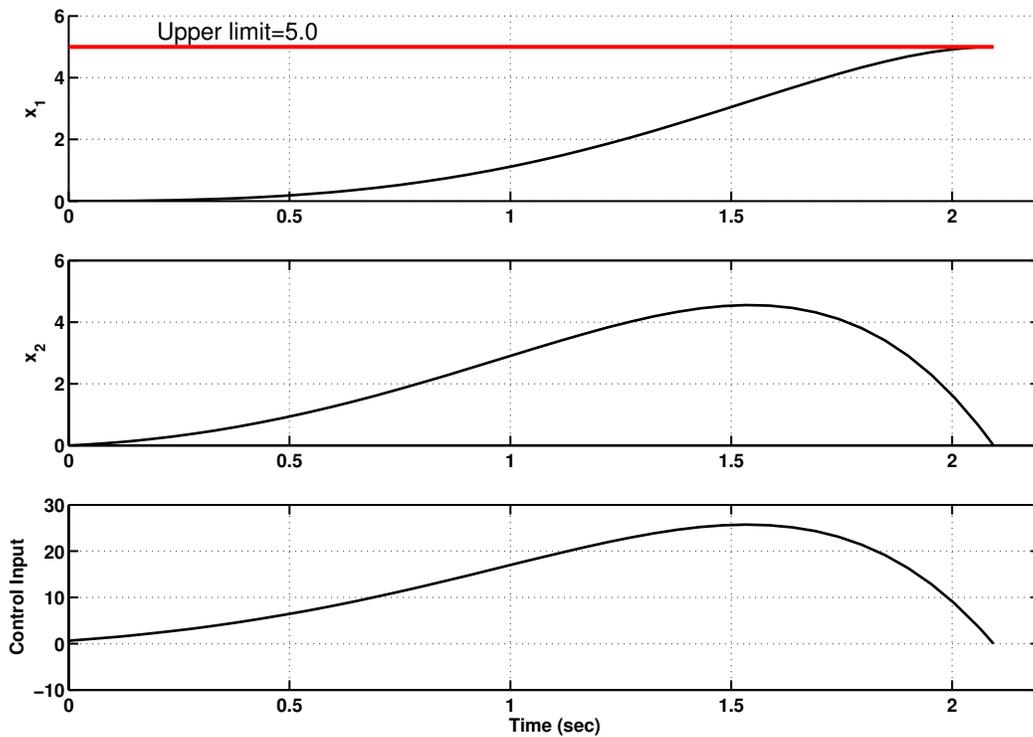


Figure 20: Spring-mass-damper- Optimal solution for $W=5.0$

time of the spring-mass-damper can be calculated using the following relations:

$$\text{Peak time } t_p = \frac{\pi}{\omega_d} = 2.2 \text{secs} \quad (82)$$

$$\text{Rise time } t_r = \frac{1}{\omega_d} \tan^{-1} \left(\frac{\omega_d}{-\sigma} \right) \quad (83)$$

where $\omega_d = \omega_n \sqrt{(1 - \zeta^2)}$ and $\sigma = \zeta \omega_n$. Comparison of the optimal final time with the peak and rise time brings out some interesting trends. The optimal final time (t_f^*) is found to be less than the peak time. Also, the optimal solution for a control weighting of 5.0 is presented in figure 20. It is found that increasing the weighting on the control results in the optimal time approaching the peak time, i.e, $t_f^* \rightarrow t_p$ and the optimal final time for $W = 5.0$ is 2.1 seconds. This is consistent with the contention that increasing control weighting corresponds to restricting pilot aggressiveness and hence more time to reach the limit boundary. Using equation 68, the control limit can be computed for both cases of control weighting. While the control limit for $W = 1.0$ is estimated to be 16.81 m/sec^2 , increasing control weighting to 5.0 results in a lower value of 16.0 m/sec^2 as expected. Therefore, the choice of the control weighting should be made judiciously so as not to overly restrict pilot aggressiveness.

3.4 Introduction to B-splines

B-spline refers to basis splines and are extremely useful for approximating functions in a domain with a small number of variables (B-spline coefficients). B-spline representation or B-form of a function is utilized in a wide variety of applications because of its ability to retain many of the important characteristics of the original function. In this section, a brief overview of B-splines and its approximation properties are presented.

The development of B-spline begins with a polynomial of order n , represented as:

$$p(x) = a_1 + a_2x + \dots + a_nx^{n-1} = \sum_{j=1}^n a_jx^{j-1} \quad (84)$$

All polynomials of order n form a linear space, denoted as Π_{n-1} . An important definition necessary for both polynomial and B-spline interpolation is the *divided difference*.

Definition 3.4.1. *The k th divided difference of a function g at the sites $\tau_i, \tau_{i+1} \dots \tau_{i+k}$ is the leading coefficient of the polynomial of order $k + 1$ that agrees with g at the sequence $\tau_i, \tau_{i+1} \dots \tau_{i+k}$. It is denoted by $[\tau_i, \tau_{i+1} \dots \tau_{i+k}]g$.*

Based on definition 3.4.1 a polynomial function approximating function g at sites τ_1, τ_2 is given by-

$$p(x) = [\tau_1]g + (x - \tau_1)[\tau_1, \tau_2]g \quad (85)$$

The same polynomial interpolate, approximating g at sites τ_1, τ_2, τ_3 will be represented as,

$$p(x) = [\tau_1]g + (x - \tau_1)[\tau_1, \tau_2]g + (x - \tau_1)(x - \tau_2)[\tau_1, \tau_2, \tau_3]g \quad (86)$$

Using equation 86 we get,

$$\begin{aligned} (\tau_3 - \tau_1)(\tau_3 - \tau_2)[\tau_1, \tau_2, \tau_3]g &= [\tau_3]g - [\tau_1]g - (\tau_3 - \tau_1)[\tau_1, \tau_2]g \\ \implies (\tau_3 - \tau_2)[\tau_1, \tau_2, \tau_3]g &= \frac{[\tau_3]g - [\tau_1]g}{\tau_3 - \tau_1} - [\tau_1, \tau_2]g \\ \implies (\tau_3 - \tau_2)[\tau_1, \tau_2, \tau_3]g &= [\tau_3, \tau_1]g - [\tau_1, \tau_2]g \\ \implies [\tau_1, \tau_2, \tau_3]g &= \frac{[\tau_3, \tau_1]g - [\tau_1, \tau_2]g}{(\tau_3 - \tau_2)} \end{aligned} \quad (87)$$

Re-sequencing the sites of interpolation $1 \rightarrow 2, 2 \rightarrow 3, 3 \rightarrow 1$ does not affect the divided difference. Therefore,

$$[\tau_1, \tau_2, \tau_3]g = \frac{[\tau_1, \tau_2]g - [\tau_2, \tau_3]g}{(\tau_1 - \tau_3)} \quad (88)$$

Generalizing equation 88 the expression for divided difference can be formulated as:

$$[\tau_i, \tau_{i+1} \dots \tau_{i+k}]g = \frac{[\tau_i, \dots, \tau_{r-1}, \tau_{r+1}, \dots, \tau_{i+k}]g - [\tau_i, \dots, \tau_{s-1}, \tau_{s+1}, \dots, \tau_{i+k}]g}{\tau_s - \tau_r} \quad (89)$$

According to definition 3.4.1, for a polynomial of order $k+1$ the k th divided difference is always a constant and also, the n th divided difference ($n > k+1$) will be zero. The expression for j th *normalized* B-spline of order k for knot sequence $\bar{\mathbf{t}}$ is given as:

$$B_{j,k,\bar{\mathbf{t}}} = (t_{j+k} - t_j)[t_j, t_{j+1}, \dots, t_{j+k}](\cdot - x)_+^{k-1} \quad (90)$$

B-splines are easily computed using the following recurrence relation derived in appendix A.

$$B_{j,k,\bar{\mathbf{t}}} = \omega_{jk}B_{j,k-1} + (1 - \omega_{j+1,k})B_{j+1,k-1} \quad (91)$$

In equation 91,

$$\omega_{jk} \triangleq \frac{x - t_j}{t_{j+k} - t_j} \quad (92)$$

Note that $B_{j,k,\bar{\mathbf{t}}}$ is zero outside the interval $[t_j, t_{j+k}]$. This is because when $x < t_j$ the function $(t - x)_+^{k-1}$ is a polynomial of order k whose $(k+1)$ th divided difference will be zero. Also, when $x > t_{j+k}$ the function $(t - x)_+^{k-1}$ is zero by definition. Inside the interval $[t_j, t_{j+k}]$ the b-spline $B_{j,k,\bar{\mathbf{t}}}$ is positive. This is because both $0 < \omega_{jk} < 1$ and using the recurrence relation of B-spline we can find that the spline, inside the support interval, is always positive.

Definition 3.4.2. *Spline Space $\mathbb{S}_{k,t}$ is defined as the collection of all functions obtained from the linear combinations of B-spline basis functions of order k and knot-sequence $\bar{\mathbf{t}}$. Mathematically,*

$$\mathbb{S}_{k,t} \triangleq \left\{ \sum_i \alpha_i B_{i,k,\bar{\mathbf{t}}} : \alpha_i \text{ is real } \forall i \right\} \quad (93)$$

Definition 3.4.3. *The linear space $\Pi_{k,\xi}$ refers to a collection of piecewise polynomial function, each of order k with break sequence $\xi = (\xi)_1^{l+1}$. What this means is that in between two consecutive breakpoints, say $[\xi_i, \xi_{i+1}]$ the polynomial P_i of order k exists. This polynomial has k free polynomial coefficients. Now there are l such polynomial*

pieces. So the total number of degrees of freedom for this linear space is kl . In other words, $\Pi_{<k,\xi}$ is the direct sum of l copies of $\Pi_{<k}$. The linear space $\Pi_{<k,\xi,\nu}$ is a subspace of the linear space $\Pi_{<k,\xi}$. The typical homogeneous conditions require that the piecewise polynomial function $f \in \Pi_{<k,\xi}$ be constructed to have a certain number of continuous derivatives. These continuity conditions are represented in the form:

$$\text{jump}_{\xi_i} D^{j-1} f = 0 \quad j = 1, \dots, \nu_i \quad \text{and} \quad i = 2, \dots, l \quad (94)$$

ν_i counts the number of continuity conditions at ξ_i . The function $\text{jump}_{\alpha} f \triangleq f(\alpha^+) - f(\alpha^-)$. Imposing these continuity conditions reduces the number of degrees of freedom from the linear space $\Pi_{<k,\xi}$ to the space $\Pi_{<k,\xi,\nu}$. The reduced number of degrees of freedom will be $n = kl - \sum_{i=2}^l \nu_i$. A very important theorem for B-splines is the Curry and Schoenberg theorem.

Theorem 3.4.1. For a given strictly increasing sequence $\xi = (\xi_i)_1^{l+1}$ (breakpoints) and a given non-negative integer sequence $\nu = (\nu_i)_2^l$ (continuity conditions at breakpoints) with $\nu_i \leq k$, all i , set

$$n \triangleq k + \sum_{i=2}^{i=l} (k - \nu_i) = kl - \sum_{i=2}^l \nu_i = \dim \Pi_{<k,\xi,\nu} \quad (95)$$

and let $\mathbf{t} \triangleq (t_i)_1^{n+k}$ be the non-decreasing sequence from ξ by the following two requirements:

1. for $i = 2, \dots, l$, the number ξ occurs exactly $k - \nu_i$ times in the knot sequence \mathbf{t} .
2. $t_1 \leq t_2 \dots t_k \leq \xi_1$ and $\xi_{l+1} \leq t_{n+1} \leq \dots t_{n+k}$.

Then the sequence B_1, \dots, B_n of B-splines of order k for the knot sequence \mathbf{t} is a basis for $\Pi_{<k,\xi,\nu}$, considered as functions on the interval $I_{k,\mathbf{t}} = [t_k, t_{n+1}]$. In symbols,

$$\mathcal{B}_{k,\mathbf{t}} = \Pi_{<k,\xi,\nu} \quad \text{in the interval} \quad I_{k,\mathbf{t}} \quad (96)$$

Summarizing Curry-Schoenberg theorem, if we are trying to approximate a function within an interval $[a, b]$, first choose the number of breakpoints (points of interpolation) within this interval. Additionally, the B-form of the function may need to satisfy certain continuity conditions at each of the *interior* breakpoints. The knot sequence is generated from the breakpoints based on the order of splines used for approximating the function and the continuity conditions required at each of the interior breakpoints in the interval.

3.5 Nonlinear trajectory generation for real-time optimal solution

The collocation method is one among the many different methods used to find solutions for optimal control problems. However, this approach cannot be used to find solutions in real-time as it is based on discretization which results in large number of unknowns to be solved simultaneously.

Nonlinear Trajectory Generator (NTG) [40, 39] is a software package that can be used to find optimal nonlinear trajectories for dimensionally flat systems in real-time [40]. Dimensionally flat systems, by definition, can be completely described using just a few variables and their derivatives. These variables are referred to as flat outputs of the system. The adaptive estimate of limit parameter dynamics available from the on-line tuning of SHL-NN weights is a dimensionally flat system. The flat output of this system is the estimate of limit parameter (\hat{y}_p). Also, while using NTG, it is important to identify any additional variables that arise in the optimal control problem from sources other than the system. In this case, the final time in the objective function is free. Therefore, the optimal control problem for estimating control limits with respect to a limit parameter has two flat outputs-

1. Limit parameter estimate that describes the system
2. Final time t_f which comes from the objective function

In order to reduce computational workload in solving the optimal control problem the flat outputs are approximated using B-spline [6] basis functions of appropriate order and multiplicity. The order and multiplicity chosen are based upon smoothness conditions for individual variables. For example, in the case of limit parameters of relative degree greater than one, the estimate of limit parameter must have at least $r - 1$ continuous derivatives. The choice of order, multiplicity and knots fixes the set of basis functions that can be used to describe the flat outputs and other variables. By varying the coefficients of the B-spline basis functions describing these variables, different solutions can be obtained. The NTG then uses commercially available optimization packages (NPSOL) to find the optimal set of these coefficients that minimize the given objective function while satisfying all the constraints.

3.5.1 B-spline approximation of continuous functions

Define,

$$\|g\| \triangleq \max |g(x)| \quad a \leq x \leq b \quad (97)$$

and the modulus of continuity of function g as,

$$\omega(g, h) \triangleq \max \left\{ |g(x) - g(y)| : |x - y| \leq h, x, y \in [a, b] \right\}. \quad (98)$$

Let breakpoints be chosen as $\xi = (\xi_i)_{i=1}^{l+1}$. Then the spline approximation of this function is given by,

$$\sum_{i=1}^n \alpha_i B_{i,k,\bar{\tau}} \quad \text{where} \quad n = kl - \sum_{i=2}^l \nu_i \quad (99)$$

Consider a special case of the above function

$$Ag \triangleq \sum_{i=1}^n g(\tau_i) B_{i,k,\bar{\tau}} \quad (100)$$

where $\tau = (\tau_i)_{i=1}^n$ is an arbitrary sequence of points within the interval $[\xi_1, \xi_{l+1}] = [a, b]$. By definition, the spline function Ag matches g at each τ_i . The significance of

Ag is that it can be used to generate a useful estimate for the error in approximation. For this consider any arbitrary location $a \leq \hat{x} \leq b$. Then,

$$Ag(\hat{x}) = \sum_{i=1}^n g(\tau_i) B_{i,k,\bar{\mathbf{t}}}(\hat{x}) \quad (101)$$

Also,

$$g(\hat{x}) = g(\hat{x}) \sum_{i=1}^n B_{i,k,\bar{\mathbf{t}}}(\hat{x}) \quad \because \sum_{i=1}^n B_{i,k,\bar{\mathbf{t}}} = 1 \quad (102)$$

$$\therefore g(\hat{x}) - Ag(\hat{x}) = g(\hat{x}) \sum_{i=1}^n B_{i,k,\bar{\mathbf{t}}}(\hat{x}) - \sum_{i=1}^n g(\tau_i) B_{i,k,\bar{\mathbf{t}}}(\hat{x}) \quad (103)$$

$$= \sum_{i=j+1-k}^j \left(g(\hat{x}) - g(\tau_i) \right) B_{i,k,\bar{\mathbf{t}}}(\hat{x}) \quad \text{Assuming } \tau_j \leq \hat{x} \leq \tau_{j+1} \quad (104)$$

Therefore,

$$\begin{aligned} |g(\hat{x}) - Ag(\hat{x})| &\leq \sum_{i=j+1-k}^j |g(\hat{x}) - g(\tau_i)| B_{i,k,\bar{\mathbf{t}}}(\hat{x}) \\ &\leq \max \left\{ |g(\hat{x}) - g(\tau_i)| : j-k < i \leq j \right\} \end{aligned} \quad (105)$$

Choose τ_i inside the support of each B-spline basis function. For example, the support of the basis spline $B_{i,k,\bar{\mathbf{t}}}$ is the non-zero interval $[t_i, t_{i+k}]$. Therefore, let $\tau_i = t_{i+k/2}$ when k is even and $\left(t_{i+(k-1)/2} + t_{i+(k+1)/2} \right) / 2$ when k is odd. Assuming k is even we get,

$$\begin{aligned} &\max \left\{ |g(\hat{x}) - g(\tau_i)| : j-k < i \leq j \right\} \\ &\leq \max \left\{ |g(\hat{x}) - g(y)| : x, y \in [t_{j+1-k/2}, t_{j+1}] \quad \text{or} \quad x, y \in [t_j, t_{j+k/2}] \right\} \\ &\leq \omega(g, k|\mathbf{t}|/2) \\ &\leq \frac{(k+1)}{2} \omega(g, |\mathbf{t}|) \end{aligned} \quad (106)$$

where $|\mathbf{t}| \triangleq \max \Delta t_i \quad \forall \quad i$ is the mesh size of the knot sequence \mathbf{t} . Furthermore,

$$\text{dist}(g, \mathcal{S}_{k,\mathbf{t}}) \triangleq \min \left\{ \|g - s\| : s \in \mathcal{S}_{k,\mathbf{t}} \right\} \leq \|g - Ag\| \quad (107)$$

since $Ag \in \mathcal{S}_{k,\mathbf{t}}$. Note that if τ_i is chosen to be the Greville sites then the transformation Ag becomes Vg which is the Schoenberg's variation diminishing approximation. In general,

$$\text{dist}(g, \mathcal{S}_{k,\mathbf{t}}) \leq \text{const}_k \omega(g, \mathbf{t}) \quad (108)$$

From the above equation it can be concluded that the distance of any continuous function g from the linear space $\mathcal{S}_{k,\mathbf{t}}$ will go to zero as the mesh size is made smaller. However, for smoother functions with many continuous derivatives much better estimates can be given.

3.5.2 Approximation of smooth functions using B-splines

Previously, it was shown that

$$\text{dist}(g, \mathcal{S}_{k,\mathbf{t}}) = \text{dist}(g - s, \mathcal{S}_{k,\mathbf{t}}) \quad \forall s \in \mathcal{S}_{k,\mathbf{t}} \quad (109)$$

However, if function g has continuous first derivative then,

$$\begin{aligned} \text{dist}(g, \mathcal{S}_{k,\mathbf{t}}) &\leq \text{const}_k \omega(g - s, \mathbf{t}) \quad \forall s \in \mathcal{S}_{k,\mathbf{t}} \\ \text{const}_k \mathbf{t} \|Dg - Ds\| &\forall s \in \mathcal{S}_{k,\mathbf{t}} \cap C[a, b] \end{aligned} \quad (110)$$

Choose $s \in \mathcal{S}_{k,\mathbf{t}} \cap C[a, b]$ such that the above bound is as small as possible.

$$\text{dist}(g, \mathcal{S}_{k,\mathbf{t}}) \leq \text{const}_k \mathbf{t} \text{dist}(Dg, \mathcal{S}_{k-1,\mathbf{t}}) \quad (111)$$

since $\mathcal{S}_{k-1,\mathbf{t}} = \left\{ Ds : s \in \mathcal{S}_{k,\mathbf{t}} \cap C[a, b] \right\}$. Using the distance inequality derived earlier once again to estimate $\text{dist}(Dg, \mathcal{S}_{k-1,\mathbf{t}})$ in case Dg is continuous we get,

$$\begin{aligned} \text{dist}(Dg, \mathcal{S}_{k-1,\mathbf{t}}) &\leq \text{const}_{k-1} |\mathbf{t}| \omega(Dg, |\mathbf{t}|) \\ \therefore \text{dist}(g, \mathcal{S}_{k,\mathbf{t}}) &\leq \text{const}_k \text{const}_{k-1} |\mathbf{t}|^2 \omega(Dg, |\mathbf{t}|) \end{aligned} \quad (112)$$

Proceeding in this way, one obtains the *Jackson's theorem* for estimate of distance of a smooth function from $\mathcal{S}_{k,\mathbf{t}}$.

Theorem 3.5.1. *For $j = 0, \dots, k - 1$ there exists $const_{k,j}$ so that, for all $\mathbf{t} = (t_i)_1^{n+k}$ with $t_1 = \dots = t_k = a < t_{k+1} \leq \dots < b = t_{n+1} = \dots = t_{n+k}$ and for all $g \in C^{(j)}[a, b]$,*

$$dist(g, \mathcal{S}_{k,\mathbf{t}}) \leq const_{k,j} |\mathbf{t}|^j \omega(D^j g, |\mathbf{t}|) \quad (113)$$

According to theorem 3.5.1 a set of B-spline basis functions can approximate a function well if it is sufficiently smooth.

3.5.3 NTG real-time control limit estimation using true limit parameter dynamics

Real-time estimation of control limits for the spring-mass-damper system of equation 80, using NTG package, is presented here. For simplicity the limit parameter dynamics is assumed to be completely determinate, i.e, the optimal solution is found for the true system- not for the adaptive estimate of limit parameter dynamics. The flat output for this system is the position state x_1 since the velocity of the system x_2 can be obtained by differentiating the position with respect to time. The final time (t_f) is free and therefore, the state-space representation of the system given in equation 81 can be rewritten in terms of non-dimensional time variable $\tau = \frac{t}{t_f}$ as shown in equation 114

$$\begin{bmatrix} x_1' \\ x_2' \end{bmatrix} = \begin{bmatrix} 0 & t_f \\ -4.0t_f & -2.8t_f \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ t_f \end{bmatrix} u_p \quad (114)$$

where ()' denotes derivative with respect to non-dimensional time τ . The spring-mass-damper response for a sequence of step inputs without control limiting is presented in figure 21. Control limit is computed from the NTG optimal solution in the non-dimensional time interval $[0, 1]$ assuming an upper limit on the limit parameter

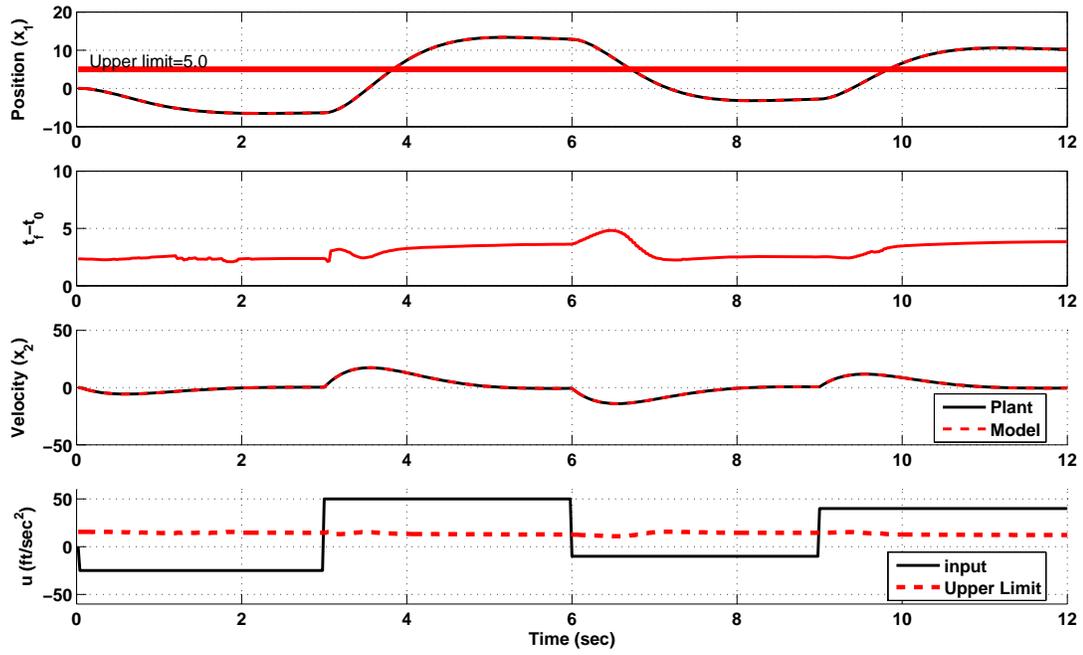


Figure 21: Spring-mass-damper response with envelope protection OFF, $W=0.5$

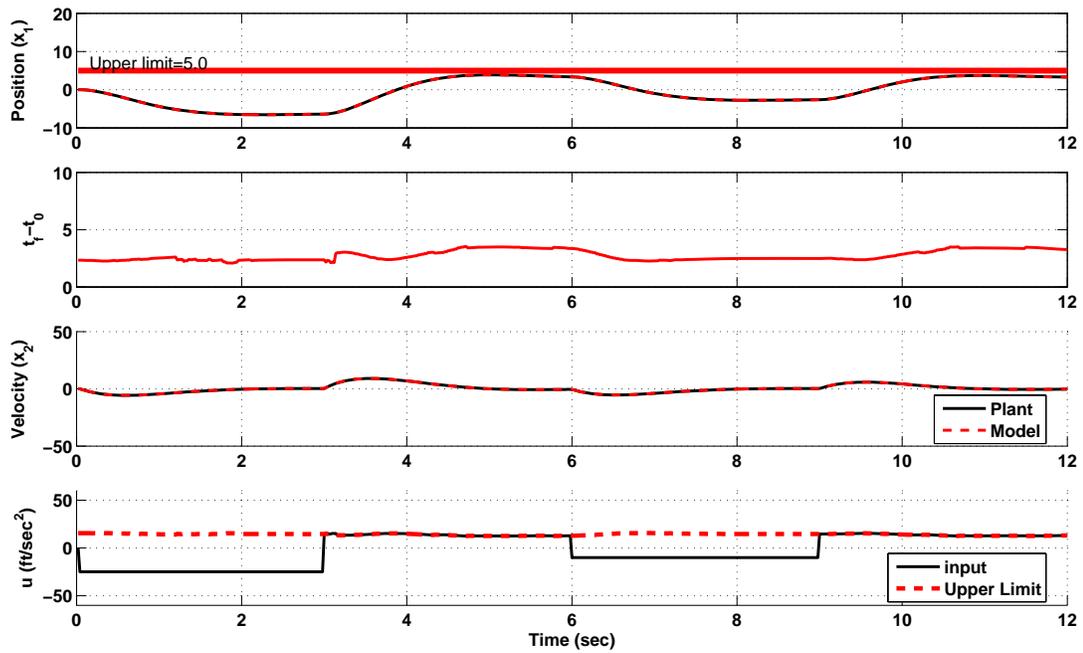


Figure 22: Spring-mass-damper response with envelope protection ON, $W=0.5$

($y_{lim} = 5$ meters) and a control weighting of 0.5. Let ξ represent the flat outputs of the problem. Details of their B-spline representations within the NTG are given in table 1. The system response with envelope protection ON is shown in figure

Table 1: Modeling of flat outputs for the spring-mass-damper system by NTG

Flat outputs	knots	order	multiplicity	Number of coefficients
$\xi_1 = x_1$	(0,0.5,1)	5	3	7
$\xi_2 = t_f$	(0,1)	1	0	1

22. When envelope protection is switched ON the control input is restricted to be within the computed control limit. Consequently, the limit parameter response (x_1) is maintained below its prescribed upper limit of 5 meters.

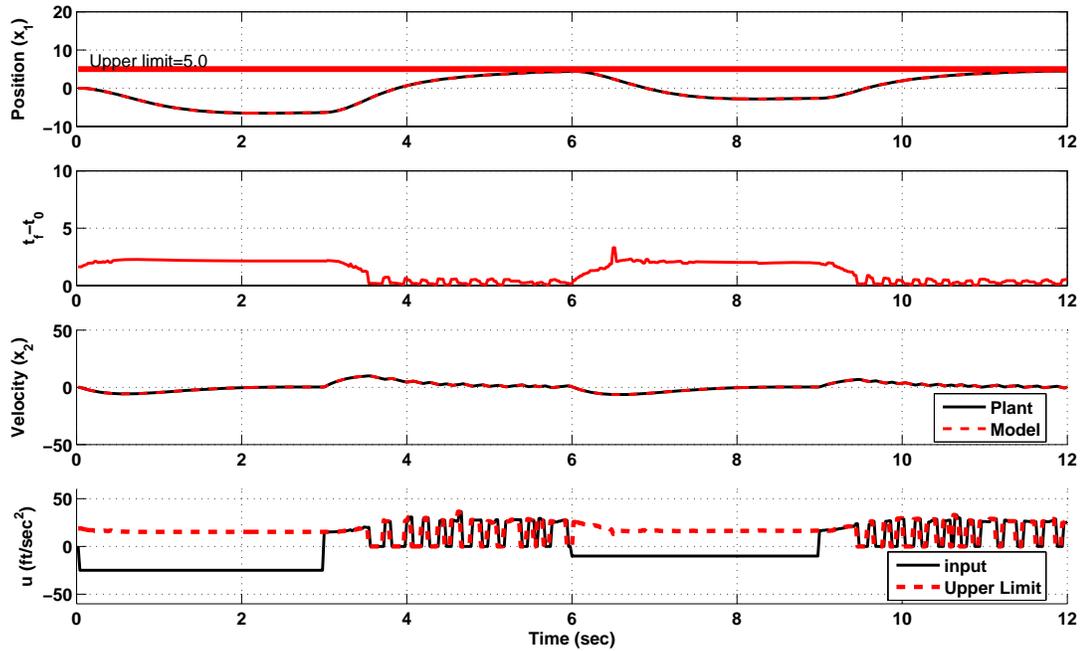


Figure 23: Spring-mass-damper response with envelope protection ON, $W=5.0$

The simulation with envelope protection ON and a control weighting value of 5.0 produces very similar results as shown in figure 23. Figure 23 also shows the variation of the critical time of the limit parameter response ($\Delta t_{crit} = t_f - t_0$). The critical time is a new and important parameter resulting from the proposed envelope protection scheme. This parameter is an additional variable not available in earlier envelope

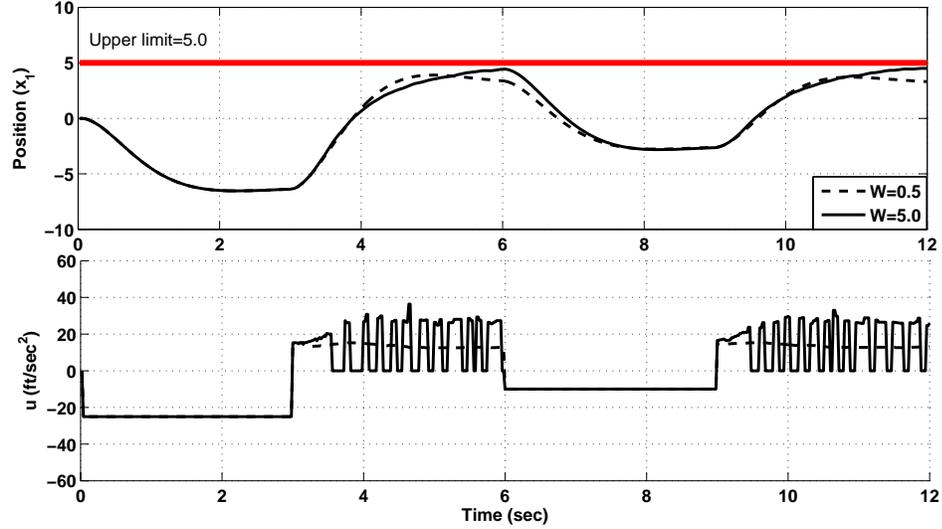


Figure 24: Spring-mass-damper limit parameter response variation with control weighting

protection methods and a sudden drop in its value is indicative of the limit parameter response approaching the limit boundary. Furthermore, the critical time of the limit parameter response in an integral part of the modified control limit calculation given in equation 76.

A comparison of the limit parameter response based on these two distinct values of control weighting is presented in figure 24. Figure 24 clearly shows that while the limit parameter response in both cases are very much similar, the value of control weighting determines how quickly the pilot is allowed to approach the limit boundary. Therefore, a good choice of this parameter would allow the pilot to safely and effectively utilize the envelope which is the main goal behind any envelope protection system.

Notice that the control input in figure 23 is oscillatory compared that of figure 22. This is caused due to the critical time Δt_{crit} approaching zero as limit parameter response approaches the limit boundary. These oscillations near the limit boundary can be prevented by choosing a smoothing function, as given in equation 77, and a threshold time. The results using adaptive estimate of limit parameter dynamics and a smoothing function are presented in the next subsection.

3.5.4 NTG real-time control limit estimation using adaptive estimate of limit parameter dynamics

In the previous subsection, it has been shown that the control limits computed from the NTG optimal solutions, when true limit parameter dynamics are known, successfully maintain the limit parameter response within the prescribed envelope. Let us assume now that spring-mass-damper system in equation 80 is only a linear approximation of the true limit parameter dynamics that are actually governed by the following differential equation,

$$\ddot{y}_p + 2.5\dot{y}_p + 8.0y_p = 1.5u_p \quad (115)$$

The estimate of the limit parameter dynamics is obtained by augmenting the linear model in equation 80 with an adaptive SHL-NN and a static error feedback assuming full-state information.

$$\ddot{\hat{y}}_p = -2.8\dot{\hat{y}}_p - 4.0\hat{y}_p + u_p + \nu_{ad}(\bar{\mu}) - K_e e - K_{\dot{e}} \dot{e} \quad (116)$$

where $e \triangleq \hat{y}_p - y_p$ and $\dot{e} \triangleq \dot{\hat{y}}_p - \dot{y}_p$. Also, the error feedback gains in equation 116 are chosen to be $K_e = 2.0$, $K_{\dot{e}} = 1.0$. The simulation time step $\Delta t_{sim} = 0.02$ seconds and the optimal control problem is solved at each simulation step. The adaptive neural network design and learning rate parameters are given in table 2. The neural network input vector consists of the following normalized system values-

$$\bar{\mu} = \left[1.0 \quad \frac{y_p}{5} \quad \frac{\dot{y}_p}{5} \quad \frac{\hat{y}_p}{5} \quad \frac{\dot{\hat{y}}_p}{5} \right]. \quad (117)$$

The system response for a series of step control inputs with envelope protection switched OFF is given in figure 25. The response of the adaptive estimate is observed to follow the true response closely in figure 25. An upper limit of 5 meters is imposed on the limit parameter response and the NTG is used to find the optimal solution using the adaptive estimate at each simulation time step. The control limit is computed from equation 76 using the area norm of the optimal solution and the smoothing

Table 2: NTG linear example- neural network design and learning rate parameters

N_{inp}	4	Output layer learning rate, Γ_m	5.0
N_{hid}	10	Hidden layer learning rate, Γ_n	0.2
N_{out}	1	E-mod parameter, κ_e	0.1
Basis function, σ	sigmoid $\frac{1}{1+e^{-a(x-c)}}$	Sigmoid parameters	$a = 1.0,$ $c = 0$

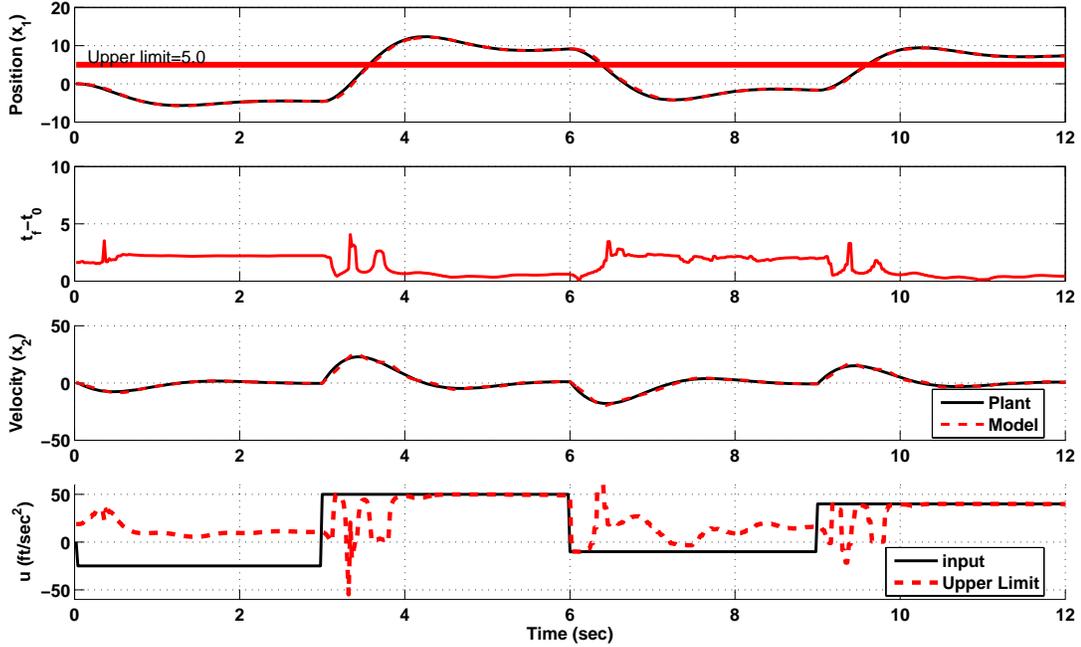


Figure 25: Spring-mass-damper control limit estimation using adaptive estimate-Envelope protection OFF, $W=5.0$

function given in equation 118. The threshold time of one second ($\Delta t_h = 1second$) is chosen for the smoothing function.

$$S(\Delta t_{crit}) = \begin{cases} 1 & t_f - t_0 \geq 1.0 \\ \exp\left(10(t_f - t_0 - 1.0)\right) & t_f - t_0 < 1.0 \end{cases} \quad (118)$$

This control limit is plotted along with the actual control input in figure 25. Notice that violation in the control channel is observed prior to the actual limit parameter violation of the upper limit. The estimated critical time of the limit parameter response (Δt_{crit}) is also presented in figure 25. In figure 25, the control margin is observed to increase sharply as the limit violation increases. This sharp increase in control margin corresponds to the sharp increase in critical time which in case of

limit parameter going outside the limit boundary is the amount of time required to aggressively bring the response back within the envelope. Figure 26 shows the limit

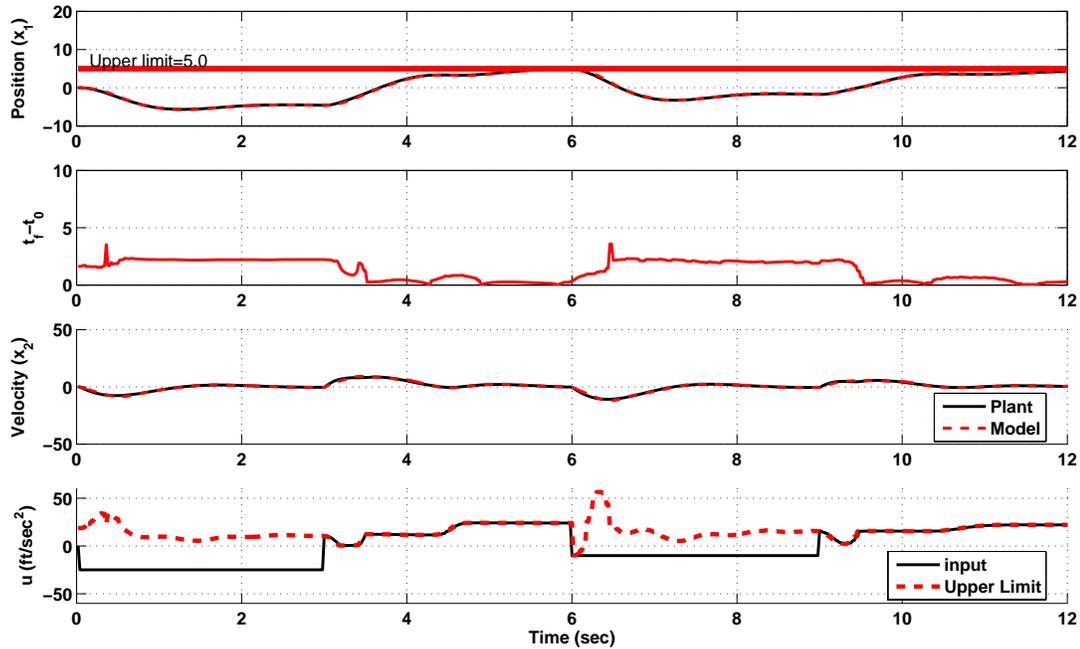


Figure 26: Spring-mass-damper control limit estimation using adaptive estimate-Envelope protection ON, $W=5.0$

parameter response when envelope protection is switched ON and the control inputs are restricted to be within the estimated control limits. Correspondingly, the limit parameter response is found to be successfully kept below the prescribed upper limit. The response of the adaptive estimate tracks the true response closely at all times. Also, notice that with the selection of an appropriate threshold time and a smoothing function the chattering observed earlier in figure 23 with control weighting 5.0 is removed. During simulation evaluations it was observed that the design and selection of smoothing function is more critical for higher values of control weighting. A value approximately near limit parameter control sensitivity is a good starting value for control weighting.

3.6 *Extension of envelope protection to multiple control limiting*

The proposed NTG approach for estimating control limits, currently formulated for a single control channel, can be easily extended to multiple control channels. Assume that the nonlinear dynamics of limit parameter $y_p \in \mathfrak{R}$ are of the following form.

$$y_p^{(r)} = h_r(\mathbf{x}, y_p, y_p^{(1)}, \dots, y_p^{(r-1)}, u_1, u_2, \dots, u_m). \quad (119)$$

The adaptive estimate of the limit parameter dynamics generated using on-line adaptation of neural network weights to minimize estimation error is represented in the following differential equation form:

$$\hat{y}_p^{(r)} = \sum_{i=0}^{r-1} a_i \hat{y}_p^{(i)} + \sum_{j=0}^m \frac{\partial \hat{h}_r}{\partial u_j} u_j + \nu_{ad}(\bar{\mu}) - \nu_{dc} \quad (120)$$

The adaptive estimate given in equation 120 can be written into the following equivalent SISO representation,

$$\hat{y}_p^{(r)} = \sum_{i=0}^{r-1} a_i \hat{y}_p^{(i)} + \left(\sum_{j=0}^m \frac{\partial \hat{h}_r}{\partial u_j} \right) u_p + \nu_{ad}(\bar{\mu}) - \nu_{dc} \quad (121)$$

In equation 121 $u_p \in R$ is some dummy control input on which limits will be estimated using the NTG approach. These limits are then re-directed or applied to each individual control channel. Minimum norm solution method is one such approach of control allocation.

$$\begin{aligned} \because \sum_{j=0}^m \frac{\partial \hat{h}_r}{\partial u_j} u_j &= \left(\sum_{j=0}^m \frac{\partial \hat{h}_r}{\partial u_j} \right) u_p \\ \sum_{j=0}^m \frac{\partial \hat{h}_r}{\partial u_j} u_{j,lim} &= \left(\sum_{j=0}^m \frac{\partial \hat{h}_r}{\partial u_j} \right) u_{lim}(t_0) \end{aligned} \quad (122)$$

where $u_{lim}(t_0)$ is given as in equation 76. According to the minimum norm solution method the control limits on each channel ($u_{j,lim}$) can be obtained using the following expression:

$$u_{j,lim} = \frac{\frac{\partial \hat{h}_r}{\partial u_j} \left(\sum_{j=0}^m \frac{\partial \hat{h}_r}{\partial u_j} \right) u_{AN}^*(t_0)}{\sum_{j=0}^m \frac{\partial \hat{h}_r}{\partial u_j}^2} \quad (123)$$

The application of the proposed extension is demonstrated using a modified form of the spring-mass-damper example given in equation 115. Let the true limit parameter dynamics be given as:

$$\ddot{y}_p + 2.5\dot{y}_p + 8.0y_p = 0.8u_1 + 0.7u_2 \quad (124)$$

with $y_p = x_1$ as the limit parameter. An upper and a lower limit of 5.0 and -4.0 are imposed on the limit parameter response respectively. The system initial condition is assumed to be $[0 \ 0]^T$. The adaptive estimate of limit parameter dynamics is given by the following equation:

$$\ddot{\hat{y}}_p = -2.8\dot{\hat{y}}_p - 4.0\hat{y}_p + 0.6u_1 + 0.4u_2 + \nu_{ad}(\bar{\mu}) - K_e e - K_d \dot{e} \quad (125)$$

The neural network parameters and system gains are chosen to be the same values used in the previous evaluations. The smoothing function used in both upper and lower control limit calculation is given in equation 126.

$$S(\Delta t_{crit}) = \begin{cases} 1 & t_f - t_0 \geq 1.0 \\ \exp\left(10(t_f - t_0 - 0.5)\right) & t_f - t_0 < 0.5 \end{cases} \quad (126)$$

The system response for a series of step control inputs with envelope protection switched OFF is given in figure 27. The response of the adaptive estimate is observed to follow the true response closely in figure 27. Notice that with envelope protection switch OFF, the control inputs violate the control limits (exceed upper control limit or go below the lower control limit) prior to the actual violation of the limit boundary.

The limit parameter response with envelope protection switched ON is shown in figure 28. Note that the control inputs are now restricted to be within the estimated control limits. Correspondingly, the limit parameter response is found to be successfully kept within the prescribed limits. Also, the response of the adaptive estimate continues to follow the true response closely.

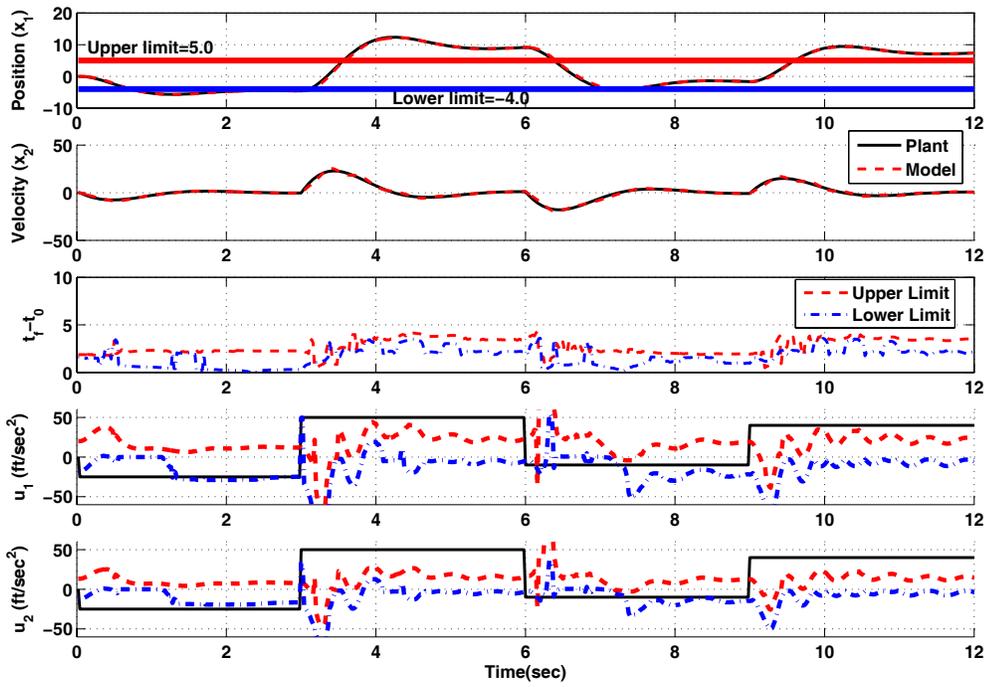


Figure 27: Spring-mass-damper control limit estimation using adaptive estimate-Envelope protection OFF, $W=1.0$

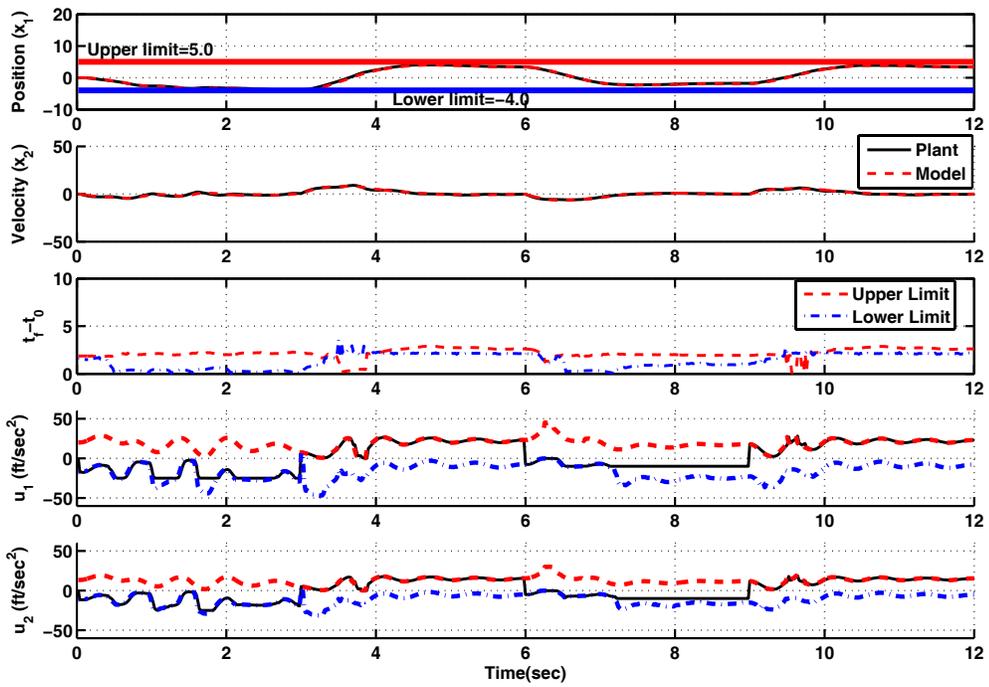


Figure 28: Spring-mass-damper control limit estimation using adaptive estimate-Envelope protection ON, $W=1.0$

3.7 Pilot-in-the-loop evaluations

Force-feedback tactile cueing has been effectively used for envelope protection in case of pilot-in-the loop systems. While, dynamic trim based limit protection modules have been successfully evaluated for cueing steady-state-response critical limits, the method is not applicable for cueing transient-response critical limits such as the longitudinal hub moment. The proposed approach is evaluated for hub moment limit protection within a test environment that includes an active sidestick inceptor for force-feedback tactile cueing.

3.7.1 Development and testing environment

The Real-Time Interactive Prototype Technology Integration Development Environment (RIPTIDE) served as the development and testing tool. RIPTIDE combines a control system executable with a helicopter math model, in this case GENHEL, and renders the states as a pilots view with OpenGL PerformerTM. GENHEL math model provides the vehicle dynamics for the UH-60A Black Hawk and its control system, including its SCAS.

The limit protection algorithm was created as a Simulink[®] block diagram that was auto-coded and compiled with Real-Time Workshop. It serves as a limit prediction and avoidance cue module within the developing Open Platform for Limit Protection (OPLP) which evolved from the design of reference [23].

This platform (see figure 29) structures limit protection mechanisms and distributes them across the control loop for cognizant, reflexive, and autonomous limit protection. The GENHEL model effectively served as both the Flight Control System (FCS) and Aircraft Dynamics blocks in the figure.

RIPTIDE allows a choice of inceptors and three were used. A stirling dynamics active sidestick model SA-S-2D-1, shown in figure 30, served as the longitudinal and lateral cyclic and provided the active force-feedback cues for those axes. The setting

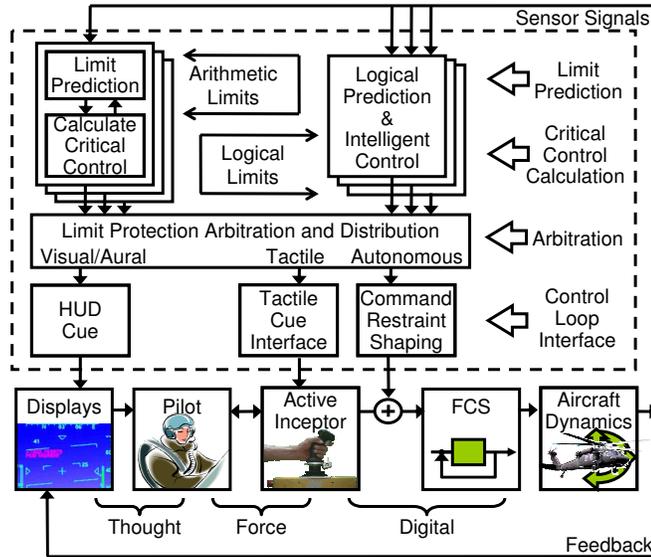


Figure 29: Open platform for limit protection

used for the nominal force-feel system of the active sidestick are also given in figure 30. A Microsoft Precision 2 joystick provided passive collective control and CH Pro Pedals provided passive anti-torque (yaw) control. The signals for these inceptors were mapped to the four cockpit control inceptor axes of the GENHEL model.

The tactile avoidance cue took the form of a 30 N softstop with a 1° length. This made it approximate a step-force softstop. Static flight simulations took place at Georgia Tech with RIPTIDE running on a Dual Xeon™ 1.7 GHz workstation with an NVIDIA® Quadro4 64Mb graphics card. A pilots 53° field of view was projected 1.7 meters before test subjects, who were seated with the cyclic active sidestick placed at their right hand, the collective joystick at their left hand, and the pedals at their feet (see figure 31). Evaluation maneuvers were performed by a rated helicopter aviator familiar with tactile cueing and the Black Hawk.

3.7.2 Test maneuvers

The Swoop maneuver (See Table 3) is a dynamic maneuver that tests pitch related limits from hover through high speed forward flight. From out of ground effect (OGE)

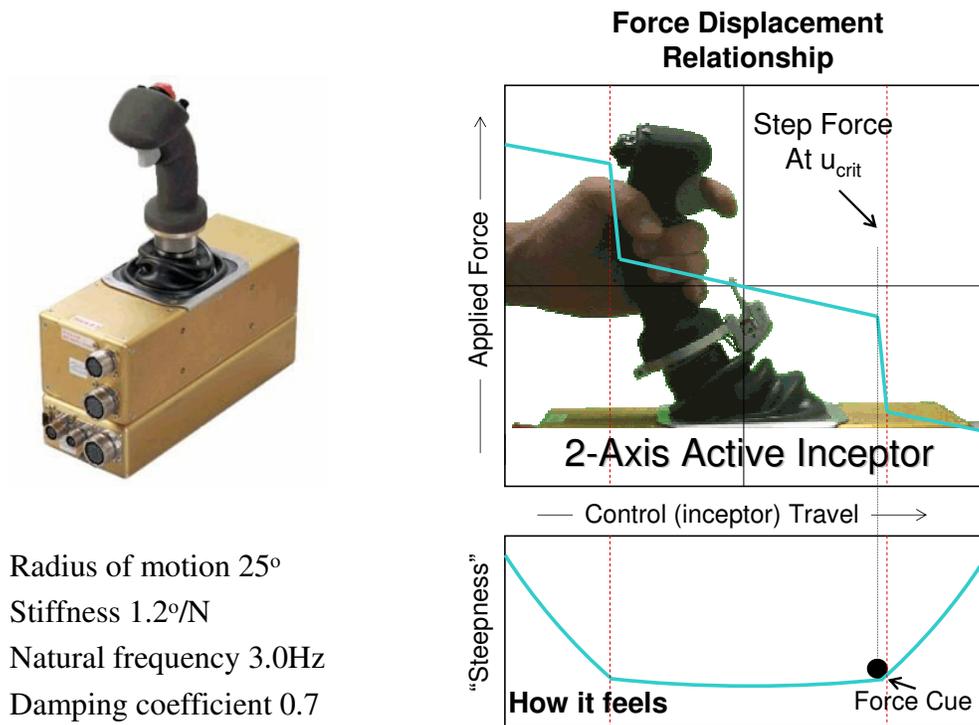


Figure 30: Active sidestick inceptor- stirling dynamics model SA-S-2D-1

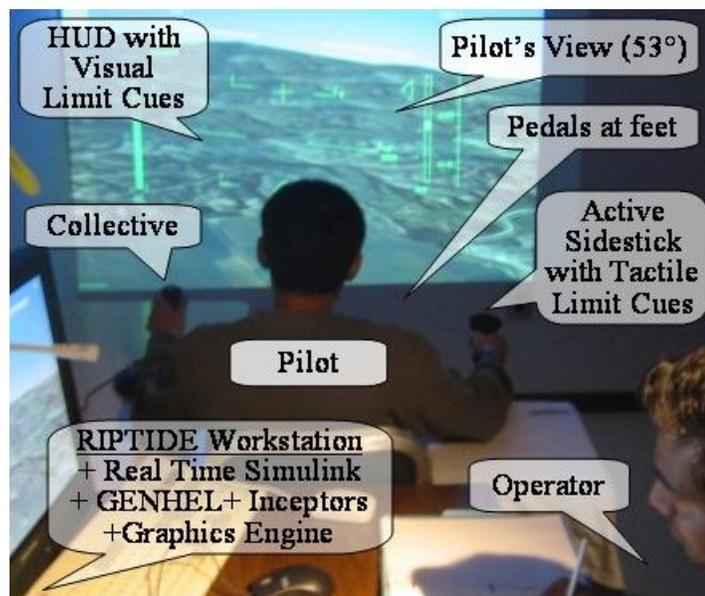


Figure 31: Carefree maneuver workshop setup at Georgia Tech

hover, using an abrupt forward cyclic command, the pilot rapidly pitches to a 50 degree nose down attitude. This attitude is held, allowing the diving aircraft to accelerate, until the airspeed reaches 50 knots. Then, via a steady but rapid aft cyclic, the pilot executes a symmetrical pull-up to a nose high +50 degree attitude. When the decelerating aircraft reaches an appropriate airspeed, the pilot executes a rapid pitch down to complete the maneuver at an OGE hover.

Table 3: Swoop maneuver performance specification

	Desired	Adequate
Begin at OGE Hover	± 5 kts	± 5 kts
Attain target pitch attitude, -50°	$\pm 5^\circ$	$\pm 10^\circ$
Begin pull-up at target airspeed, 50 kts	± 5 kts	± 10 kts
Attain target pitch attitude, $+50^\circ$	$\pm 10^\circ$	$\pm 15^\circ$
Complete maneuver at OGE Hover	± 10 kts	± 15 kts
Maintain angular deviations in roll and yaw within $\pm X$ degrees from the initial unaccelerated level flight condition to completion of the maneuver	$\pm 10^\circ$	$\pm 20^\circ$
Collective pitch remains constant throughout the maneuver	✓	✓

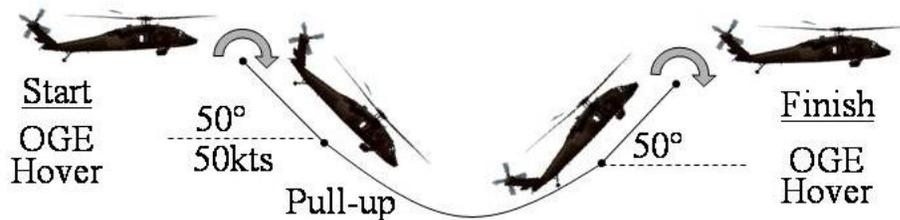


Figure 32: Graphic representation of swoop maneuver

The collective pitch setting (for OGE power) remains fixed throughout the maneuver. The maneuver is performed up-and-away, without a specific test course or the need

for peripheral visual displays.

3.7.3 Force-feedback tactile cueing for hub moment limit protection

The pilot task was to execute swoop maneuvers with and without the aid of tactile cues. Equation 127 presents the linear approximate model for the limit parameter which is chosen based upon the off-line analysis of the swoop maneuver data using the system identification toolbox in Matlab.

$$\dot{\hat{y}}_p = -5\hat{y}_p + 400000u_\delta \quad (127)$$

The control input u_δ in equation 127 refers to the longitudinal cyclic input. The same analysis is used to realize the limit parameter response as fourth order dynamics with relative degree one. The linear approximate model chosen is given in equation 127 which is augmented with an adaptive SHL-NN with design and learning rate parameters given in table 4.

Also, static error feedback with a gain of 10.0 is used in the role of linear dynamic compensator. The resulting adaptive estimate of the limit parameter dynamics is given by equation 128.

$$\dot{\hat{y}}_p = -5\hat{y}_p + 400000u_\delta + \nu_{ad}(\bar{\mu}) - K(\hat{y}_p - y_p) \quad (128)$$

Table 4: Hub moment limit protection- NN design and learning rate parameters

N_{inp}	7	Output layer learning rate, $\mathbf{\Gamma}_m$	8.0
N_{hid}	14	Hidden layer learning rate, $\mathbf{\Gamma}_n$	0.9
N_{out}	1	E-mod parameter, κ_ϵ	0.01
Basis function, σ	sigmoid $\frac{1}{1+e^{-a(x-c)}}$	Sigmoid parameters	$a = 1.0,$ $c = 0$

The simulation step size is 0.01 seconds and the NN input vector is:

$$\bar{\mu} = \left[\begin{array}{cccccc} \frac{y_p(t)}{40000} & \frac{y_p(t-d)}{40000} & \frac{\hat{y}_p(t)}{40000} & \frac{u_B}{120} & \frac{w_B}{50} & \frac{a1}{10} & q \end{array} \right]^T \quad (129)$$

where u_B , w_B are the body x and z velocity components, respectively. Also, q is the pitch rate response of the vehicle and $a1$ is the flapping angle. The control limit is

computed from the NTG solution of the optimal control problem which has two flat outputs (ξ). Table 5 gives details of how these flat outputs are modeled using b-spline curves in the non-dimensional unit time interval.

Table 5: Modeling of flat outputs for hub moment limit protection system

Flat outputs	knots	order	multiplicity	Number of coefficients
$\xi_1 = \hat{y}_p$	(0,0.5,1)	5	3	7
$\xi_2 = t_f$	(0,1)	1	0	1

The limit protection system is evaluated for two different vehicle configurations to emphasize the adaptive architecture of the proposed approach. Conf-1 is called the nominal vehicle configuration and conf-2 is referred to as the heavy vehicle configuration the details of which are presented in table 6.

Table 6: Vehicle configurations used for evaluating hub moment limit protection system

Configuration No.	CG location	Weight	Name
1	357.3 inches	16825 lb	nominal
2	357.3 inches	20000 lb	heavy

The smoothing function used in computing the control limit from the area norm of the optimal control solution [see equation 76] is plotted in figure 33 and is obtained from equation 130.

$$S(t_f - t_0) = \begin{cases} 1 & t_f - t_0 \geq 0.05 \\ \exp\left(172(t_f - t_0 - 0.05)\right) & t_f - t_0 < 0.05 \end{cases} \quad (130)$$

Note that the threshold time value for the smoothing function is fixed as 0.05 seconds. The hub moment response plot for a typical swoop maneuver without the aid of tactile cue for the nominal vehicle configuration is presented in figure 34. Without tactile cueing in the longitudinal channel of the active inceptor the hub moment response is observed to be violating both the upper and lower response limits of 20,000 lb-ft and -20,000 lb-ft respectively.

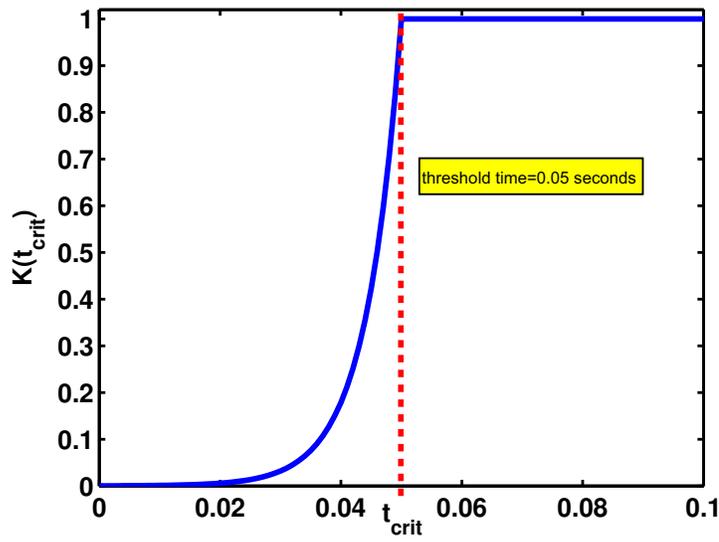


Figure 33: Smoothing function for hub moment limit protection

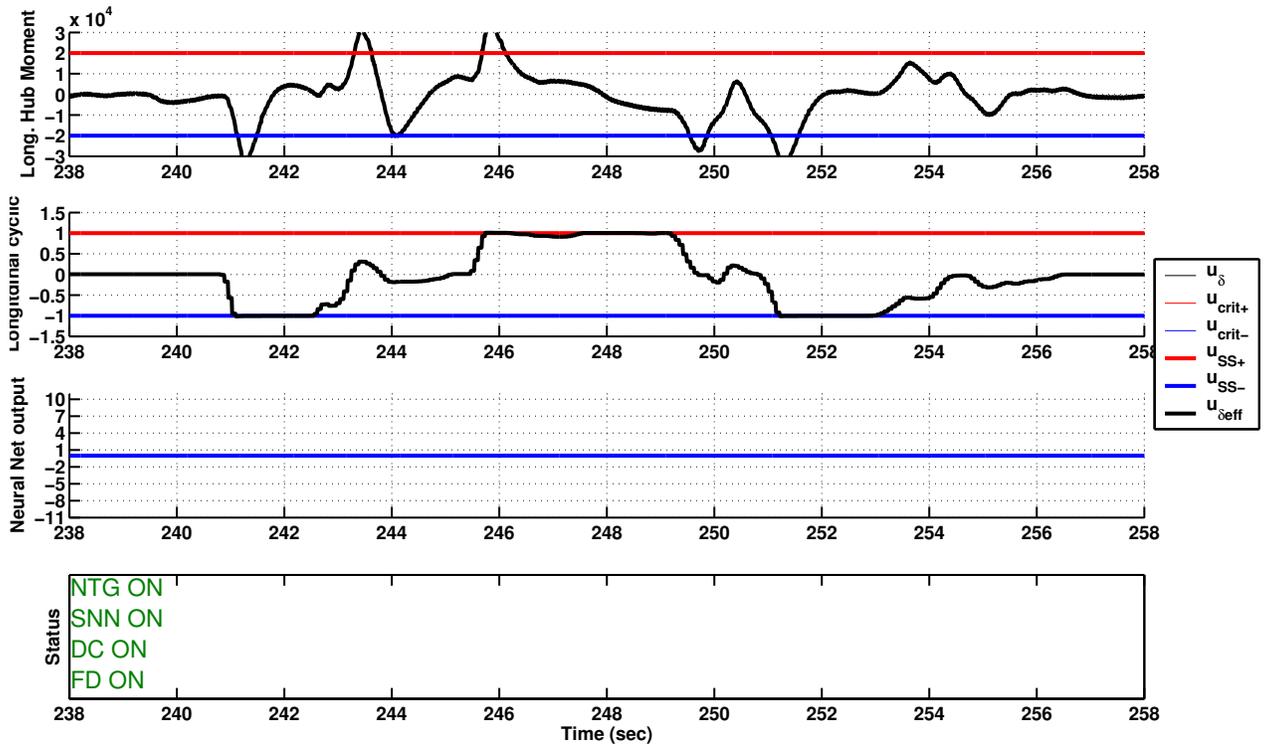


Figure 34: Swoop maneuver without hub moment limit protection- Nominal configuration

In figure 35 the hub moment response is shown for the sloop maneuver with tactile cueing for hub moment limit protection. The adaptive estimate approximates the hub moment dynamics closely as observed from response plots in figure 35. The output of the neural network is also plotted. The status of all the individual modules, involved in the evaluation, are also shown in the plots.

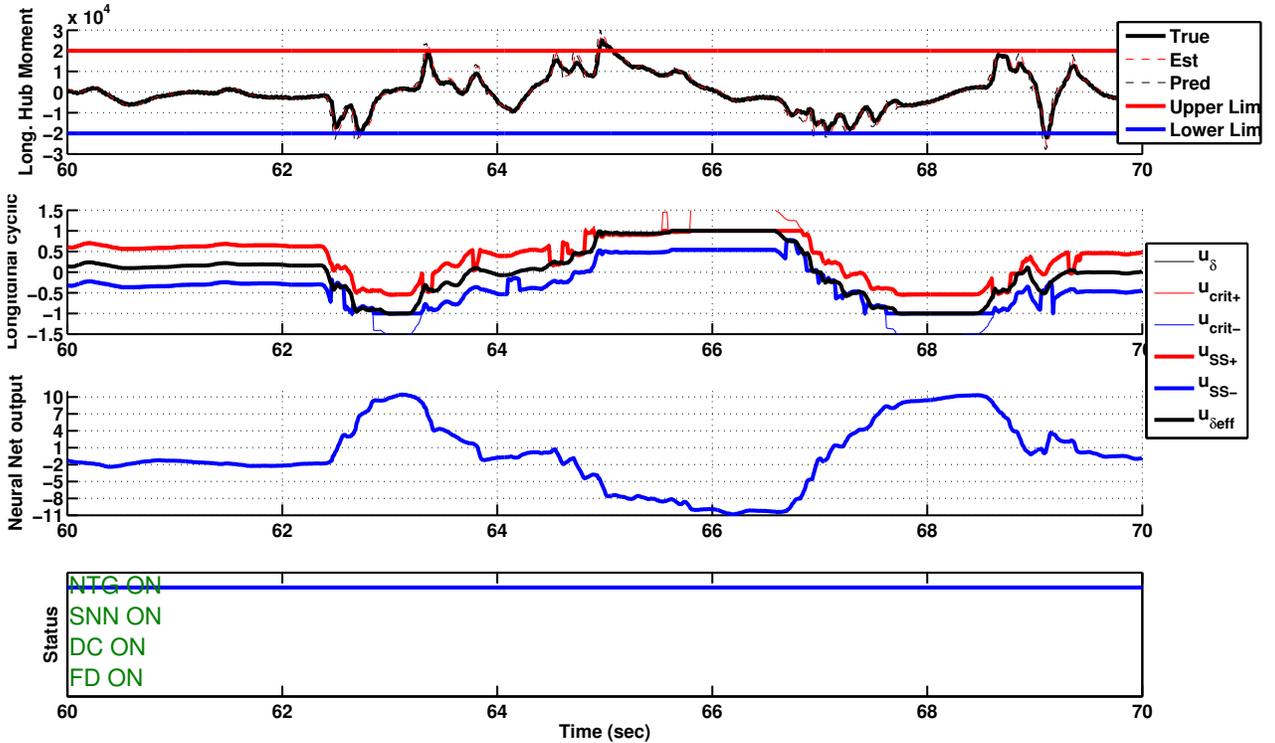


Figure 35: Sloop maneuver with hub moment limit protection- Nominal configuration

For the results presented in figure 35 note that only the NTG based hub moment limit protection system is active. The control limits prescribed by the NTG hub moment limit cue module enable the pilot to safely execute the sloop maneuver. It is important to note that the hub moment response is very sensitive to the longitudinal cyclic control input (see equation 127). Therefore, the calculation of control limit relies heavily on the critical time which in this case is highly oscillatory during the maneuver. Hence, the control limits and consequentially the softstops are observed to

drop suddenly and approach the current stick location. This is nothing but a reflection of the high frequency and transient nature of the hub moment limit parameter dynamics. There are in total six variables associated with the longitudinal cyclic control channel. Among these two variables are associated with each limit boundary. For instance, u_{crit+} and u_{crit-} refer to the critical control positions corresponding to the upper and lower limits respectively. Similarly, u_{SS+} and u_{SS-} are the softstop locations on the active sidestick associated with the upper and lower limits of hub moment response. Usually, the critical control positions differ from their softstop counterparts by just a fixed bias value. During this evaluation their relationship can be represented by the following equation,

$$u_{SS+} = u_{crit+} - 0.04 \quad (131)$$

$$u_{SS-} = u_{crit-} + 0.04 \quad (132)$$

In equations 131 and 132 the bias value of 0.04 is also the length of the softstop. Finally, the remaining two variables u_{δ} and $u_{\delta_{eff}}$ pertain to the control system. The control system translates the pilot stick location into an equivalent longitudinal cyclic command which is referred to as u_{δ} whereas $u_{\delta_{eff}}$ is the effective longitudinal cyclic command that is passed on to the flight control system. In most cases these two variables are exactly the same and a large difference in their values, for a significant duration of time, is not desirable. For example, $u_{\delta_{eff}}$ will be different from u_{δ} in case envelope protection is authorized to automatically remove control inputs exceeding critical values. If the difference between u_{δ} and $u_{\delta_{eff}}$ is large then it may cause a significant deviation in vehicle response from pilot expectations.

3.7.4 Post-inceptor command shaping for high frequency limit protection

In figure 36 the hub moment response is presented again with just the NTG limit protection system and unlike the results in figure 35 a sharp upper limit violation is

observed between 35 and 37 seconds of the maneuver.

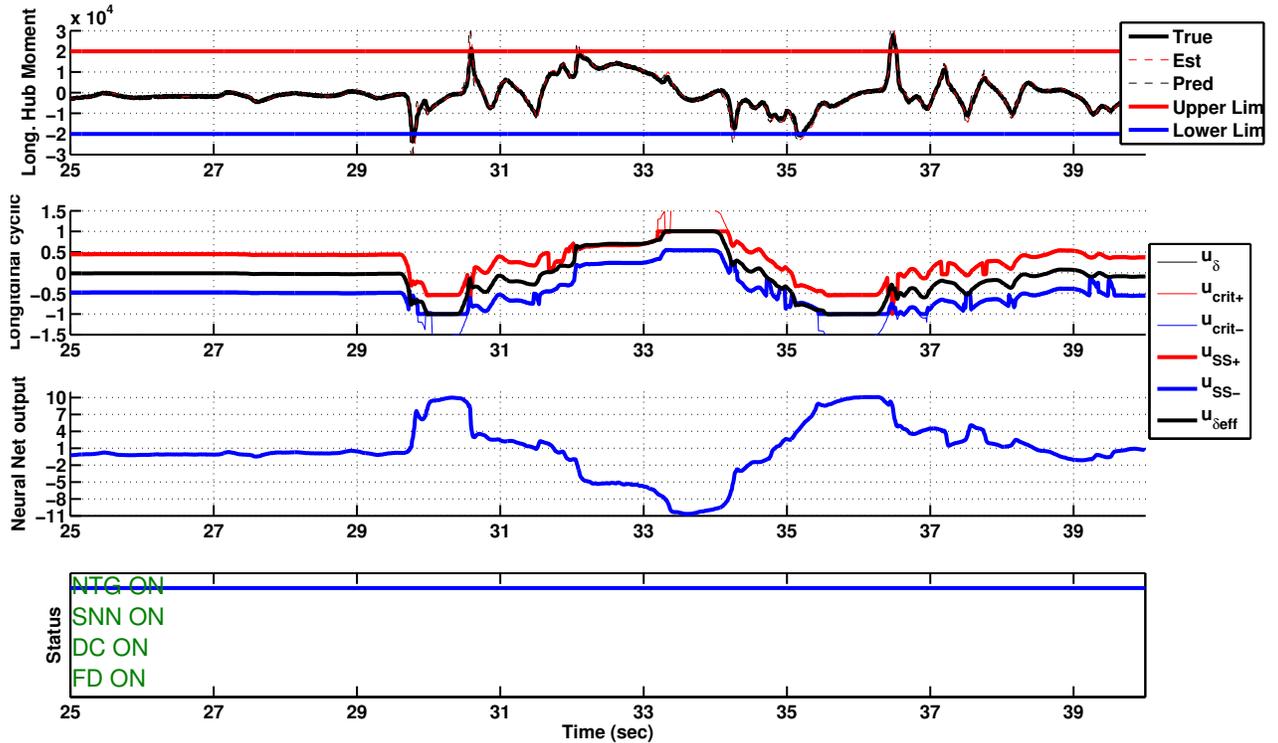


Figure 36: Swoop maneuver with hub moment limit protection- Nominal configuration

This is because in some instances, while the inceptor was stationary or moving slowly within limits, the limit cue module (with its prediction) placed the constraint inside of the inceptor. That is, the inceptor was steady, but the softstop moved through it because the aircraft was moving beyond its limit boundary. However, because the inceptor has its own two degree of freedom dynamics, the net force applied by the softstop is a forcing function that a few hundredths perhaps a tenth of a second to accelerate the inceptor and push it back within the limit constraints. Meanwhile, without moving, the inceptor is overriding the softstop cue. In other words, the limit dynamics are faster than the physical dynamics of the sidestick-limb system. One potential solution is the application of both tactile cues for the pilot and some post-inceptor command restraint shaping to deal with the high frequency limit protection that cannot be addressed using by the stick.

Dynamic overshoot compensation (DC) and frequency distribution (FD) method [22] are two schemes that have been implemented within the OPLP for this purpose. The dynamic overshoot compensation uses a cut-off frequency to extract the high frequency content of the softstop positions. The dynamic overshoot compensation does not change or modify the softstop locations commanded by the envelope protection system but automatically removes the high frequency component from the pilot command resulting in a new effective command, i.e,

$$u_{\delta_{eff}} - u_{\delta} = \text{High frequency correction} \quad (133)$$

The hub moment response for the sloop maneuver using the NTG based hub moment limit protection system and dynamic overshoot compensation is presented in figure 37.

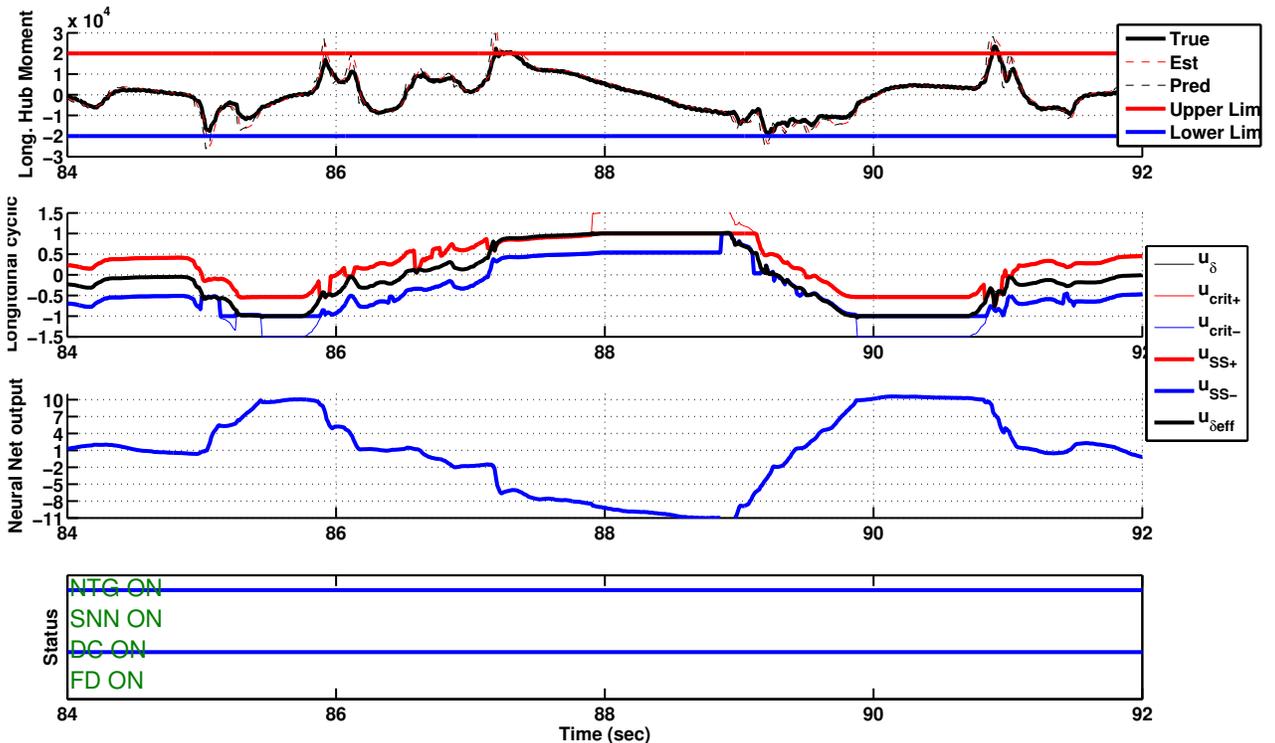


Figure 37: Sloop maneuver with hub moment limit protection and DC compensation- Nominal configuration

Upon closer inspection of results in figure 37, the DC high frequency limit protection

scheme is observed to modify the pilot longitudinal cyclic control commands during time intervals 86-87, 87-88 and 90-92. The longitudinal cyclic control channel activity during these time segments are presented in figure 38. Notice how the DC compensation scheme causes the effective longitudinal cyclic control input ($u_{\delta_{eff}}$) to follow the prescribed critical control (u_{crit+}) more closely than the original pilot control input (u_{δ}) thereby preventing hub moment upper limit violations.

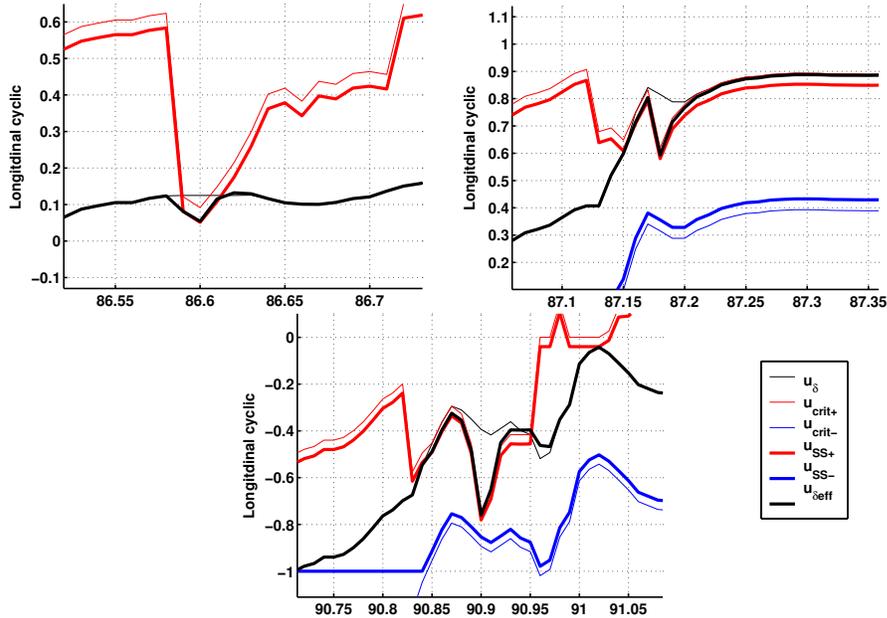


Figure 38: Post-inceptor command shaping using dynamic overshoot compensation-Nominal configuration

Frequency distribution method on the other hand filters the softstop locations into two components based upon a cut-off frequency. Only the low frequency component is used to set the softstop locations on the active sidestick whereas the high frequency component is automatically subtracted from the pilot commands resulting in the condition of equation 133. Therefore, the main difference between the FD and DC high frequency limit protection schemes is the way it effects the positioning of softstops. While the DC compensation does not modify the softstop location prescribed by the envelope protection module, the FD compensation scheme only allows the slow portion of the critical control position to set the softstop locations.

The hub moment response, without tactile cueing for the sloop maneuver when the vehicle is in heavy configuration, is presented in figure 39. Comparing the results

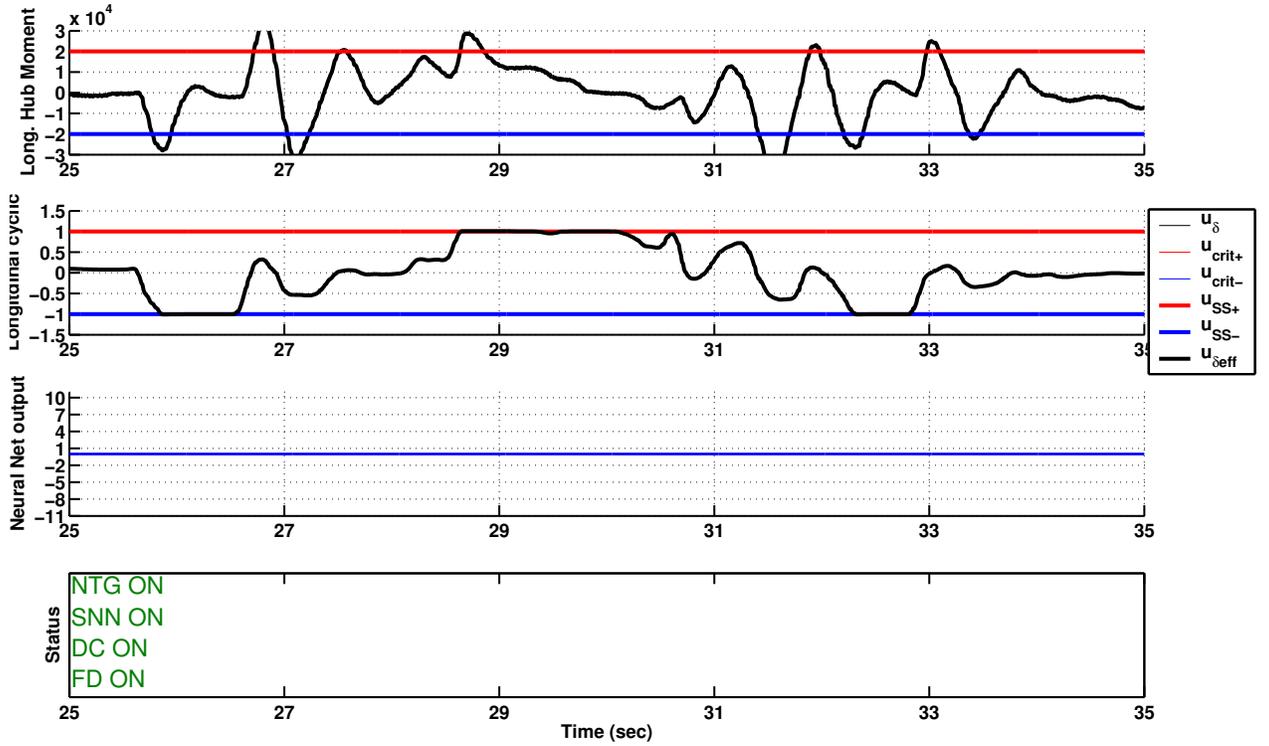


Figure 39: Sloop maneuver without hub moment limit protection- Heavy configuration

for hub moment response without tactile cueing presented in figures 34 and 39, it is observed that more severe limit boundary violations occur in the heavier configuration. Now, the exact same form of the adaptive estimate (see equation 128) used for evaluating the proposed approach in the nominal configuration is used for hub moment limit protection in the heavy configuration. The sloop maneuver hub moment response, for the heavy vehicle configuration with tactile cueing, are presented in figures 40, 41 and 42.

In figure 40 only the NTG based hub moment limit protection system is active. In figure 41 NTG based hub moment limit protection system is active with dynamic overshoot compensation. Finally, for the sloop maneuver results shown in figure 42 the NTG based limit protection system with frequency distribution method is active.

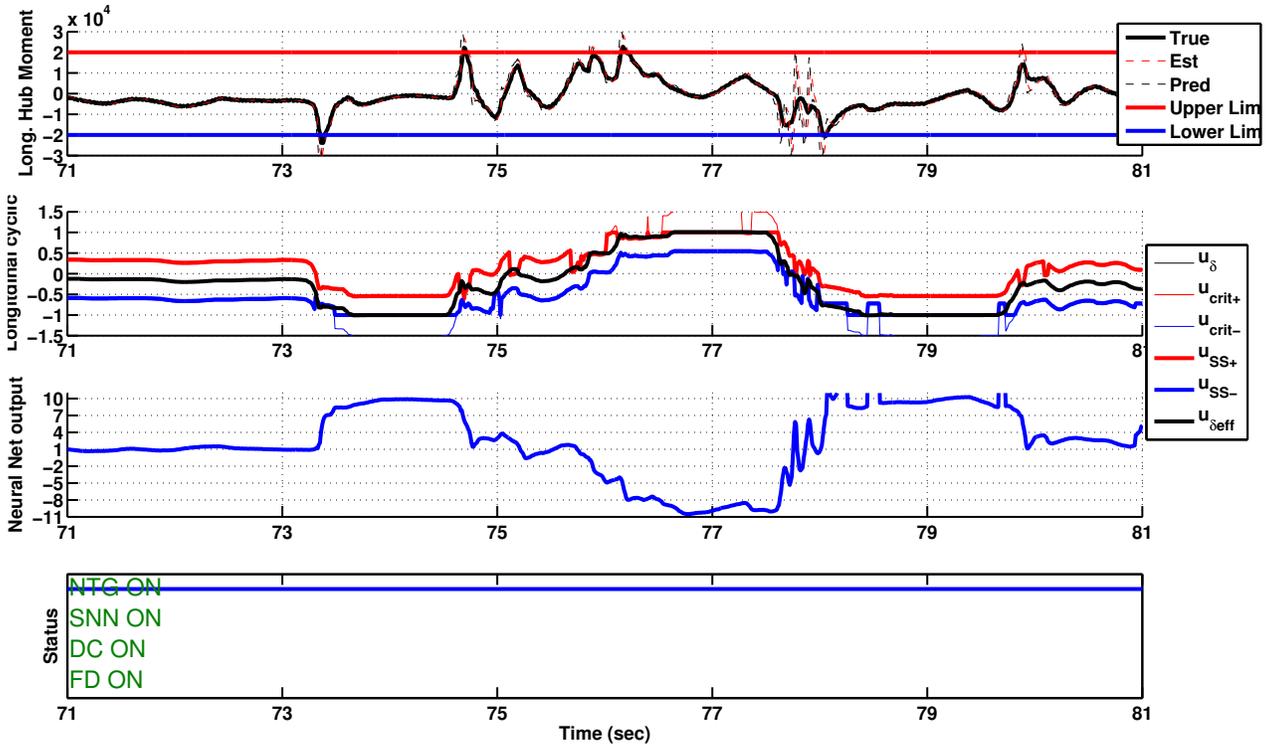


Figure 40: Swoop maneuver with hub moment limit protection- Heavy configuration

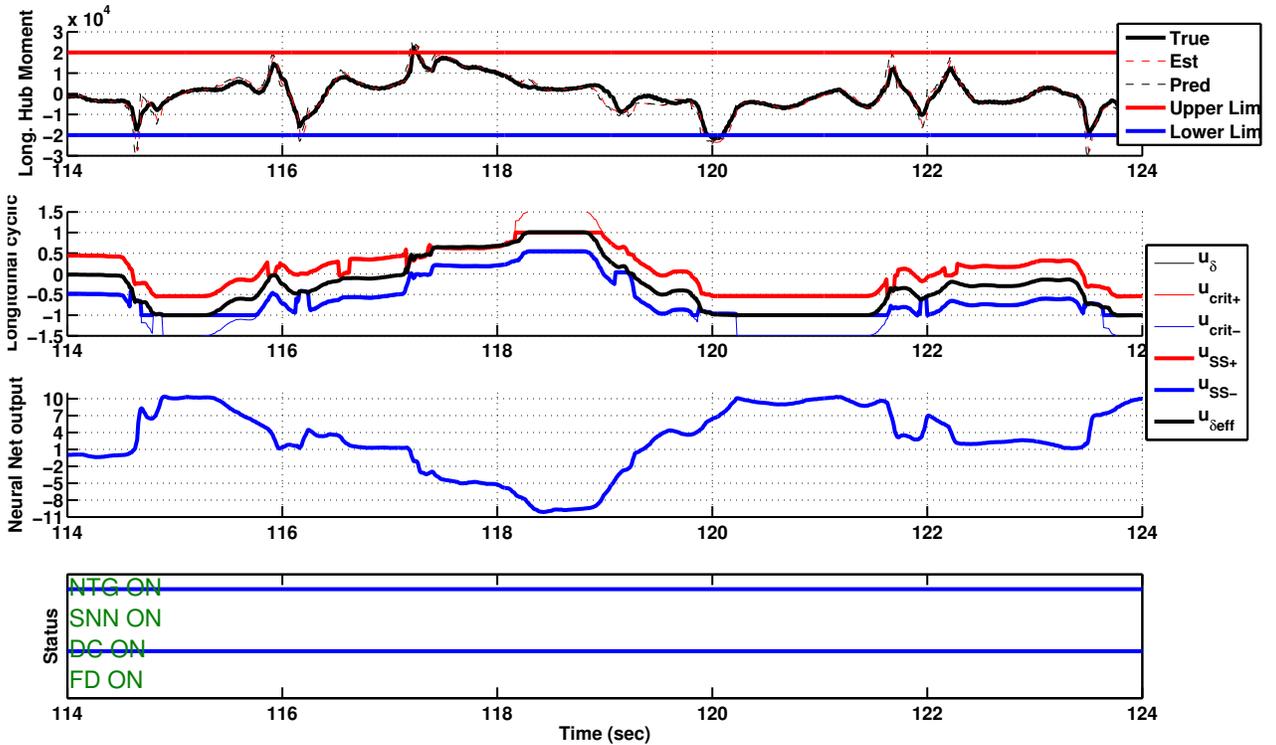


Figure 41: Swoop maneuver with hub moment limit protection and DC compensation- Heavy configuration

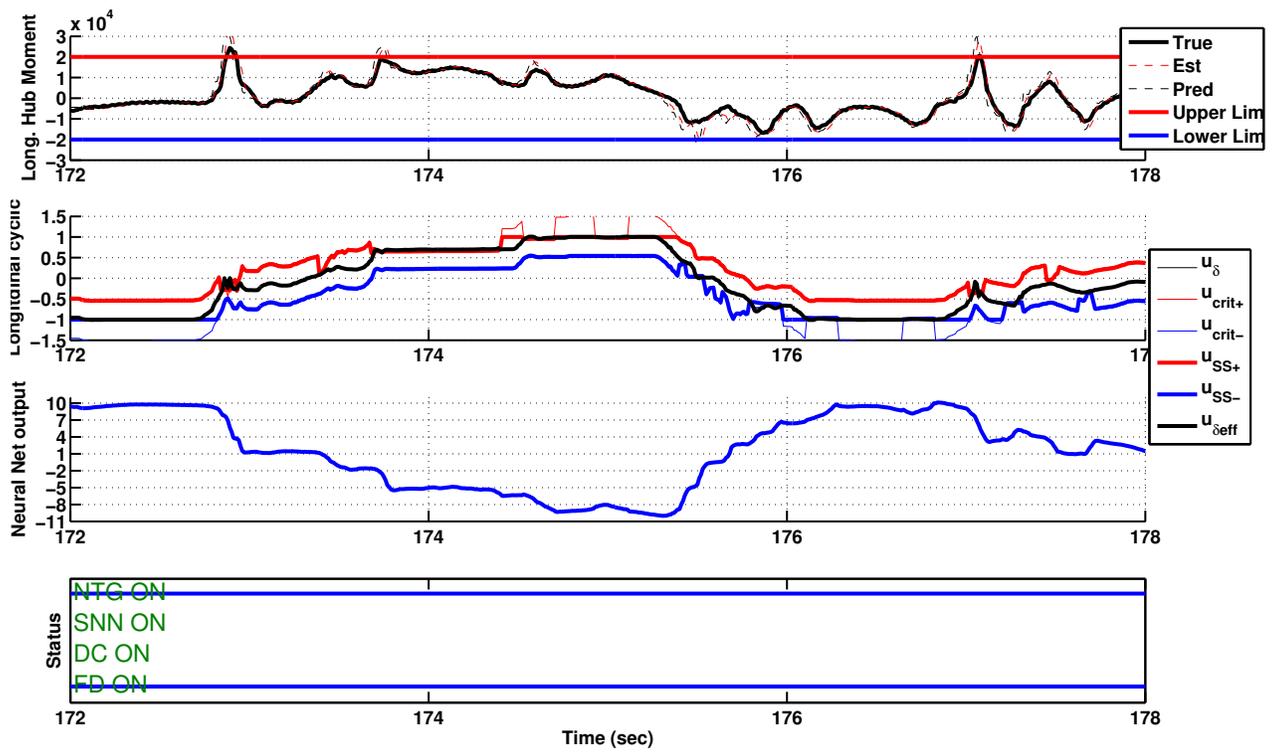


Figure 42: Sloop maneuver with hub moment limit protection and FD compensation- Heavy configuration

Notice that, even though the vehicle configuration has changed, the NTG based tactile cueing is able to successfully maintain the hub moment response within the prescribed limits. This is because the adaptive SHL-NN present in the adaptive estimate of limit parameter dynamics allows it to capture modeling uncertainties resulting from change in the vehicle configuration. As mentioned previously, DC [see figure 41] and FD [see figure 42] high frequency limit protection schemes improve upon the pure NTG based hub moment limit protection [see figure 40] using post-inceptor command shaping to compensate for limitations due to stick dynamics. The effect of DC compensation scheme on the pilot commands in the longitudinal cyclic control channel has already been analyzed [see figure 38]. Similarly in figure 42, the FD scheme is observed to modify the pilot longitudinal cyclic control commands during time intervals 172-173, 175-176 and 177-178. The longitudinal cyclic control channel activity during these time segments are presented in figure 43. Notice how the FD compensation scheme

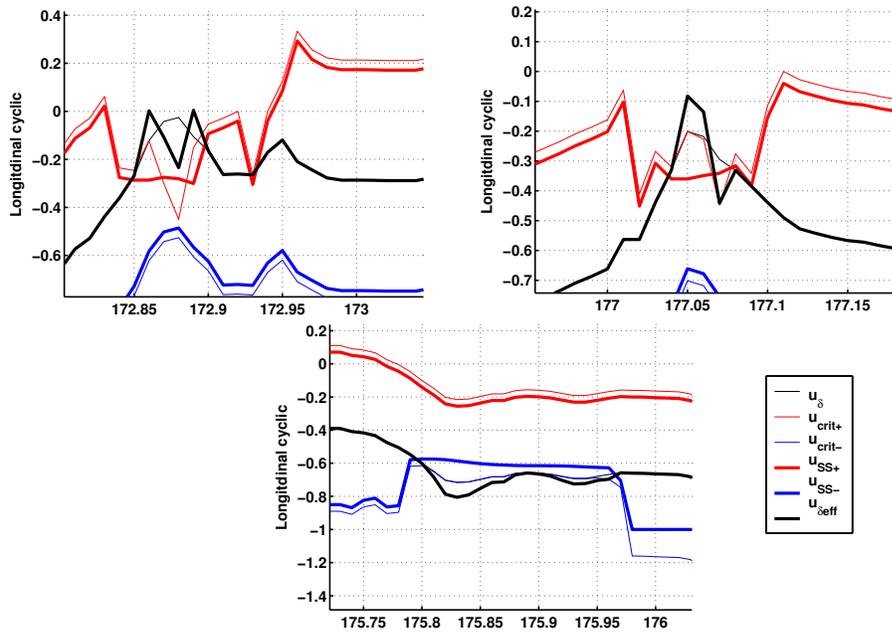


Figure 43: Post-inceptor command shaping using frequency distribution scheme-Heavy configuration

uses the slower component of the prescribed critical control position to determine the softstop location for tactile cueing. At the same time, FD compensation scheme

modifies the original pilot control input (u_δ) such that effective longitudinal cyclic control input ($u_{\delta_{eff}}$) stays within the prescribed critical control (u_{crit+}) thereby preventing hub moment upper limit violations. By using only the slower component of the critical control position to set the softstop locations, the pilot perceives a steady softstop rather than an oscillating control boundary that may reflect either fast limit parameter dynamics or stick dynamic limitations. Without this frequency separation, envelope protection for systems with fast limit parameter dynamics and comparatively slower stick dynamics could result in pilot-induced-oscillations.

The results demonstrate that the NTG based envelope protection system was successful in maintaining the hub moment response within the prescribed limits. Also, post-inceptor command shaping schemes (DC, FD) when utilized in augmenting the NTG based envelope protection system, were useful in providing high frequency limit protection.

3.8 Safety and performance evaluation

3.8.1 Quantitative evaluation using sloop maneuver results

The performance and safety benefits of the proposed hub moment limit protection system are studied using a number of sloop maneuvers. The NTG hub moment limit protection system is also evaluated against the recent nonlinear function response method based hub moment limit protection system. This alternative hub moment limit protection system has been also implemented within the OPLP and tested using the RIPTIDE environment. The details of the test setup and results from the evaluation of limit protection are reported in reference [49].

Maneuver safety is quantified using the *Integrated Hub Moment Limit Exceedance Factor*(IHMLEF) which is defined as the time integrated part of the hub moment when it exceeds the prescribed limits. Equation 134 represents how IHMLEF is

calculated:

$$IHMLEF = \begin{cases} \int |y_p - y_{lim}| dt & \text{if } |y_p| > |y_{lim}| \\ 0 & \text{otherwise} \end{cases} \quad (134)$$

The time the hub moment response lies within 10% of the limit boundaries is also noted and compared for each of the individual cases. Maneuver aggressiveness is quantified using maneuver time or time required to execute a swoop maneuver. Using the swoop maneuver specifications listed in table 3 the maneuver start and end times are identified. The maneuver time is then calculated as the time difference between the start and end times.

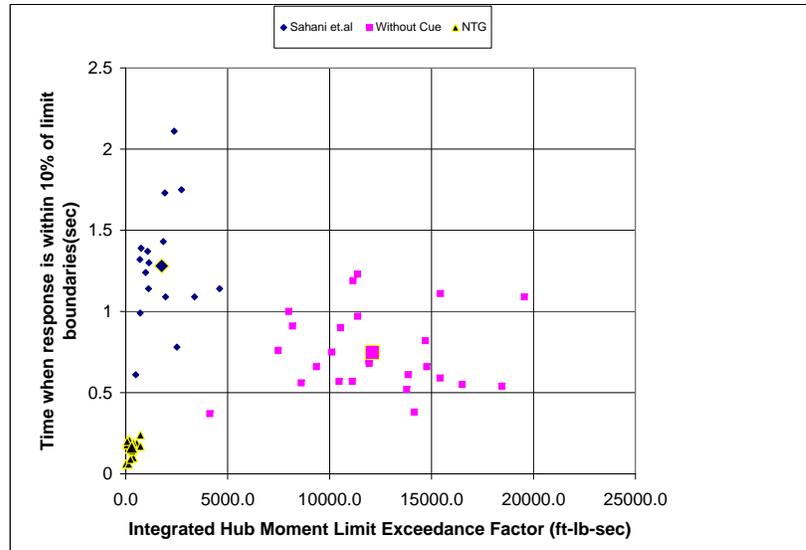


Figure 44: Time when limit parameter response is within 10% of the boundary vs Integrated hub moment limit exceedance factor.

The data presented in figure 44 shows that the NTG based envelope protection method resulted in 97.6% reduction in the average value of integrated hub moment limit exceedance factor as compared to when no tactile cueing is present. This reduction in the average value of IHMLEF using NTG cueing is an additional 12.17%

lower than the average value reported for nonlinear function response method.

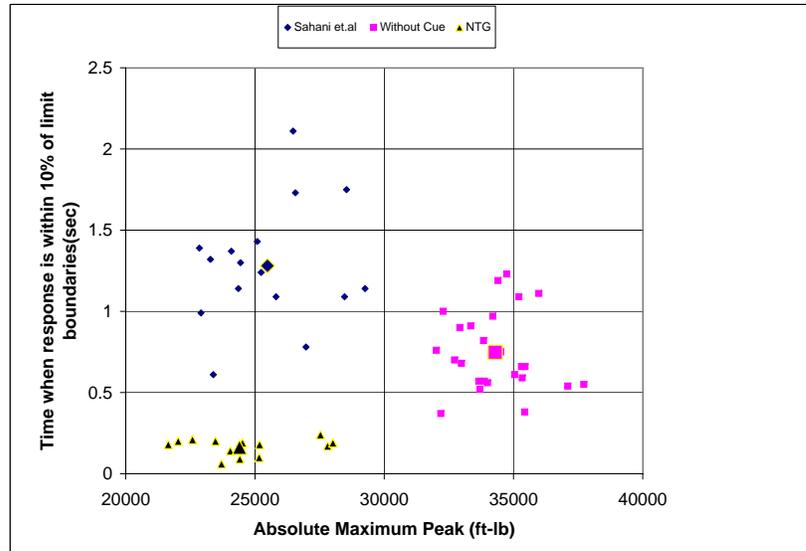


Figure 45: Time when limit parameter response is within 10% of the boundary vs maximum absolute peak.

The integrated hub moment limit exceedance factor and maximum absolute hub moment peak are plotted against the average time the hub moment response lies within 10% of the limit boundaries in figures 44 and 45, respectively. Notice that the average time the hub moment response lies within 10% of the limit boundaries increases from 0.7476 seconds to 1.28 seconds when nonlinear function response method is used. However, this average value reduces to 0.16 seconds when NTG based cueing is utilized. This behavior is surprising for the following reasons-

- The average maneuver time with NTG hub moment limit cueing is the lowest among all cases. This essentially means that pilot was most aggressive and experienced when executing the maneuver with NTG based hub moment limit protection.

- The average integrated hub moment exceedance factor during the NTG hub moment limit protection evaluations is the lowest. Therefore, the maneuver was executed the safest with NTG based hub moment limit protection.
- The average maximum absolute hub moment peak during the NTG hub moment limit protection evaluations is very close to prescribed limits. This result demonstrates that the NTG based hub moment limit protection system allowed the pilot to effectively utilize the prescribed operational envelope.

The difference and apparent inconsistency in the average time limit parameter response resides near 10% of the limit boundaries is due to the difference in envelope protection philosophy between the NTG and nonlinear function response method. The control limit in the NTG approach is an approach for informing the pilot the limits on vehicle aggressiveness with respect to a particular limit parameter. In other words, it warns the pilot through softstops of approaching the limit boundary aggressively. Therefore, the pilot reaches the limit boundary gradually and stays near the boundary only for a short duration. On the other hand, the nonlinear function response method outputs stick constraints that correspond to the limit boundaries. Hence, the pilot is warned only when the response is very close to the limit boundary which effectively results in increasing the time the limit parameter response resides near the limit boundaries.

In figures 46, 47 the maneuver time is plotted against the IHMLEF and maximum absolute peak respectively. Notice that, using nonlinear function response method for cueing results in a 25% reduction in the average value of the maximum absolute hub moment response. Using the NTG based approach reduces this average value of maximum absolute hub moment response by an additional 3%. This reduction in the average absolute maximum of hub moment response together with the significantly lower value of IHMLEF and average time near limit boundary clearly demonstrates increased safety of the NTG based hub moment limit protection system. Furthermore,

it is observed that while the average maneuver time *increased* with nonlinear function response based cueing, the average maneuver time *decreased* when NTG based cueing was used. This is due to the fact that the test-pilot, surprisingly, was more aggressive while performing the swoop maneuver with NTG based cueing. Pilot aggressiveness may have increased because of increased confidence and or familiarity with the test maneuver.

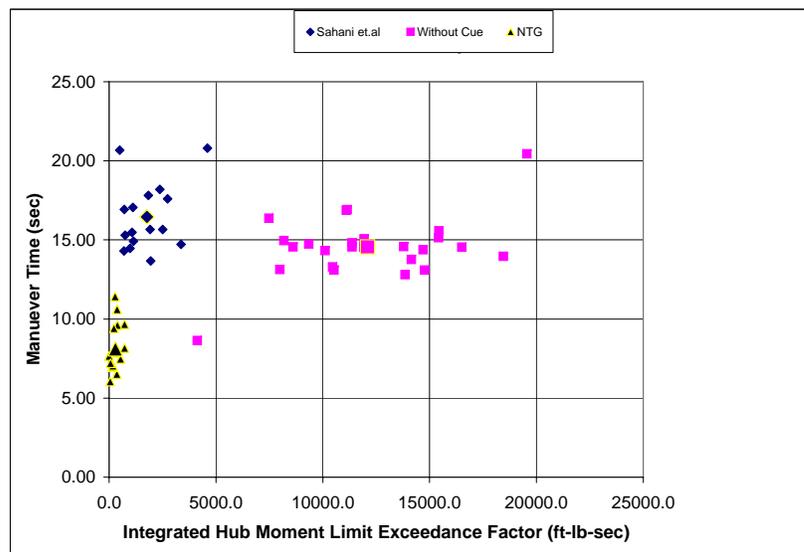


Figure 46: Maneuver time vs Integrated hub moment limit exceedance factor

Figure 48 compares the velocity and attitude variations during the swoop maneuver with and without NTG cueing. Notice that the velocity responses for swoop maneuver with NTG cueing are much sharper and quicker. Therefore, in summary the NTG based cueing significantly improved safety of the swoop maneuver by reducing the magnitude and duration of the hub moment limit violations. Furthermore, the NTG based cueing enabled the test-pilot to execute the swoop maneuver more aggressively and safely.

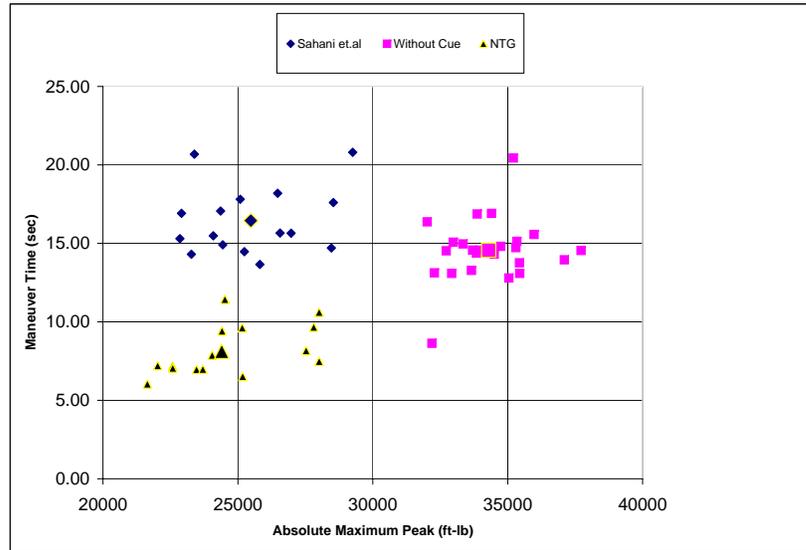


Figure 47: Maneuver time vs maximum absolute peak

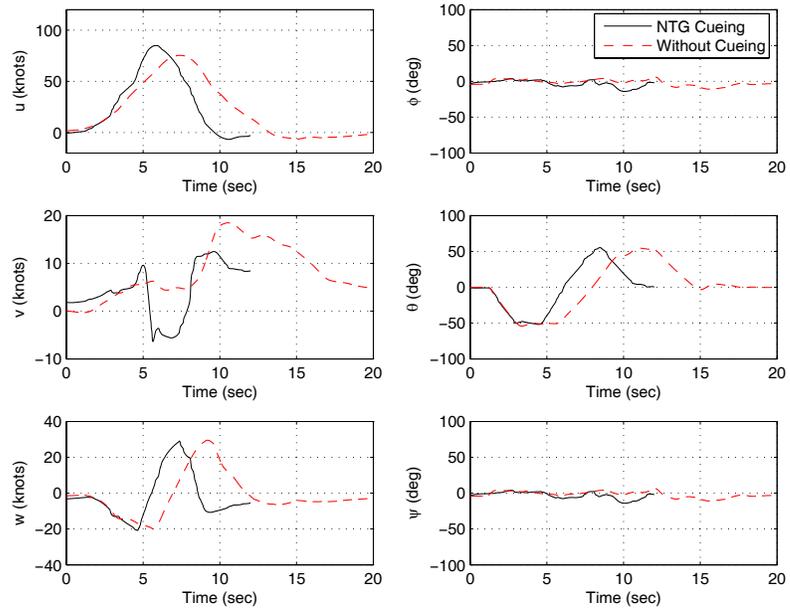


Figure 48: Sloop maneuver velocity and attitude responses with and without NTG cueing

3.8.2 Qualitative evaluation of NTG and nonlinear function response method

Both the nonlinear function response method and NTG based envelope protection algorithms cue the pilot of approaching the limit boundary by placing softstops in the control channel. The algorithms determine the softstop locations by computing the critical control position or control limit corresponding to the limit parameter. However, they differ in their method of control limit calculation. The differences between these two competing approaches, with their inherent advantages and or disadvantages are outlined below.

1. Limit parameter response modeling:

- **Nonlinear function response:** Limit parameter response modeled using nonlinear response functions identified during off-line analysis.
 - **Advantages:** The model is static and does not adapt to minimize error between model response and true limit parameter response. However, model is functionally determinate and therefore, easier to analyze.
 - **Disadvantages:** The identification of the nonlinear response functions requires generating large amounts of simulation or flight test data. In case, simulation data is used for identification the simulation model must be a good approximation of vehicle dynamics. On the other hand, when flight test data is used for model identification, generating data near envelope boundaries may not always be feasible.
- **NTG:** An adaptive estimate of limit parameter dynamics is generated using an approximate linear model augmented with an adaptive single hidden layer neural network.

- **Advantages:** Since the adaptive estimate is generated using on-line training of the neural network weights there is no need to generate large amounts of simulation or flight test data. With the appropriate choice of learning rates and adaptation law the adaptive estimate of limit parameter dynamics has the ability to be applicable to a wide range of flight and vehicle configurations.
- **Disadvantages:** Analyzing or predicting model performance is almost impossible because the system continuously adapts the weights of the SHL-NN so as to minimize limit parameter estimation error. Also, the choice of learning rates for weight adaptation is not readily available from off-line analysis and must be obtained from off-line simulation evaluations.

2. Computation of control limit or critical control position: Both the nonlinear function response and NTG method rely upon the limit parameter response model to estimate the control limits. While the nonlinear function response method uses a static input-output model of the limit parameter response, the NTG based approach utilizes an adaptive dynamic estimate of limit parameter dynamics for computing control limits.

- **Nonlinear function response method:** The control limit is computed using a conservative estimate of the control margin. As shown in figure 4, the control margin is the difference between the critical control position and the current control position. By definition the upper and lower control margin correspond to the upper and lower limits respectively. The upper(lower) control margin is estimated by dividing the difference of the upper(lower) limit boundary and the *maximum(minimum)* unforced limit parameter response with the maximum value of control sensitivity.

- **Advantages:** The control margin calculation proposed by Sahani et.al is more computationally efficient than other more accurate methods of estimating control margins such as that used in the peak response estimation algorithm [13]. Also, in case the pilot over-rides the limit the control margin calculation provides a relative estimate of how far the limit the vehicle is operating.
- **Disadvantages:** The estimation of control margin requires additional neural networks to represent nonlinear response functions such as minimum and maximum values of unforced limit parameter response and also maximum value of control sensitivity.
- **NTG:** The NTG based approach uses the adaptive estimate of limit parameter dynamics to estimate control limits. The approach utilizes the area norm [see equation 67] of the optimal control solution that takes the limit parameter response to the limit boundary, while minimizing the objective function in equation 63, as the control limit.
 - **Advantages:** The approach utilizes Nonlinear Trajectory Generator package for obtaining real-time solution to the optimal control problem. The approach allows the envelope protection designer to impose a desired level of aggressiveness by choosing the control weighting term in objective function appropriately.
 - **Disadvantages:** Calculating the control limit requires solving constrained nonlinear optimal control problem in real-time. Even though using the Nonlinear Trajectory Generator package significantly reduces the computational cost of solving the optimal control problem, it nevertheless imposes considerable burden on the computing resources of the system.

3.8.3 Analysis of nonlinear function response method and NTG based limit protection method using linear spring-mass-damper

The details regarding the application of nonlinear function response method to linear spring-mass-damper system are given in appendix C. The nonlinear function response method provides a conservative estimate of control margins as given below:

$$\tilde{\Delta}u_{lim,upper} = \frac{y_{lim,upper} - \max[Q(x, t)]}{\max[H(x, t)]}$$

$$\tilde{\Delta}u_{lim,lower} = \frac{y_{lim,lower} - \min[Q(x, t)]}{\max[H(x, t)]}$$

where $\tilde{\Delta}u_{lim,upper}$, $\tilde{\Delta}u_{lim,lower}$ are the upper and lower control margins respectively, $\max[H(x, t)] = \frac{1}{\omega_d} \exp(-\zeta\omega_n t_{2,max}) \sin(\omega_d t_{2,max})$. An upper limit of 5 meters is im-

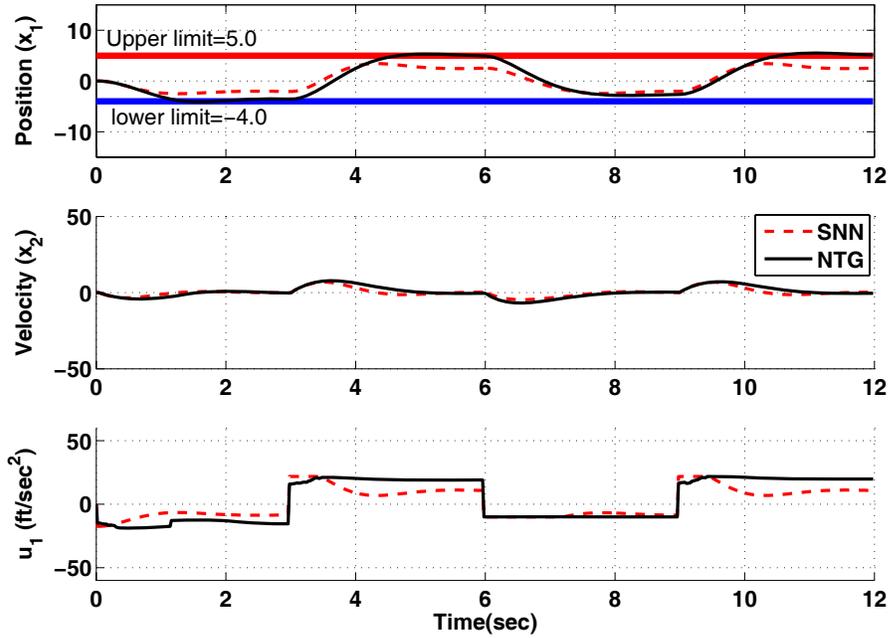


Figure 49: Spring-mass-damper limit parameter response comparison with NTG and smoothing function ($W = 5.0$) and nonlinear function response method based envelope protection

posed on the position response of the spring-mass-damper system given in equation 80. The limit parameter response obtained using the two different envelope protection methods is shown in figure 49. As shown in figure 49 the nonlinear function response method results in slightly more conservative limit parameter response.

CHAPTER IV

REACTIONARY APPROACH FOR AUTOMATIC ENVELOPE PROTECTION

In this chapter a new automatic envelope protection method for autonomous unmanned aerial vehicles is proposed and developed. The method uses the adaptive estimate of limit parameter dynamics for prediction of envelope violation. The proposed approach is referred to as the reactionary automatic envelope protection method.

4.1 Methodology

The idea behind the development of the reactionary automatic envelope protection method differs significantly from the existing methods. Recent envelope protection methods, applied to both manned and unmanned systems, rely mostly on translating the information about the limit boundary value into the control channel in the form of critical control position [see figure 4]. Almost all the existing approaches achieve this by using either a static or an adaptive model for the limit parameter. The emphasis was not only on the accuracy of the control boundary calculation but also on maximizing the prediction horizon of the approach. However, maximization of prediction horizon is an important issue in the design of envelope protection systems for manned vehicles where it may be necessary to inform the pilot regarding impending limit boundary violations as early as possible. This lead time available from the prediction is the key to an effective manned envelope protection system because it helps to compensate for delays associated with pilot response. But, in the case of unmanned autonomous aircraft, the flight control system can respond instantaneously to the command/control corrections from an envelope protection system. Hence,

maximizing the prediction horizon is not necessary and lead time available from finite time based prediction is sufficient for envelope protection.

4.2 Framework

The reactionary automatic envelope protection method involves three important steps that work together in maintaining the limit parameter response within the confines of the envelope-

1. Predict envelope violation using finite time horizon prediction of the estimate of limit parameter response.
2. Prescribe a safe-response-profile for the limit parameter response.
3. Compute command/control corrections that will force the true limit parameter response to track the safe-response-profile near the envelope boundary for envelope protection.

The reactionary envelope protection method, as described above, is similar to obstacle avoidance [48] as shown in figure 50. In obstacle avoidance when an obstacle is detected in the path of the vehicle a new safe trajectory is computed following which would avoid collision with the obstacle. In reactionary envelope protection, the obstacle is the well-defined limit boundary as shown in figure 50.

4.3 Step-1: Finite time horizon based prediction of envelope violation

The adaptive estimate obtained by augmenting an approximate linear model with an adaptive SHL-NN [see figure 7] is considered to be a reasonable estimate of the true limit parameter dynamics when the error in the estimate of limit parameter response is bounded, i.e,

$$\|e(t)\| = \|\hat{y}_p(t) - y_p(t)\| \leq \epsilon \quad (135)$$

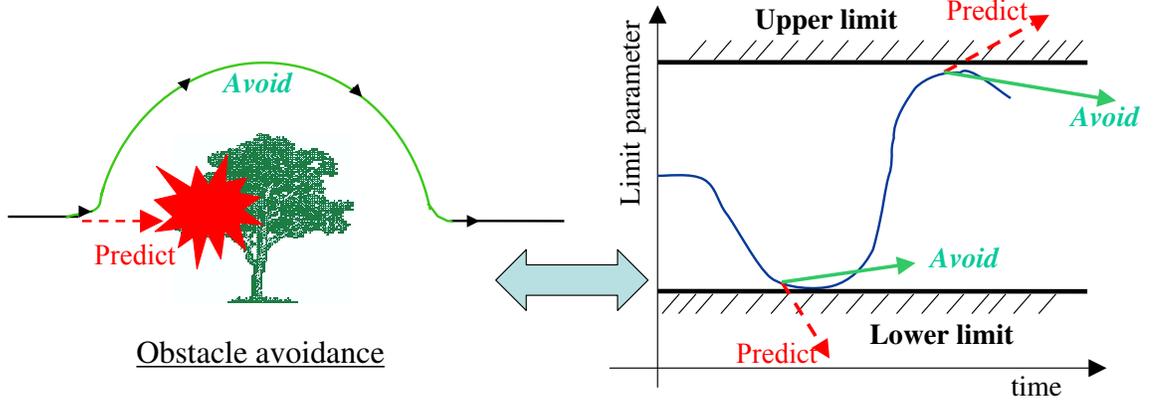


Figure 50: Reactionary envelope protection method and obstacle avoidance

This adaptive estimate of the limit parameter dynamics can be used to approximately determine the future response value of the limit parameter assuming that the command/control input remains at the present value. Let Δt_{fixed} be the fixed prediction horizon used in the future limit parameter response computation. When this future response value lies on or outside the prescribed limit parameter boundaries, then we say that an impending limit violation has been predicted. More precisely, equation 136 predicts a lower limit violation and equation 137 predicts an upper limit violation.

$$\hat{y}_p(t + \Delta t_{fixed}) \leq y_{lim}^{lower} \quad (136)$$

$$\hat{y}_p(t + \Delta t_{fixed}) \geq y_{lim}^{upper}. \quad (137)$$

The future limit parameter response value in equations 136 and 137 can be computed as follows,

$$\hat{y}_p(t + \Delta t_{fixed}) = \gamma^+(t, \hat{y}_p(t), \bar{u}, \Delta t_{fixed}) \quad (138)$$

where γ^+ refers to the forward time trajectory of the estimate of limit parameter dynamics. The control input is fixed at its current value during the computation of forward time trajectory. An important parameter that can be identified here is the *critical time of the limit parameter response*, Δt_{crit} . It is defined as the estimated time in which the limit parameter response will lie on the limit boundary assuming

it continues to evolve at the same rate. The critical time of the limit parameter response is a crucial variable in the generation of safe-response-profile near the limit boundary which is the step immediately following prediction of limit violation. The critical time, according to the definition, can be computed as follows-

$$0 \leq \Delta t_{crit} = \frac{y_{lim} - \hat{y}_p(t)}{\dot{\hat{y}}_p(t)} \leq \Delta t_{fixed}. \quad (139)$$

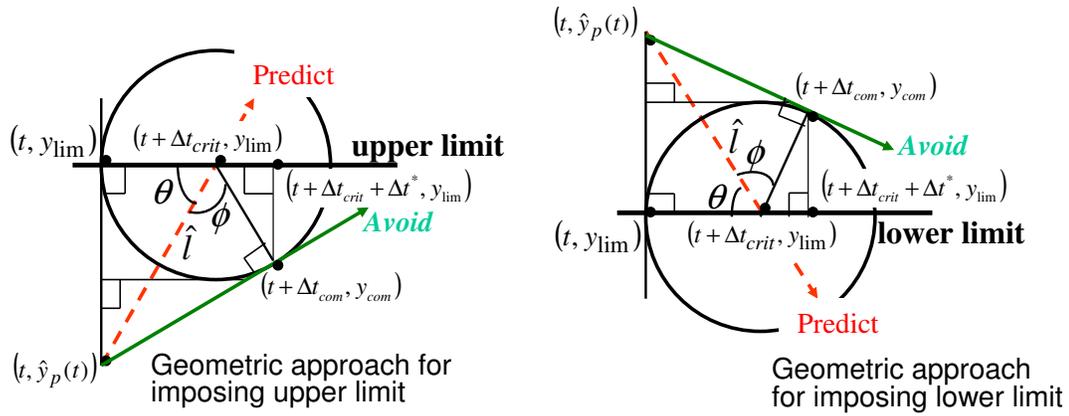
where y_{lim} is the value of the limit boundary where limit violation is predicted to occur.

$$y_{lim} = \begin{cases} y_{lim}^{lower} & \text{if equation 136 is satisfied} \\ y_{lim}^{upper} & \text{if equation 137 is satisfied} \end{cases} \quad (140)$$

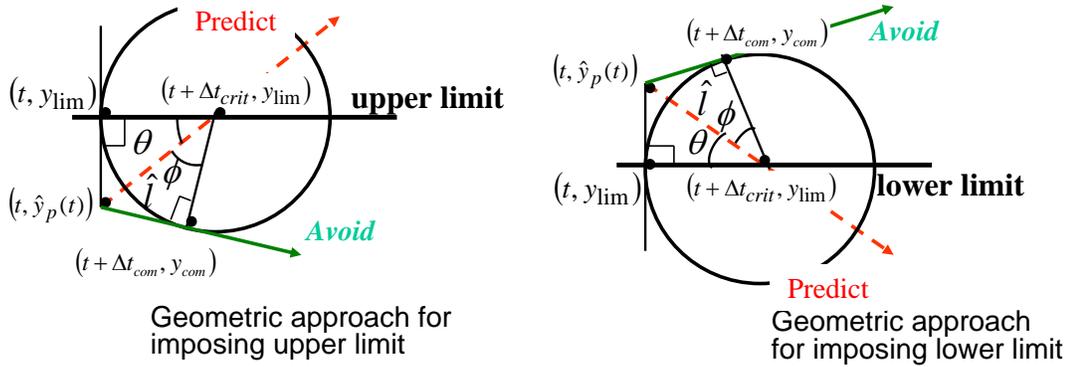
4.4 Step-2: Safe response profile

The next step following the prediction of envelope violation is to prescribe a safe-response-profile near the limit boundary. The safe-response-profile in a reactionary envelope protection architecture is the first step towards preventing an envelope violation and serves many purposes. Firstly, the safe-response-profile has to remain within the limit boundaries. Secondly, it must prevent large envelope violations from occurring. Finally, it is desirable that the safe-response-profile does not cause under-utilization of the OFE.

In the proposed method, the safe-response-profile for the limit parameter response is constructed by assuming the existence of an imaginary circular obstacle with center at $(t + \Delta t_{crit}, \hat{y}_p(t + \Delta t_{crit}))$ and radius Δt_{crit} . According to the definition of critical time, this center point lies on the limit boundary where violation is predicted [see figure 51]. Note that unless corrective action is taken, the limit parameter response is predicted to violate the envelope and the response is directed towards the center of the obstacle. The direction of the safe-response-profile is chosen to be along the *tangential* path avoiding collision with the imaginary obstacle. Following this will



$$(a) \|\hat{y}_p(t) - y_{lim}\| \geq \Delta t_{crit}$$



$$(b) \|\hat{y}_p(t) - y_{lim}\| \leq \Delta t_{crit}$$

Figure 51: Safe response profile for envelope protection near limit boundary

force the response to move away from the limit boundary at each instant in time [see figure 51].

Let the coordinates of the point of contact of the tangent, avoiding the imaginary obstacle, be $(t + \Delta t_{com}, y_{com})$. Then the unknown variables $\Delta t_{com}, y_{com}$ can be obtained as functions of θ, \hat{l}, ϕ as shown in figure 51. Using basic Euclidean geometry and trigonometric identities the following relations can be obtained-

$$\theta = \arctan \frac{\|\hat{y}_p - y_{lim}\|}{\Delta t_{crit}}, \quad (141)$$

$$\phi = \arccos \frac{\Delta t_{crit}}{\hat{l}}, \quad (142)$$

The estimated distance of the limit parameter response from the center of the obstacle, represented as \hat{l} , is calculated as:

$$\hat{l}(t) = \sqrt{(y_{lim} - \hat{y}_p(t))^2 + \Delta t_{crit}^2} \quad (143)$$

where y_{lim} is limit parameter boundary value obtained from equation 140. The value of Δt^* , shown in figure 51 depends on the available limit margin and is given by equation 144.

$$\Delta t^* = \begin{cases} \Delta t_{crit} \cos(\pi - \theta - \phi) & \text{if } \|\hat{y}_p(t) - y_{lim}\| \geq \Delta t_{crit} \\ \Delta t_{crit} \cos(\theta + \phi) & \text{if } \|\hat{y}_p(t) - y_{lim}\| < \Delta t_{crit} \end{cases} \quad (144)$$

Also,

$$(y_{com} - y_{lim})^2 + \Delta t^{*2} = \Delta t_{crit}^2 \quad (145)$$

and therefore,

$$y_{com} = \begin{cases} y_{lim} + \sqrt{\Delta t_{crit}^2 - \Delta t^{*2}} & \text{if } y_{lim} = y_{lim}^{lower} \\ y_{lim} - \sqrt{\Delta t_{crit}^2 - \Delta t^{*2}} & \text{if } y_{lim} = y_{lim}^{upper} \end{cases} \quad (146)$$

The obstacle avoidance based approach presented in equations 141-146 can be used only to determine the direction of the safe-response-profile. Also, the approach outlined can be invoked only when the estimate of limit parameter response is within the envelope when limit boundary violation is predicted, i.e, $y_{lim}^{lower} \leq \hat{y}_p(t) \leq y_{lim}^{upper}$. However, the safe-response-profile must also be designed in the event the limit parameter response strays outside the envelope. This could occur in situations where the safe-response-profile is not perfectly tracked. In the event that the limit parameter response strays into the region outside envelope, $\hat{y}_p < y_{lim}^{lower}$ or $\hat{y}_p > y_{lim}^{upper}$, then safe-response-profile is prescribed to bring the limit parameter response back inside the envelope.

4.4.1 Safe response profile for $r = 1$

For a limit parameter with relative degree equal to one, the safe-response-profile must be at least continuous, i.e, $y_s(t) \in C^0$. Therefore, the time derivative of the safe-response profile (\dot{y}_s) is computed using the following conditional equation-

$$\dot{y}_s(t) = \begin{cases} \frac{y_{com} - \hat{y}_p}{\Delta t_{com}} & \text{if } y_{lim}^{lower} \leq \hat{y}_p \leq y_{lim}^{upper} \\ 0 & \text{if } |\hat{y}_p| \leq |y_{lim}| \text{ and } |\hat{y}_p - y_{lim}| \leq \epsilon_b \\ c_1 > 0 & \text{if } \hat{y}_p < y_{lim}^{lower} \\ c_2 < 0 & \text{if } \hat{y}_p > y_{lim}^{upper} \\ \dot{\hat{y}}_p(t) & \text{if no violation is predicted} \end{cases} \quad (147)$$

where,

$$\Delta t_{com} = \begin{cases} \Delta t_{crit} + \Delta t^* & \text{if } \|\hat{y}_p(t) - y_{lim}\| \geq \Delta t_{crit} \\ \Delta t_{crit} - \Delta t^* & \text{if } \|\hat{y}_p(t) - y_{lim}\| < \Delta t_{crit} \end{cases} \quad (148)$$

In equation 147, c_1 and c_2 are design constants to be chosen by the designer to bring the limit parameter response lying outside back inside the envelope. Furthermore, the time-derivative of safe-response profile goes to zero when the limit parameter response approaches very close to the limit boundary (ϵ_b) from within the envelope. Finally, the safe response variable y_s is updated as given in equation 149.

$$y_s(t) = \hat{y}_p(t - \Delta t_{sim}) + \dot{y}_s(t - \Delta t_{sim})\Delta t_{sim} \quad (149)$$

From equations 149 and 147, it can be deduced that the safe response profile variable, when no envelope violation is predicted, is same as that of the adaptive model response.

Remark 4.4.1. *In the present formulation the radius of the imaginary obstacle is not fixed but depends on the rate of estimated limit parameter dynamics, $\dot{\hat{y}}_p(t)$ [see equation 139]. If the dynamics are very fast close to the edge of the envelope this would translate into a smaller size obstacle. At first glance this may seem counter*

intuitive, however, note that the safe response profile for envelope protection is not based purely on the finite time prediction but also on the rate of the limit parameter dynamics. If the dynamics are such that $\Delta t_{crit} \rightarrow 0$ then,

$$\theta \rightarrow \frac{\pi}{2}, \quad (150)$$

$$\Delta t_{com} \rightarrow 0, \quad (151)$$

resulting in,

$$\dot{y}_s(t) \rightarrow \begin{cases} +\infty & \text{if } y_{lim} = y_{lim}^{lower} \\ -\infty & \text{if } y_{lim} = y_{lim}^{upper} \end{cases} \quad (152)$$

Hence, when $\Delta t_{crit} \rightarrow 0$, equation 152 shows that the envelope protection also mimics the fast behavior of the actual limit parameter dynamics.

4.4.2 Safe response profile for $r > 1$

Assumption 2.1.1 guarantees that the relative degree r of the limit parameter is well-defined and known. Let $y_s(t)$ represent the safe-response-profile of the limit parameter response near the limit boundary. Then application of reactionary automatic envelope protection method requires that the safe-response profile be a smooth trajectory which is at least $(r - 1)$ times differentiable.

However, the approach presented in figure 51 can only be used to determine the relative direction of the safe-response-profile with respect to the current projected path of the response. While this information can be utilized in many ways to generate a safe response profile that has desired smoothness properties, the following are two viable approaches:

1. Design a command filter whose input will be y_{com} and output will be

$$y_s(t), y_s^{(1)}(t), \dots, y_s^{(r)}(t)$$

2. Fit a B-spline curve of desired smoothness from the current position of limit parameter response (t, \hat{y}_p) to the point of contact of tangent $(t + \Delta t_{com}, y_{com})$.

4.5 Tracking safe response profile

Once the appropriate safe response profile $y_s(t)$ is generated, the next step is to provide corrections to the nominal control/command channel so that the true limit response (y_p) is able to track it for envelope protection. In this section one such correction is proposed that is based on the adaptive estimate of limit parameter dynamics. Further analysis of the error dynamics is used to show that the proposed correction will result in limit parameter dynamics tracking the safe response profile provided the adaptive element in the estimate of limit parameter dynamics, the SHL-NN, is able to cancel modeling uncertainty.

$$u_{corr} = \left[y_s^{(r)} - \left(\sum_{i=0}^{r-1} a_i y_s^{(i)} + \frac{\partial \hat{h}_r}{\partial u_p} u_p + \nu_{ad}(\bar{\mu}) - \nu_{dc} \right) - d_l(\hat{y}_p - y_s) \right] \left(\frac{\partial \hat{h}_r}{\partial u_p} \right)^{-1} \quad (153)$$

When this control correction is added to the original control input of the true limit parameter dynamics in equation 3, the following form is obtained.

$$\begin{aligned} y_p^{(r)} &= \hat{h}_r(y_p, y_p^{(1)}, \dots, y_p^{(r-1)}, u_p) + \Delta(\bar{\mu}) + \frac{\partial \hat{h}_r}{\partial u_p} u_{corr} \\ &= \sum_{i=0}^{r-1} a_i y_p^{(i)} + \frac{\partial \hat{h}_r}{\partial u_p} u_p + \Delta(\bar{\mu}) + y_s^{(r)} \\ &\quad - \left(\sum_{i=0}^{r-1} a_i y_s^{(i)} + \frac{\partial \hat{h}_r}{\partial u_p} u_p + \nu_{ad}(\bar{\mu}) - \nu_{dc} \right) - d_l(\hat{y}_p - y_s) \end{aligned} \quad (154)$$

where,

$$\begin{aligned} \Delta(\mathbf{x}, y_p, y_p^{(1)}, \dots, y_p^{(r-1)}) &\triangleq h_r(\mathbf{x}, y_p, y_p^{(1)}, \dots, y_p^{(r-1)}, u_p) \\ &\quad - \hat{h}_r(y_p, y_p^{(1)}, \dots, y_p^{(r-1)}, u_p) \end{aligned} \quad (155)$$

This equation can be further simplified by canceling the linear control terms and expanding the contribution of the linear dynamic compensator as given in equation 14.

$$\begin{aligned} y_p^{(r)} - y_s^{(r)} &= \sum_{i=0}^{r-1} a_i \left(y_p^{(i)} - y_s^{(i)} \right) + \mathbf{C}_l \eta + \left[d_l(\hat{y}_p - y_p) - d_l(\hat{y}_p - y_s) \right] \\ &\quad + (\Delta(\bar{\mu}) - \nu_{ad}(\bar{\mu})) \end{aligned} \quad (156)$$

The safe-response-profile tracking error dynamics, given in equation 156, is re-cast into state-space representation using the following definition for safe-response-profile tracking error vector,

$$E_s \triangleq \begin{bmatrix} e_s & e_s^{(1)} & \dots & e_s^{(r-1)} \end{bmatrix}^T \quad (157)$$

with,

$$e_s \triangleq y_p - y_s \quad \text{and} \quad \hat{e}_s \triangleq \hat{y}_p - y_s \quad (158)$$

where e_s is referred to as the safe-response profile tracking error and \hat{e}_s is called the estimate of the safe-response-profile tracking error. Also, $E \triangleq \begin{bmatrix} e & e^{(1)} & \dots & e^{(r-1)} \end{bmatrix}$. Using equations 3,10,158 and 156 the final form of the complete error dynamics can be written into the following matrix form,

$$\begin{bmatrix} \dot{E}_s \\ \dot{\eta} \\ \dot{E} \end{bmatrix} = \begin{bmatrix} \mathbf{A} & \mathbf{C}_l & \mathbf{0}_{(r,r)} \\ \mathbf{0}_{(p,r)} & \mathbf{A}_l & \mathbf{B}_l \mathbf{B}_{(1,r)}^T \\ \mathbf{0}_{(r,r)} & -\mathbf{B}_{(r,r)} \mathbf{C}_l & \mathbf{A} - d_l \mathbf{B}_{(1,r)}^T \end{bmatrix} \begin{bmatrix} E_s \\ \eta \\ E \end{bmatrix} + \begin{bmatrix} \mathbf{B}_{(r,r)} \\ \mathbf{0}_{(p,1)} \\ \mathbf{B}_{(r,r)} \end{bmatrix} (\Delta - \nu_{ad}) \quad (159)$$

where $\mathbf{0}_{(i,j)}$ is the zero matrix of dimension $\mathfrak{R}^{i \times j}$ and $\mathbf{B}_{(j,r)} \in \mathfrak{R}^r$ is a r - dimensional unit vector with its j^{th} element equal to one. Also, matrix \mathbf{A} is defined in equation 6. The stability matrix of the linearized error dynamics in equation 159 is Hurwitz and is driven by the modeling error $\Delta - \nu_{ad}$. Notice that the control correction proposed in equation 153 is such that the estimate of limit parameter dynamics \hat{y}_p will be asymptotically driven to the safe-response profile y_s . This can be shown by deriving the dynamics for the estimate of safe-response profile tracking error (\hat{e}_s) defined in equation 158. Plugging in the control correction into the adaptive estimate of limit parameter dynamics given in equation 10 we get,

$$\begin{aligned}
\hat{y}^{(r)} &= \sum_{i=0}^{r-1} a_i \hat{y}^{(i)} + \frac{\partial \hat{h}_r}{\partial u_p} u_p + \nu_{ad}(\bar{\mu}) - \nu_{dc} + \frac{\partial \hat{h}_r}{\partial u_p} u_{corr} \\
&= \sum_{i=0}^{r-1} a_i \hat{y}^{(i)} + \frac{\partial \hat{h}_r}{\partial u_p} u_p + \nu_{ad}(\bar{\mu}) - \nu_{dc} \\
&\quad + y_s^{(r)} - \left(\sum_{i=0}^{r-1} a_i y_s^{(i)} + \frac{\partial \hat{h}_r}{\partial u_p} u_p + \nu_{ad}(\bar{\mu}) - \nu_{dc} \right) - d_l (\hat{y}_p - y_s) \\
&= \sum_{i=0}^{r-1} a_i \left(\hat{y}_p^{(i)} - y_s^{(i)} \right) - d_l (\hat{y}_p - y_s)
\end{aligned}$$

The above equation can be rewritten into the state-space form as follows:

$$\dot{\hat{E}}_s = \left(\mathbf{A} - d_l \mathbf{B}_{(1,r)}^T \right) \hat{\mathbf{E}}_s \quad (160)$$

where $\hat{E}_s \triangleq \begin{bmatrix} \hat{e}_s & \hat{e}_s^{(1)} & \dots & \hat{e}_s^{(r-1)} \end{bmatrix}$. Equation 160 clearly shows that the control correction makes the dynamics of the estimate of safe-response profile tracking error asymptotically stable. Therefore, $\hat{y}_p \rightarrow y_s$. Now using the error vector definitions it can be shown that,

$$E_s = \hat{E}_s - E \quad (161)$$

The adaptive neural network is designed to guarantee that the error in the estimate of limit parameter dynamics (E) remains bounded. Also, this bound can be made sufficiently small as to guarantee that the adaptive estimate of limit parameter dynamics is a reasonable approximation of the true limit parameter dynamics. Therefore, assuming that a good adaptive estimate of limit parameter dynamics is available and the fact that control correction guarantees the asymptotic convergence of the estimate of safe-response profile tracking error to the origin, it can be concluded that \mathbf{E}_s will be bounded.

4.6 Linear simulation results

In this section, the proposed approach is applied to a linear first order system. The linear example will establish the ideas behind the generation of safe response profile

and will also demonstrate the ability of computed control correction to make the true response track the safe response profile close to the envelope boundary.

Consider a linear plant (eigenvalue -2) with first order dynamics given as in equation 162,

$$\dot{y}_p = -2y_p - 5u_p \quad (162)$$

A square wave input of amplitude 2 units and time period 20 seconds is applied to this plant. A lower limit of -2 is assumed and the linear approximate model is chosen with eigenvalue -1,

$$\dot{\hat{y}}_p = -\hat{y}_p - 3u_p \quad (163)$$

The parameter values chosen for the augmenting single hidden layer neural network are provided in table 7. The normalized neural net input vector is $\bar{\mu} = \begin{bmatrix} \frac{y_p}{5} & \frac{u_p}{2} \end{bmatrix}$. A simple static error feedback, with gain 4, is used instead of the linear dynamic compensator.

A comparison of the true plant response and model response, without reactionary envelope protection system, is shown in figure 52. The simulation time step is 0.02 seconds and the prediction horizon for the response is chosen to be 0.1 seconds.

Table 7: Reactionary envelope protection linear example- neural network design and learning rate parameters

N_{inp}	2	Output layer learning rate, Γ_m	8.0
N_{hid}	4	Hidden layer learning rate, Γ_n	0.4
N_{out}	1	Emod parameter, κ_ϵ	0.2
Basis function, σ	sigmoid $\frac{1}{1+e^{-a(x-c)}}$	Sigmoid parameters	$a = 1.0,$ $c = 0$

In figure 52, the model response and the true response are on top of each other. Also, since the prediction horizon is only 0.1 seconds it is difficult to distinguish the predicted future response from the true response. The design parameter c_1 is selected to be 0.3249 and ϵ_b is set at 0.1. Figure 53(a) shows the plant response with envelope

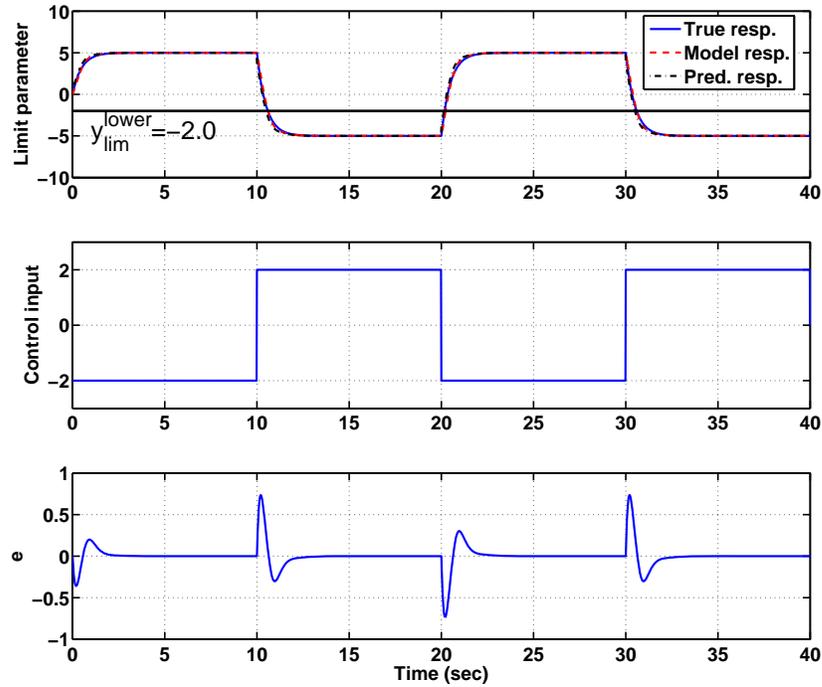
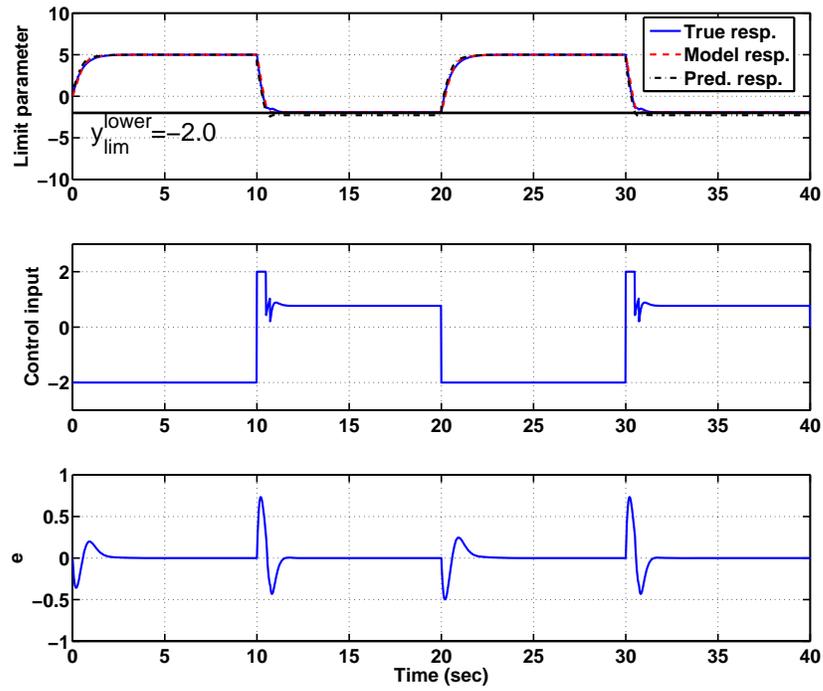


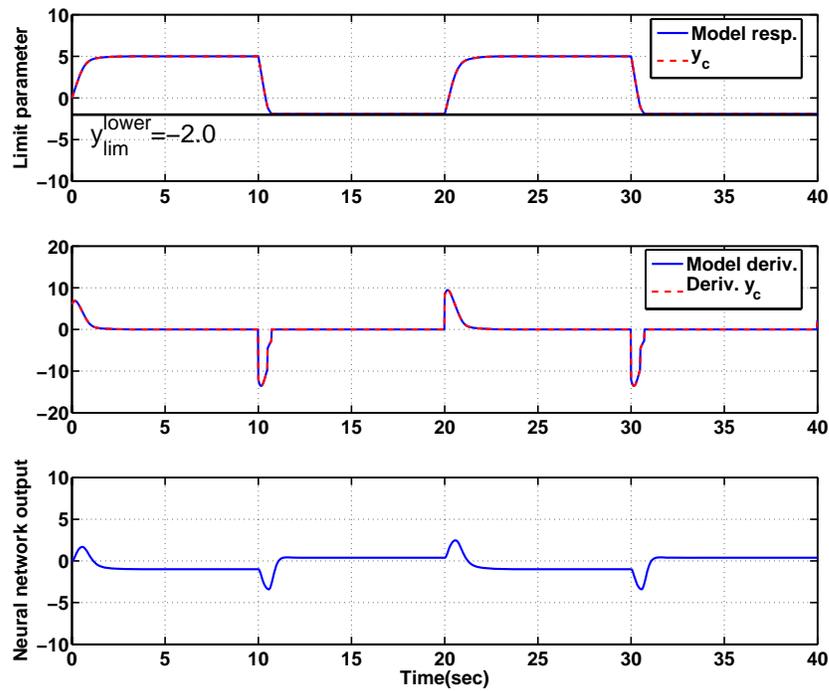
Figure 52: Linear example- Limit parameter response with reactionary envelope protection OFF

protection system ON. The response of the estimate of limit parameter dynamics and the finite time horizon prediction are also shown in figure 53(a). Notice that the plant response follows the model response closely along the envelope boundary. The control input and error, e , are plotted in subplots of figure 53(a).

The model response is compared with the safe response profile in figure 53(b). The neural network output is plotted within a subplot of figure 53(b). The model response and hence the true response follows the safe response profile close to the envelope boundary as expected and therefore the limit parameter response is maintained above its lower limit.



(a) Plant response



(b) Model response

Figure 53: Linear example- Limit parameter response with reactionary envelope protection ON

4.7 *GTMax integrated simulation and flight testing architecture*

The GTMax [see figure 54] is a modified Yamaha RMax helicopter that uses a unique integrated simulation and flight testing architecture. This architecture has been developed by the Georgia Tech UAV program and facilitates smooth transition from SITL to HITL simulation, followed by flight testing. Detailed description of the GTMax hardware configuration can be found in [25]. A simplified pictorial representation of the overall GTMax system architecture is shown in figure 55.



Figure 54: Georgia Tech unmanned aerial vehicle testbed- GTMax

The first component in figure 55 is the trajectory generator which provides the position, velocity and attitude commands to the flight controller as a function of time, based on a prescribed flight plan. The flight plan consists of a set of waypoints along with values for nominal velocity, acceleration and desired trajectory types (stop, aggressive turn, cut etc.) through these waypoints. The guidance system also has provision to generate trajectory for special cases such as takeoff, landing, formation flight etc. The guidance commands are passed into a baseline flight controller. The default flight controller is an adaptive neural network trajectory following controller

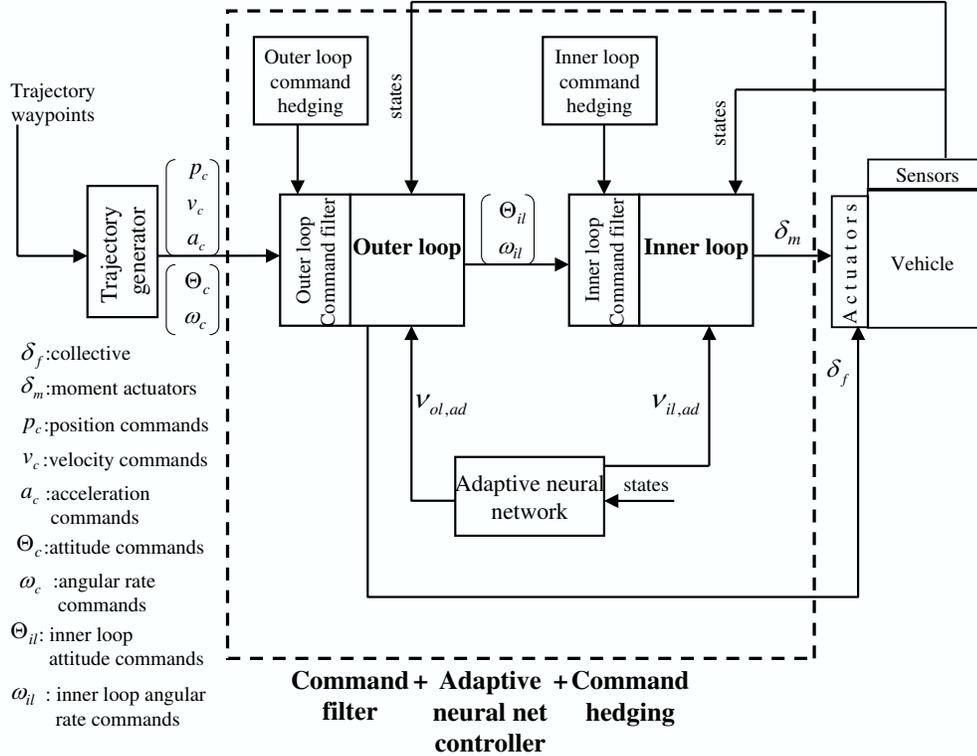


Figure 55: GTMax simulation architecture

with 18 neural network inputs, 5 hidden layer neurons, and 7 outputs for each of the 7 degrees of freedom [24,26]. The 7 degrees of freedom include the usual 6 rigid-body degrees of freedom plus a degree of freedom for rotor RPM. This adaptive neural net controller can also be configured as a conventional inverting controller. Also, included in the system architecture, but not shown in figure 55, is a baseline navigation system. The navigation system is a 17 state Extended Kalman Filter that fuses information from the five related sensors (GPS, IMU, sonar, radar, and magnetometer) to provide estimates of vehicle position, velocity, attitude (quaternion), accelerometer biases, gyro biases, and terrain height error. The navigation software also estimates whether the aircraft is on the ground or in the air by estimating the height of the airplane above the ground and assuming it is on the ground if this height is below the threshold value. The system is all-attitude capable and updates at 100 Hz. The flight controller

determines actuator commands based on the navigation system output, and the guidance system commands. The flight controller and navigation system, coupled with the trajectory generator, is capable of automatic takeoff, landing, hover, aggressive maneuvering at flight up to a maximum attainable speed of around 85 feet/sec.

GTMax helicopter model (shown in figure 55), the helicopter interface model, and sensor models have been developed as simulator tools. These simulator tools, like most of system components, are written primarily in C/C++ and have been developed to allow the test architecture to run on a high-end personal computers or laptops that uses the Windows 2000/NT or Linux operating system. The GTMax helicopter model has six rigid-body degrees of freedom plus engine, fuel, and rotor dynamics. The simulation also includes a helicopter interface model that simulates the servo interface unit functionality and RS-232 serial interface. The sensor models (IMU, GPS, sonar altimeter, magnetometer, radar altimeter) have been designed incorporating detailed information such as sensor errors, mounting location and orientation, time delays, and digital interfaces.

A previously-developed ESim simulation environment is used to produce a Graphical User Interface (GUI) as a simulator tool to this otherwise-basic C/C++ code. The GUI allows real-time 3-D display of the aircraft and the terrain, and has additional functionality to aid in data visualization or use in the Ground Control Station (GCS) during flight tests. The data visualization functionality can be used for plotting, data logging and easily modifying any data during the simulation (for example changing controller parameters). Furthermore, it also allows the simulator to run the simulation in real-time or batch mode (faster than real-time). The development of these simulator tools enables rigorous and extensive testing of any new and/or existing system modules in a way that is not possible to do in flight tests.

The test architecture also includes a generic and highly capable data communication software that has been developed to support a large number of potential flight

and simulator test configurations. This data communication software is made up of routines that support serial data reading and writing between the Commercial Off The Shelf (COTS) sensors and other custom components used on the GTMax. Also, these same routines can be used to re-route any data through Ethernet or as memory within a single executable.

4.8 *Software-in-the-loop evaluation of reactionary load factor limit protection system*

The reactionary envelope protection method is evaluated on the GTMax by using it to implement a load factor limit protection system. The objective of this load factor limit protection system is to prevent load factor response of the vehicle from exceeding 1.5g by modifying the guidance commands from the trajectory generator to the flight controller.

The first step in the design process is to choose an approximate linear model to represent load factor dynamics. In this case, the approximate linear model used to represent the load factor response is given by equation 164,

$$\dot{\hat{N}}_z = -(\hat{N}_z - 1) + 4q_c. \quad (164)$$

where q_c is the pitch rate command provided by the trajectory generator to the adaptive neural net based controller [see figure 55].

Table 8: GTMax load factor limiting- neural network design and learning rate parameters

N_{inp}	5	Output layer learning rate, Γ_m	4.0
N_{hid}	8	Hidden layer learning rate, Γ_n	0.1
N_{out}	1	Emod parameter, κ_ϵ	0.02
Basis function, σ	sigmoid $\frac{1}{1+e^{-a(x-c)}}$	Sigmoid parameters	$a = 1.0,$ $c = 0$

A single hidden layer neural network, with design and learning rate parameters as provided in table 8, is used to augment this linear model. The normalized input

vector to the neural net is selected to be the following-

$$\bar{\mu} = \begin{bmatrix} q_c & \frac{N_z}{2} & \frac{v_{x,B}}{80} & 0 & \frac{v_{z,B}}{50} \end{bmatrix}. \quad (165)$$

where N_z is the load factor response of the vehicle estimated from sensor measurements. Also, $v_{x,B}$ and $v_{z,B}$ are the aircraft velocities expressed in the body x and z frame, respectively. The noise is filtered from the acceleration sensor measurements by using a second-order low pass digital butterworth filter with a cut-off frequency of 0.2. Actual load factor response of the vehicle is then computed from the filtered acceleration measurements (\mathbf{a}_I) using the following expression-

$$N_z = \frac{\|\mathbf{a}_I\|}{g} \quad (166)$$

where $\|\cdot\|$ represents the 2-norm. In equation 166, subscript I is used to denote that the acceleration measurements are expressed in the inertial frame. Also, the variable g represents the value of acceleration due to gravity. The final form of the adaptive estimate used to represent load factor dynamics, within the reactionary envelope protection system, is given in equation 167.

$$\dot{\hat{N}}_z = -(\hat{N}_z - 1) + 4q_c + \nu_{ad}(\bar{\mu}) - 8.0(\hat{N}_z - N_z) \quad (167)$$

Notice that in equation 167 static error feedback with gain 8 is used instead of a linear dynamic compensator.

The reactionary envelope protection system is implemented as a software module on the secondary flight computer. In the test architecture, shown in figure 56, a communication link is established between the primary and secondary flight computers using routines contained within the data communications software. When this datalink is enabled the following information is sent from the primary to the secondary flight computer-

- time onboard the primary flight computer.

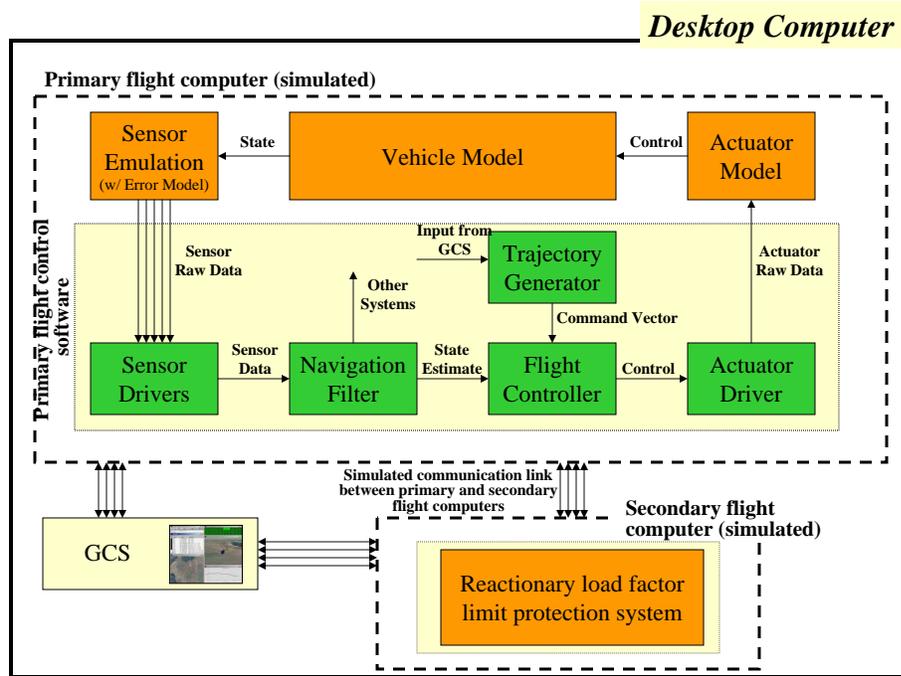


Figure 56: GTMax software-in-the-loop evaluation architecture

- vehicle position vector expressed in inertial frame.
- vehicle velocities expressed in body frame.
- vehicle acceleration measurements expressed in inertial frame.
- vehicle angular rate commands expressed in body frame.
- switch to enable/disable data recording within the envelope protection system.

The reactionary load factor limit protection system, running onboard the secondary flight computer, gets updated whenever the primary flight computer updates the datalink. A fixed time horizon of 0.4 seconds is used for predicting upper limit violations. The design parameter c_2 in equation 147 is selected to be -0.1 and ϵ_b is set at 0.01. The safe-response profile is updated according to equations 147 and 149 where Δt_{sim} is around 0.02 seconds. The exact value of Δt_{sim} is calculated based upon the onboard time sent through the datalink.

The reactionary load factor limit protection system is first evaluated on a desktop computer using Software-in-the-Loop simulation evaluations. Software-in-the-Loop (SITL) simulation configuration refers to the combined simulation of the GCS, on-board routines, and simulated sensor and vehicle dynamics, as a single executable, on any desktop computer. In this configuration, all hardware (including the helicopter itself, sensors, etc.) is simulated to the level of its digital communication to other components. This configuration is useful for rigorous software testing without requiring any actual flight hardware.

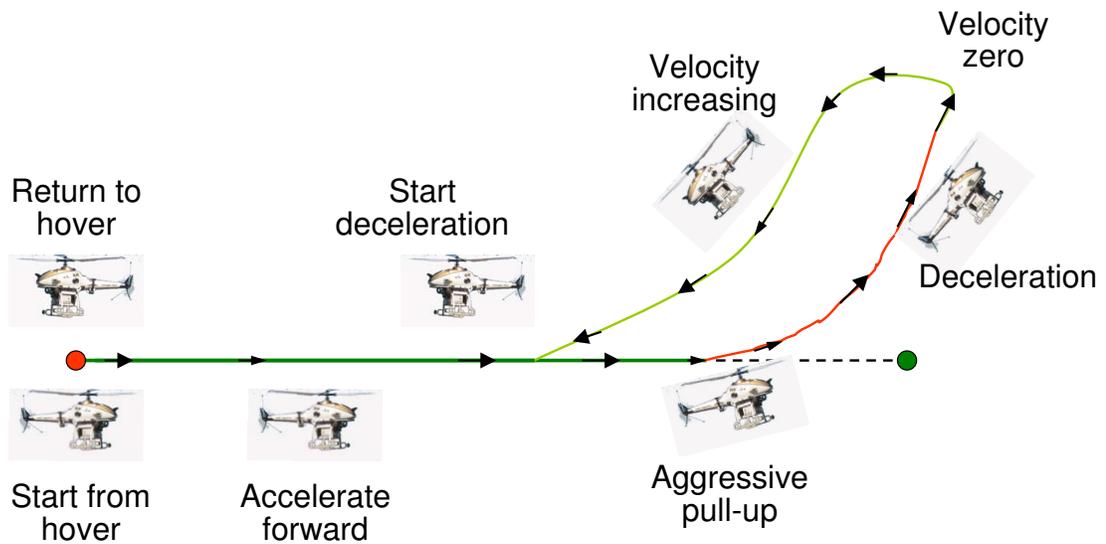


Figure 57: Graphic representation of the aggressive turn maneuver

An aggressive turn maneuver is designed for evaluating the load factor limit protection system. In the aggressive turn maneuver, as shown in figure 57, the vehicle starts from hover and accelerates to a certain forward speed. The vehicle then decelerates by pulling up, turns around and returns back to hover at the starting position. Figure 58 presents SITL results for the GTMax load factor response for an aggressive turn maneuver executed at 85 ft/sec.

Note that in figure 58 the estimate of load factor response follows the true response

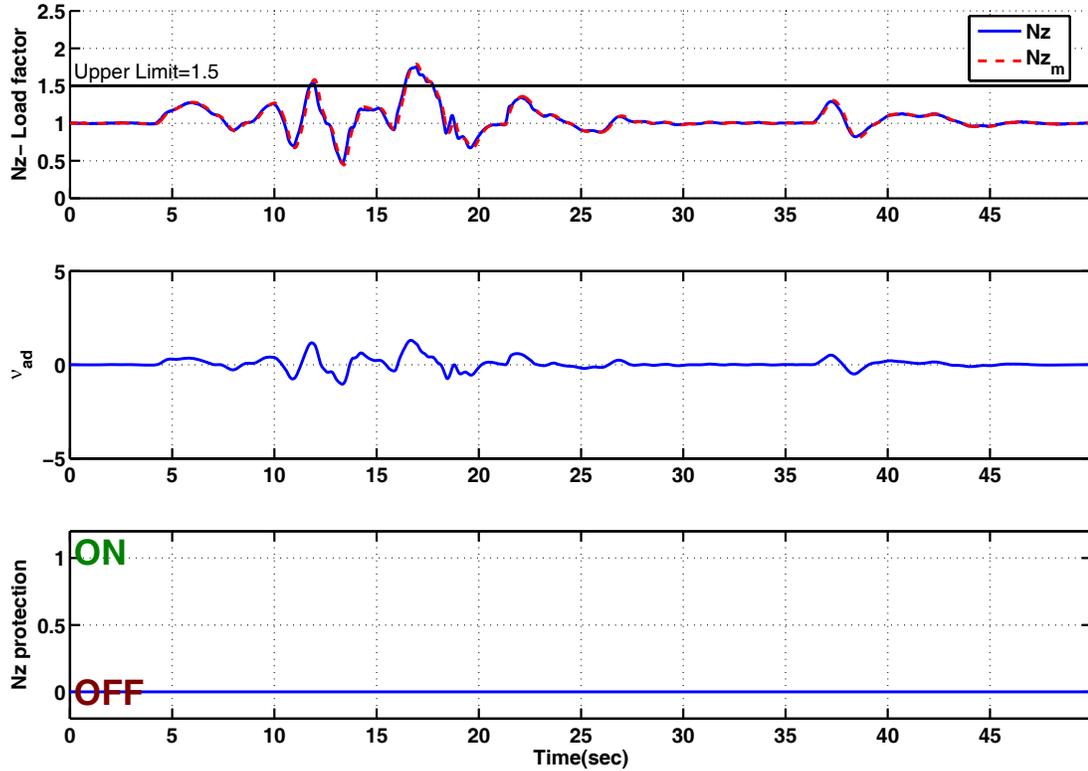
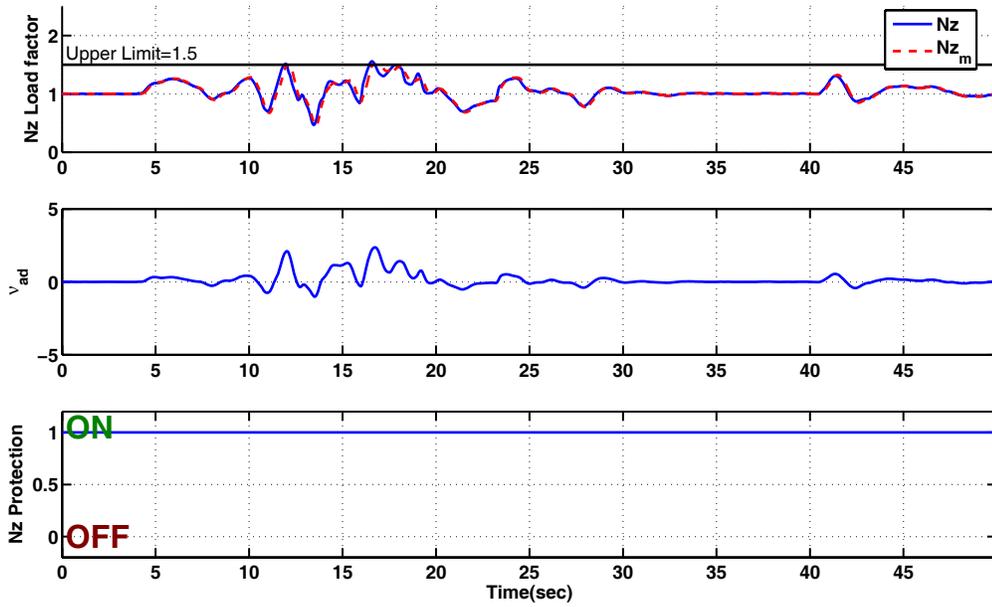
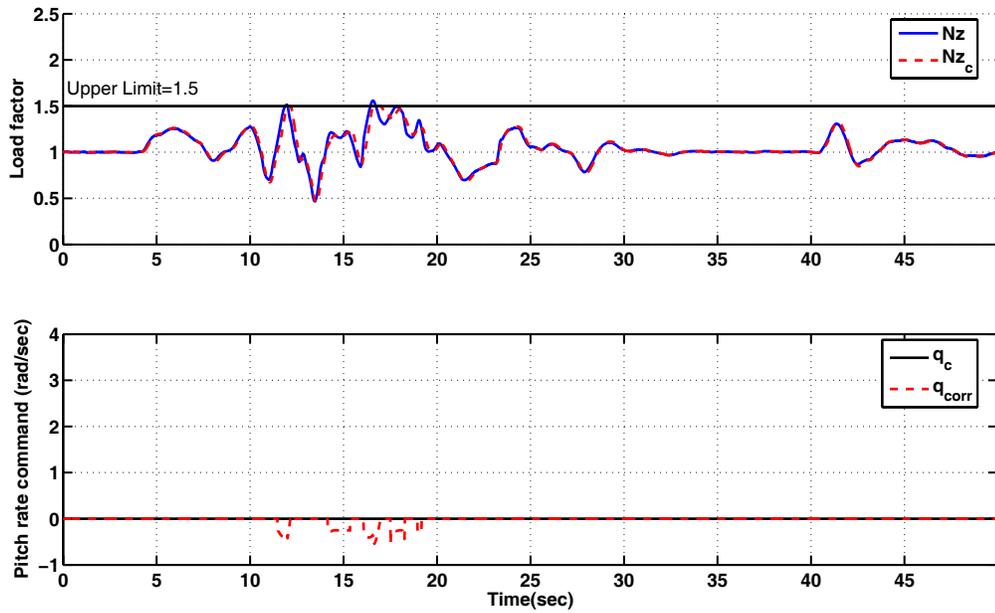


Figure 58: SITL results- Load factor response with envelope protection OFF

closely. The load factor response for the same maneuver with envelope protection activated is shown in figure 59(a). The nominal pitch rate command from the trajectory generator is zero. The load factor limit protection system calculates appropriate corrections to the pitch rate command (q_{corr}) necessary to make the limit parameter response track the safe-response profile for envelope protection. These corrections to the nominal pitch rate command, injected directly into the flight controller, has no significant effect on the vehicle response. This is mainly because the attitude as well as the angular rate commands are used by the inner loop subsystem of the flight controller (see figure 55) for calculating moment actuator commands (cyclic and pedal). The controller architecture is designed based upon time-scale separation. The outer loop subsystem calculates the force required or collective control necessary to follow the prescribed velocity and position commands. The outer loop also has the authority to modify the attitude commands to the inner loop, from the trajectory generator,



(a) Load Factor and estimated response



(b) Load factor and safe response profile

Figure 59: SITL results- Load factor response with envelope protection ON

in order to track the position and velocity commands. Hence, any corrections to inner loop angular rate commands, without corresponding changes to the outer loop commands, will be overridden. Therefore, it is necessary to translate pitch rate command corrections into appropriate corrections in acceleration, velocity and position commands.

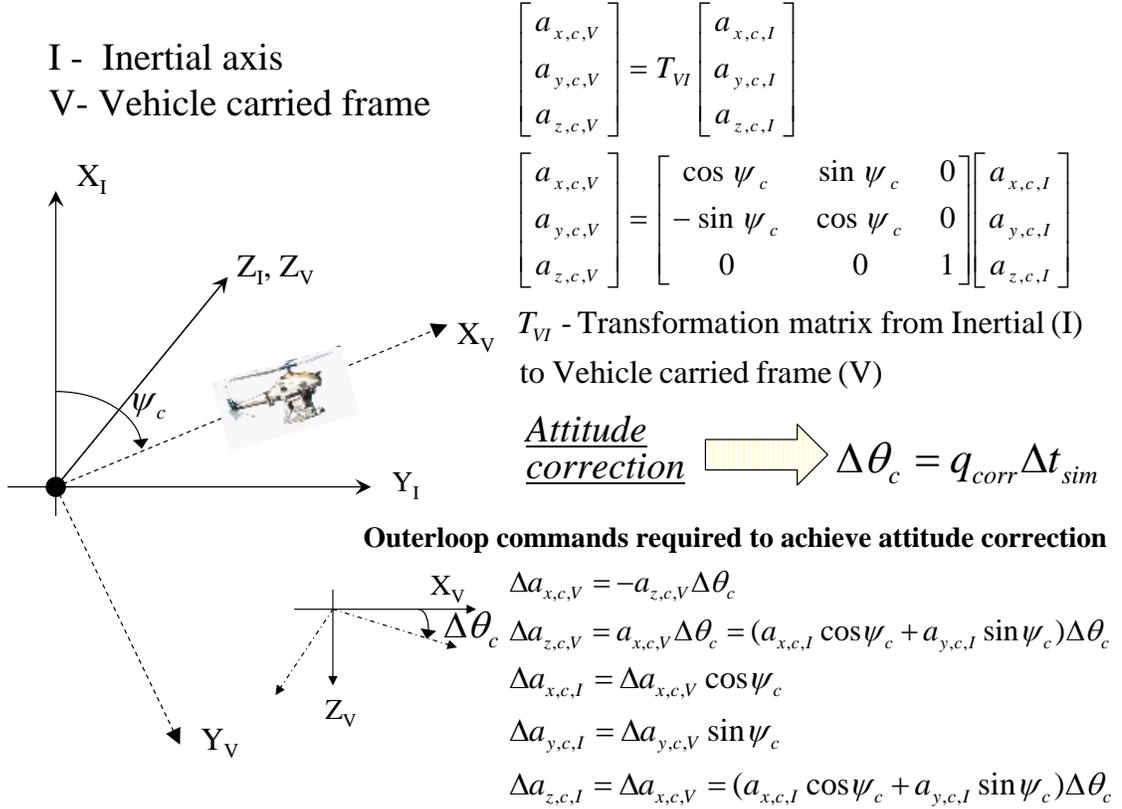


Figure 60: Pictorial representation of inertial and vehicle carried reference frames

Corrections to the acceleration commands are calculated by reasoning that tracking a pitch rate command corresponds to rotating the desired force vector. Accordingly, corrections to the nominal acceleration commands are calculated from equations 168 and 169.

$$\Delta a_{x,c,V} = a_{z,c,I} q_{corr} \Delta t_{sim} \quad (168)$$

$$\Delta a_{z,c,V} = \left(a_{x,c,I} \cos \psi + a_{y,c,I} \sin \psi \right) q_{corr} \Delta t_{sim}. \quad (169)$$

The corrections given in equations 168 and 169 are based upon the pitch rate command corrections (q_{corr}) and the simulation time-step (Δt_{sim}). Furthermore, $\Delta a_{x,c,V}$ and $\Delta a_{z,c,V}$ represent corrections to the acceleration command along the x and z directions, respectively of the vehicle carried frame [see figure 60]. Also, $a_{x,c,I}$, $a_{y,c,I}$ and $a_{z,c,I}$ are the acceleration commands along the inertial x, y and z directions, respectively. The results presented in figure 59(b) show that with these corrections, the envelope protection system successfully maintains the load factor response within the set upper limit of 1.5g. Also, in the test results with envelope protection active [see figure 59(a)] the response from the adaptive estimate of load factor dynamics continues to track the true response well.

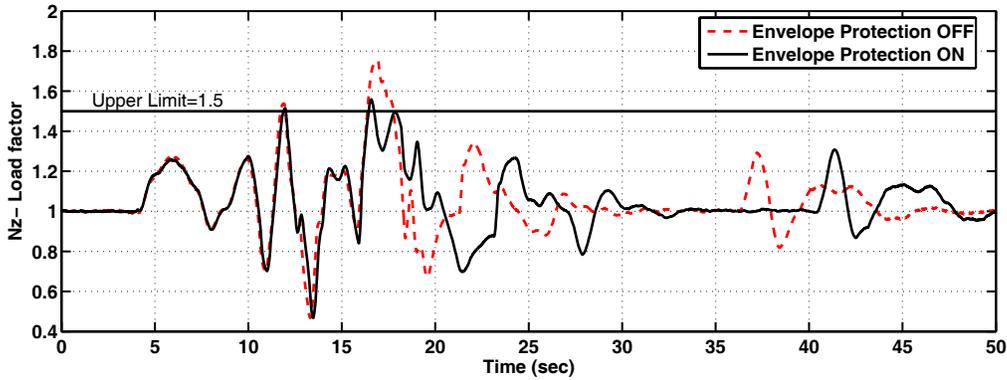


Figure 61: SITL results- Load factor response comparison with and without envelope protection

Figure 61 compares the load factor response with and without envelope protection and for this particular maneuver the proposed envelope protection approach results in maximum utilization of available envelope.

Also, the vehicle trajectory with and without imposing load factor limit protection is shown in figure 63. Notice that the pull-up during the aggressive turn maneuver, with load factor limit protection active, is less steeper compared to that with limit protection switched off. This observation is easily corroborated by comparing the velocity responses with and without envelope protection. In figure 62, when load

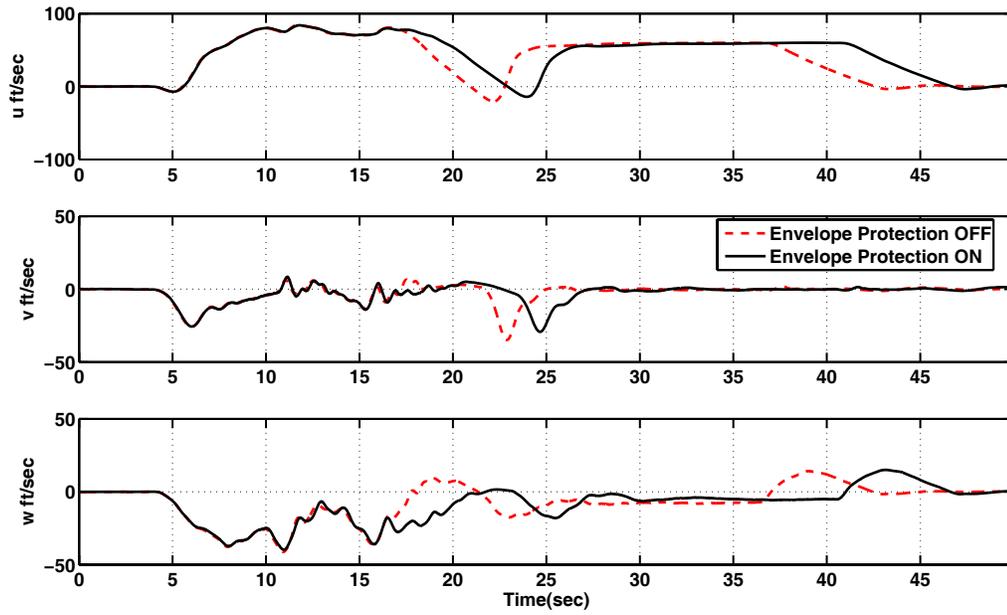


Figure 62: SITL results- Velocity response profiles with and without envelope protection

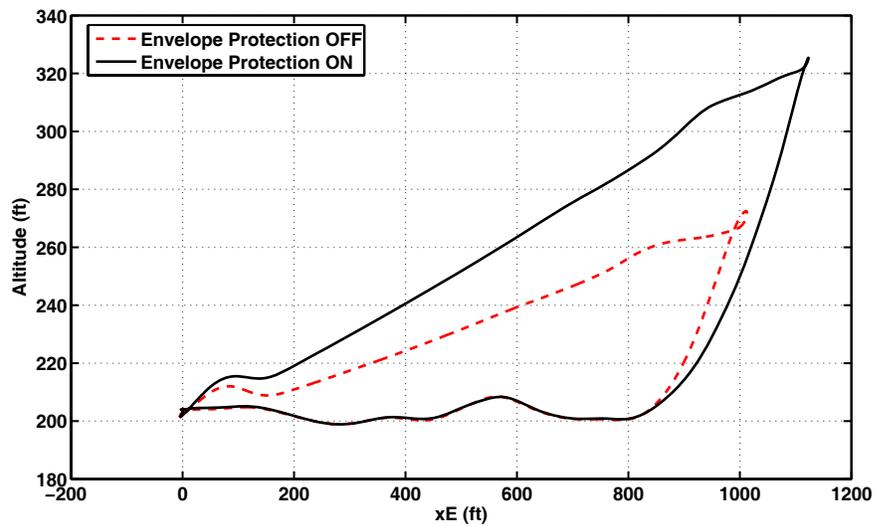


Figure 63: SITL results: aggressive turn trajectory with and without load factor limit protection

factor limit protection is active the vehicle decelerates more slowly than without envelope protection. This prevents large pitch rate responses from developing and causing load factor upper limit violations. Furthermore, the vehicle is prevented from developing large vertical acceleration responses during the pull-up by limiting the vertical acceleration commands. This difference is easily noticed by comparing the vertical velocity responses with and without envelope protection in figure 62. As a penalty to limit protection, the vehicle takes longer to complete the aggressive turn maneuver. In a real scenario, where the limit protection is critical for the safety of the vehicle, this penalty on aggressiveness should be weighed against maintaining the overall safety of the vehicle.

4.9 Flight evaluation of reactionary load factor limit protection system

The simulator tools developed make software execution possible on the actual flight hardware (HITL). Hardware-in-the-Loop (HITL) simulation test is done as an intermediary step prior to the flight evaluation. HITL test configuration is shown in figure 64. In this configuration, the onboard software is compiled and executed on the actual flight computer as shown in figure 64. The sensor and vehicle interface models are used to interface the flight computer with the simulation-host computer. The simulation-host computer is used to execute the combined vehicle, actuator and sensor model simulation. The configuration of the primary and secondary flight computer in HITL test is identical to that used in flight test. The HITL simulation configuration is used to test all guidance, navigation, and control algorithms software and as much of the hardware as practical, in real-time. The hardware under test are the servos and the flight computers with software running on them.

The flight test configuration is very similar to the HITL test configuration. In the flight test configuration the simulation-host computer, in figure 64, is replaced

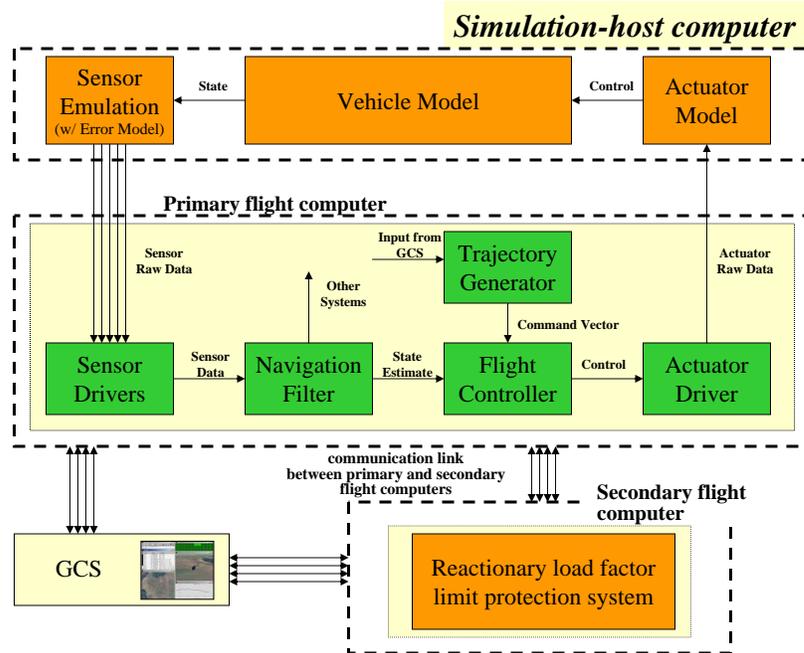


Figure 64: Hardware-in-the-loop test architecture

by the actual vehicle. Also, during flight tests the GCS software is run on one or more laptop computers and is used by human operators to interact with the onboard systems. The GCS is also equipped to read differential correction data from a GPS reference system and send it to the vehicle. The datalink software provides connection between the GCS and onboard software. It is optimized to minimize bandwidth use (for example, by sending “float” instead of “double” wherever possible) and also supports communication redundancy. Currently, a message sent at 10 Hz is used to update status and current state information on the GCS. A 1 Hz message provides other significant onboard data.

The flight test results for aggressive turn at 85 ft/sec, without activating reactionary load factor limit protection system, is shown in figure 65. As observed previously in SITL tests, the load factor response exceeds the upper limit of 1.5g during aggressive pull-up phase of the maneuver.

The same maneuver is now repeated with envelope protection active and the results are shown in figures 66(a) and 66(b). Observe that in both test cases, with and

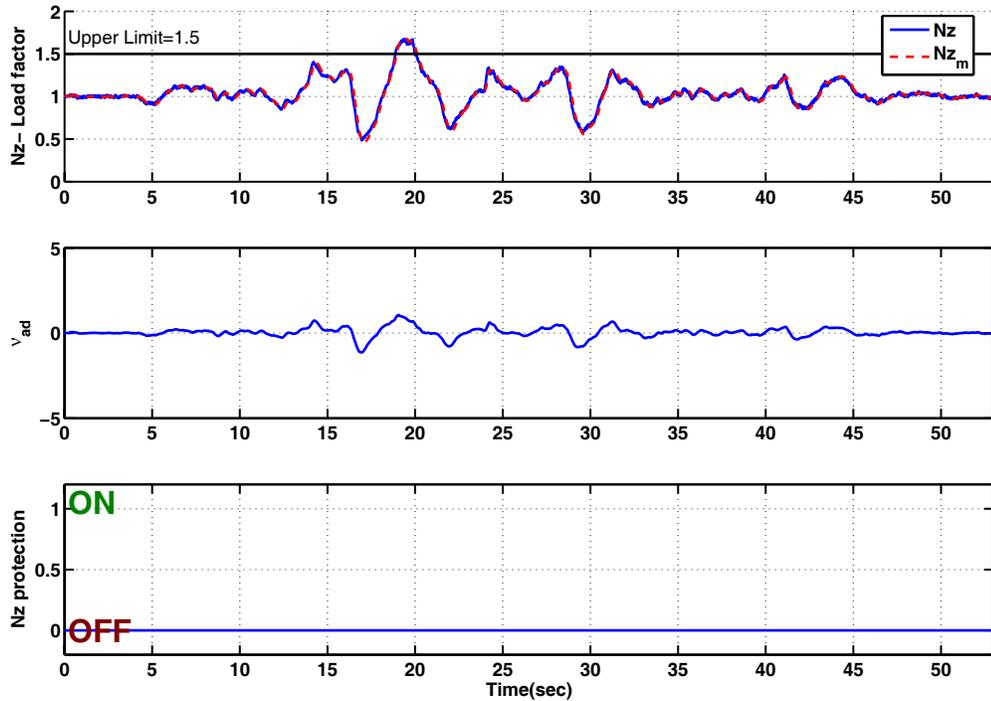
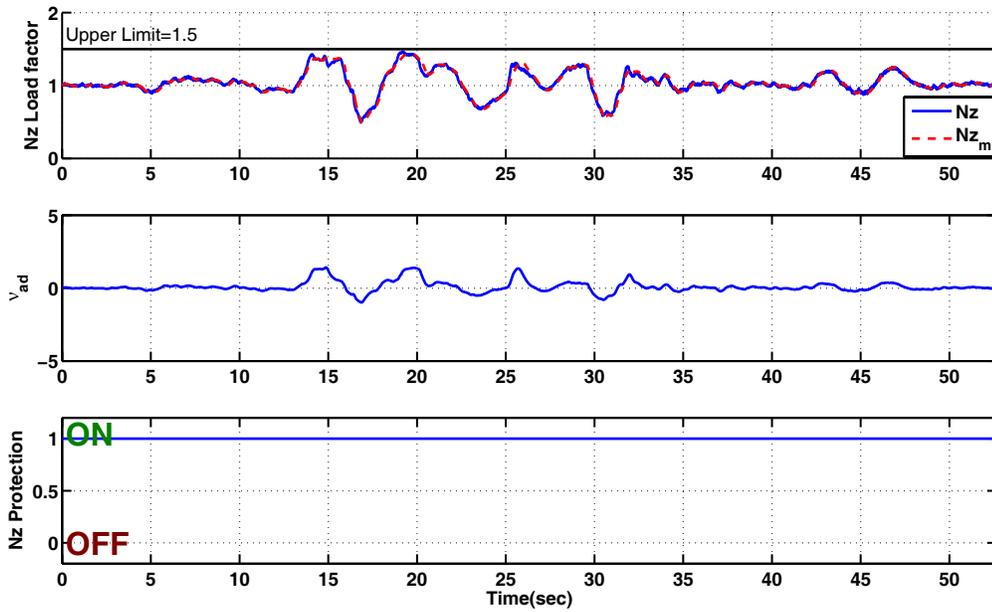


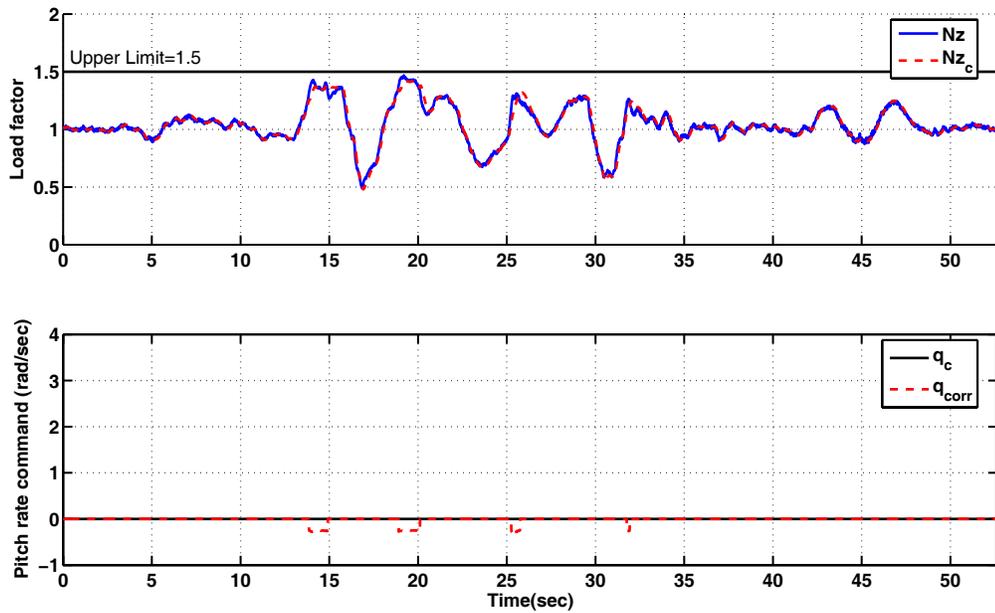
Figure 65: Flight test results- GTMax load factor response with envelope protection OFF

without envelope protection, the estimate of the limit parameter dynamics follows the true load factor response closely. Also, notice in figure 66(b) that when load factor limit protection is switched on, the true load factor response is made to track the safe-response profile. This safe-response profile will differ from the estimate of the limit parameter dynamics only when envelope violation is predicted or detected. The corrections estimated by the envelope protection system to the pitch rate command are also shown in figure 66(b). As mentioned before, these corrections are translated into acceleration command corrections within the trajectory generator according to equations 168 and 169.

The effect of reactionary envelope protection on the overall performance can be studied using figures 67, 68 and 69. In figure 67 a comparison of load factor response with and without envelope protection is presented. Notice that using the envelope protection does not cause the load factor response to be overly conservative with respect to



(a) Load Factor and estimated response



(b) Load factor and safe response profile

Figure 66: Flight test results- GTMax load factor response with envelope protection ON

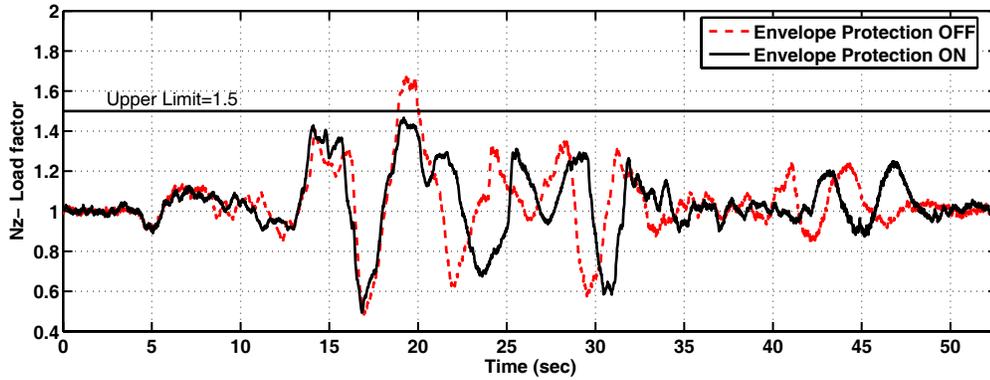


Figure 67: Flight test results- GTMax load factor response comparison with and without envelope protection

the prescribed envelope.

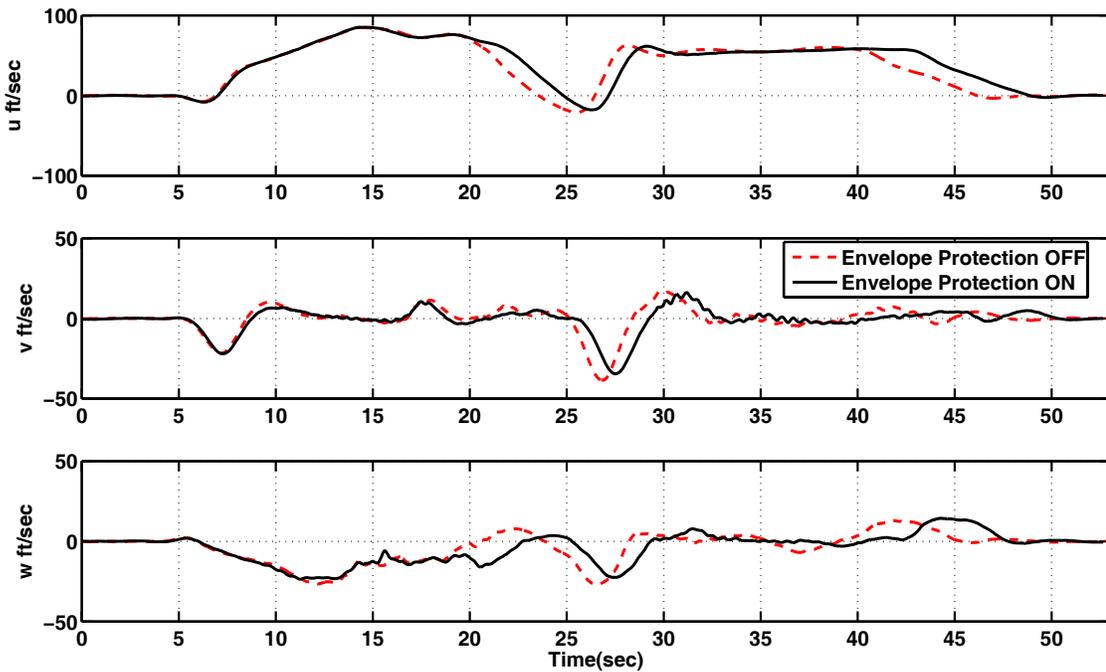


Figure 68: Flight test results- GTMax velocity response profiles with and without envelope protection

Figure 68 shows the differences in the velocity responses. The differences observed are similar to those seen during SITL evaluations. The reactionary envelope protection modifies the nominal commands by reducing the rate of deceleration and the rate of climb during the pull-up. This inturn has the effect of increasing the distance

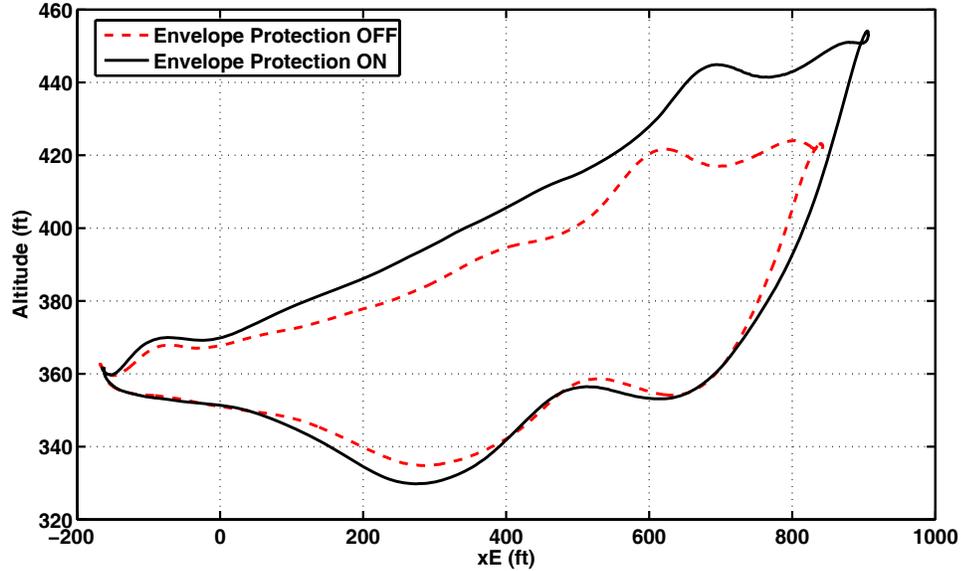


Figure 69: Flight test results- aggressive turn trajectory comparison with and without load factor limit protection

and height the vehicle gains during the pull-up as shown in figure 69. Overall the performance of the reactionary load factor limit protection system in flight test is similar to that observed during SITL evaluations.

4.10 R22 longitudinal flap angle limiting

The aerospace engineering UAV research lab at Georgia Tech participated in the DARPA HURT project that involves the use of Maverick UAV. The Maverick UAV is sold by Frontier systems now currently owned by Boeing[®]. It is a retrofitted commercially available helicopter to the U.S. Special Operations Command and has been used as a test bed for A-160 (Hummingbird) technologies.

The UAV research lab has incorporated a Flightlab[®] generated math model of the Maverick UAV, as the vehicle model, within the existing GTMax control architecture [see figure 55]. For this the GTMax math model in figure 55 is replaced with an equivalent high fidelity math model of the Maverick UAV. The vehicle math model within the control architecture has been tested and verified using SITL simulation

Table 9: Maverick UAV flap angle limit protection- neural network design and learning rate parameters

N_{inp}	12	Output layer learning rate, Γ_m	2.0
N_{hid}	13	Hidden layer learning rate, Γ_n	0.2
N_{out}	1	Emod parameter, κ_ϵ	0.01
Basis function, σ	sigmoid $\frac{1}{1+e^{-a(x-c)}}$	Sigmoid parameters	$a = 1.0,$ $c = 0$

evaluations.

Flapping angle is an important limit parameter for the Maverick UAV that needs to be monitored and maintained within reasonable values. A longitudinal flap angle limit protection system is implemented using the reactionary envelope protection method. The flap angle outputs from the math model are used since sensor measurements are unavailable. MATLAB[®] system identification toolbox is used for off-line analysis of data recorded during the aggressive turn maneuver [see figure 57]. The analysis is used to conclude that the order of the flapping dynamics is four and that relative degree with respect to pitch rate command is one. The system identification is also used for choosing the approximate linear model for flapping dynamics given in equation 170.

$$\dot{\hat{\beta}}_{lon} = -\hat{\beta}_{lon} + 10q_c \quad (170)$$

The linear approximate model in equation 170 is augmented with a SHL-NN the parameters of which are given in table 9. The input to the network is given in equation 171. Also, a simple static error feedback with gain of 10 is used in place of linear dynamic compensator.

$$\bar{\mu} = \left[\beta_{lon}(t) \quad \dots \quad \beta_{lon}(t-3d) \quad \hat{\beta}_{lon}(t) \quad \dots \quad \hat{\beta}_{lon}(t-3d) \quad \frac{u_B}{80} \quad q \quad \frac{w_B}{20} \right]^T \quad (171)$$

The final form of the adaptive estimate of the flapping dynamics is given in equation 172-

$$\dot{\hat{\beta}}_{lon} = -\hat{\beta}_{lon} + 10q_c + \nu_{ad}(\bar{\mu}) - 10(\hat{\beta}_{lon} - \beta_{lon}) \quad (172)$$

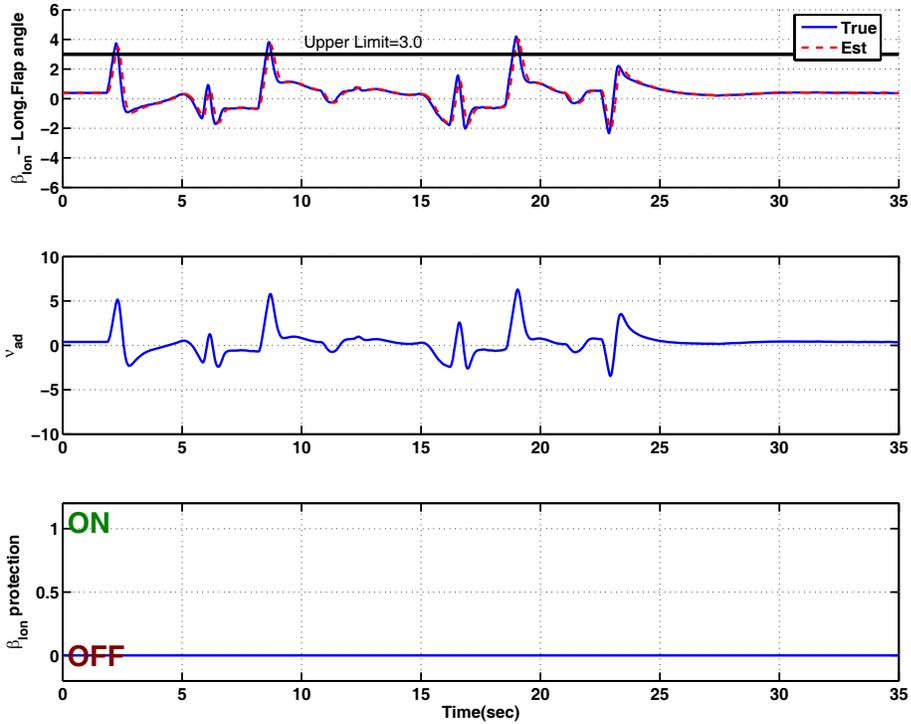


Figure 70: R22- Response with envelope protection OFF

For evaluating the effectiveness of the reactionary flap angle limit protection system an artificial upper limit of 3.0 is imposed on the flap angle response during turn. Figure 70 shows the response during an aggressive turn maneuver of the earlier specifications without envelope protection. Notice that the response of the adaptive estimate tracks the true response closely. In figure 70 without envelope protection two significant upper limit boundary violations occur.

The response with reactionary envelope protection switched on is shown in figures 71 and 72. In figure 71 the response of the adaptive limit parameter estimate is compared to the true response and found to be close to each other. When the flap angle reactionary envelope protection system is activated it prevents the response from violating the prescribed upper limit. The nominal pitch rate command from the trajectory generator is zero. The corrections to this nominal pitch rate command are shown in figure 72. These corrections, translated into acceleration command correction as given in equations 168 and 169, force the flap angle response to track

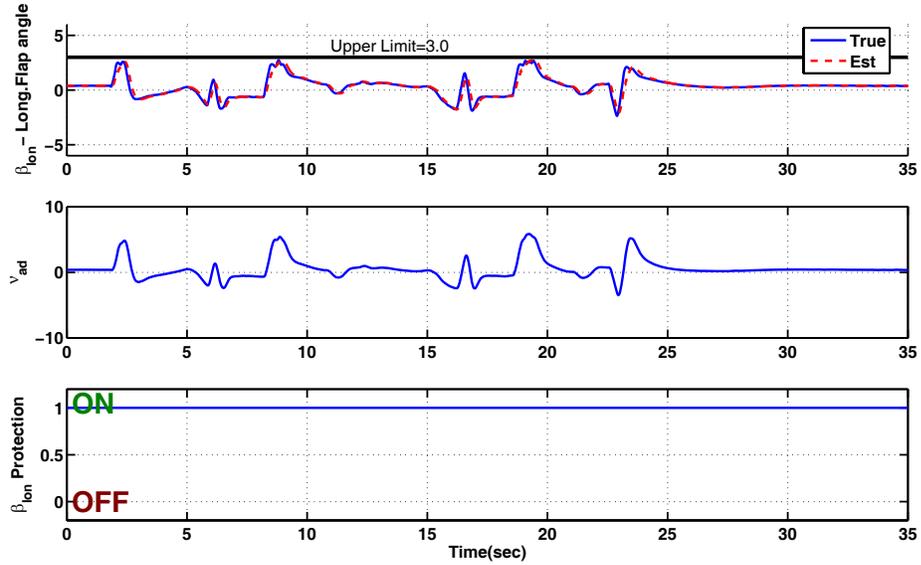


Figure 71: R22- Response with envelope protection ON

the safe-response profile near the upper limit boundary as shown in figure 72.

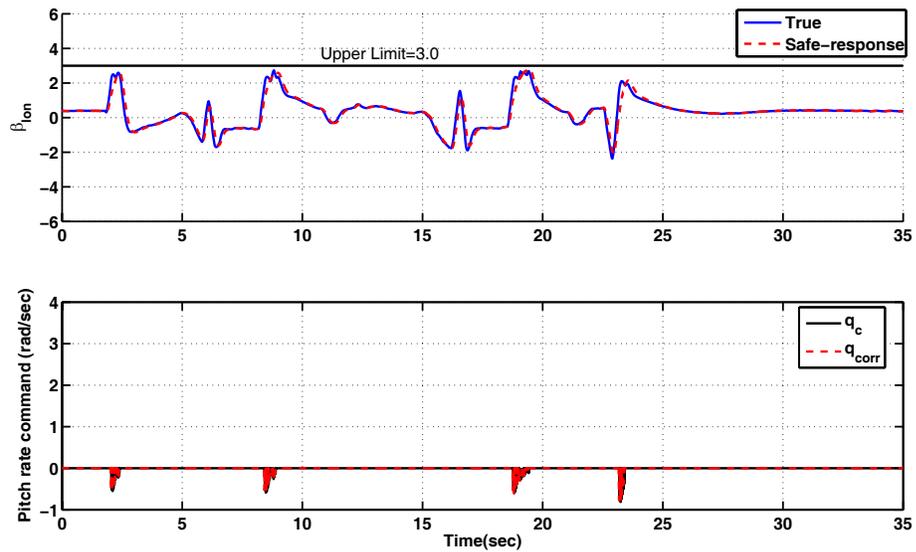


Figure 72: R22- Safe response tracking for envelope protection

A comparison of the longitudinal flap angle with and without reactionary envelope protection is shown in figure 73. The simulation evaluations using the turn maneuver clearly demonstrates that reactionary envelope protection method is a promising approach for longitudinal flap angle limit protection in the Maverick UAV.

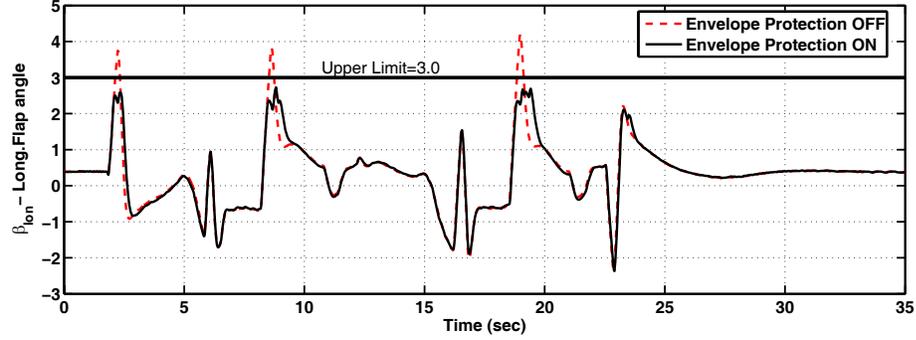


Figure 73: R22- Comparison of flap angle response with and without envelope protection

4.11 Calculation of control margin using reactionary envelope protection method

The idea of reactionary envelope protection can be extended to construct an estimate of the available control margin which, as mentioned earlier, is useful for manned envelope protection systems.

Proposition 4.11.1. *The control margin corresponding to a limit boundary (upper or lower) of a limit parameter response can be estimated from the following equation:*

$$u_{margin} = \left[y_s^{(r)} - \left(\sum_{i=0}^{r-1} a_i y_s^{(i)} + \frac{\partial \hat{h}_r}{\partial u_p} u_p + \nu_{ad}(\bar{\mu}) - \nu_{dc} \right) - d_l(\hat{y}_p - y_s) \right] \left(\frac{\partial \hat{h}_r}{\partial u_p} \right)^{-1} \quad (173)$$

by choosing appropriate values for safe-response profile variables $(y_s, y_s^{(1)}, \dots, y_s^{(r)})$.

Equation 173 is of the same form as that of the control correction proposed for safe-response profile tracking in equation 153. However, the safe-response profile variables used in equation 173 do not vary based on the aggressiveness of the limit parameter response. Instead, they are set to pre-determined values based on the acceptable limit parameter response aggressiveness near the limit boundaries. In other words, the control margin is based upon the notion that the limit parameter response, when it reaches near the limit boundary, should slow down sufficiently enough not to violate the boundary and the pre-determined characteristic represents the limits on the aggressiveness of the limit parameter response as it approaches the

limit boundary. In case of relative degree one ($r=1$) limit parameter response for example, the safe-response profile is set as:

$$\begin{bmatrix} y_s & \dot{y}_s \end{bmatrix}^T = \begin{bmatrix} y_{lim} & 0 \end{bmatrix}^T \quad (174)$$

Notice that the above settings for $r = 1$ is similar to that used in the adaptive dynamic trim method [68]. For relative degree two limit parameter dynamics, ($r=2$) control margin is calculated by setting the safe-response profile variables as-

$$\begin{bmatrix} y_s & \dot{y}_s & \ddot{y}_s \end{bmatrix}^T = \begin{bmatrix} y_{lim} & 0 & k \end{bmatrix}^T \quad (175)$$

where

$$k = \begin{cases} \leq 0 & \text{for upper limit} \\ \geq 0 & \text{for lower limit} \end{cases} \quad (176)$$

Henceforth, without loss of generalization, assume that the control sensitivity ($\frac{\partial \hat{h}_r}{\partial u_p}$) is positive.

Proposition 4.11.2. *The control margin, for the upper limit, goes from positive to negative as the limit parameter response approaches the limit boundary and the available limit margin, when control margin is zero, will be positive. Furthermore, the amount of limit margin available at the point of zero control margin increases as the aggressiveness of limit parameter response increases.*

The control margin with respect to the upper limit is referred to as the upper control margin and according to proposition 4.11.2 this upper control margin goes to zero prior to the actual violation of the upper limit boundary. Further analysis will show that this prediction horizon is implicitly present in the proposed control margin calculation.

Re-arranging terms in equation 10 we get,

$$bu_p + \nu_{ad}(\bar{\mu}) - \nu_{dc} = \hat{y}_p^{(r)} - \sum_{i=0}^{r-1} a_i \hat{y}_p^{(i)} \quad (177)$$

Substituting equation 177 into equation 173 we get the following,

$$u_{marg} = \left[y_s^{(r)} - \left(\sum_{i=0}^{r-1} a_i y_s^{(i)} + \hat{y}_p^{(r)} - \sum_{i=0}^{r-1} a_i \hat{y}_p^{(i)} \right) - d_l (\hat{y}_p - y_{lim}) \right] \left(\frac{\partial \hat{h}_r}{\partial u_p} \right)^{-1} \quad (178)$$

$$= \left[y_s^{(r)} - \hat{y}_p^{(r)} - \sum_{i=0}^{r-1} a_i (y_s^{(i)} - \hat{y}_p^{(i)}) - d_l (\hat{y}_p - y_{lim}) \right] \left(\frac{\partial \hat{h}_r}{\partial u_p} \right)^{-1} \quad (179)$$

$$= \left[y_s^{(r)} - \hat{y}_p^{(r)} - \sum_{i=1}^{r-1} a_i (y_s^{(i)} - \hat{y}_p^{(i)}) - (a_0 - d_l) (y_{lim} - \hat{y}_p) \right] \left(\frac{\partial \hat{h}_r}{\partial u_p} \right)^{-1} \quad (180)$$

In equation 180, $(a_0 - d_l) < 0$ for stability. Therefore, when the limit parameter response is far away from the limit boundary then the term $(a_0 - d_l)(y_{lim} - \hat{y}_p)$ is much greater than the other terms in equation 180. Therefore, the control margin will be positive (negative for negative control sensitivity) far away from the limit boundary and negative (positive for negative control sensitivity) near it. The value of limit margin $(y_{lim} - \hat{y}_p)$ for which control margin is zero is obtained from equation 180 by setting $u_{marg} = 0$.

$$(y_{lim} - \hat{y}_p) = \frac{y_s^{(r)} - \hat{y}_p^{(r)} - \sum_{i=1}^{r-1} a_i (y_s^{(i)} - \hat{y}_p^{(i)})}{(a_0 - d_l)} \quad (181)$$

Analyzing equation 181 for the special case of $r = 1$ we can see that as the limit parameter approaches the upper limit boundary $0 - \dot{\hat{y}}_p$ is negative and $(a_0 - d_l) < 0$. Therefore, control margin becomes zero before approaching the limit boundary which signifies a lead in prediction of envelope violation. Furthermore, equation 181 can be used to deduce that the limit margin corresponding to the zero control margin will vary depending on the aggressiveness of the limit parameter response $(a_0 - d_l)$. The more aggressive the limit parameter response is, the larger is the limit margin (or control limiting starts further away from the limit boundary). This is an extremely

desirable behavior for an envelope protection method. The pilot of a manned vehicle would expect to be informed sooner about approaching the limit boundary for a faster limit parameter response than the one which is more gentle and non-aggressive. Note that the proposed approach for computing control margin is based on extending the idea underlying the reactionary envelope protection method. It does not utilize any functional forms for estimating maximum peak response corresponding to a given control input. The proposed method is based on how aggressively the limit parameter response is approaching the limit boundary and whether this aggressiveness is within the specified safety settings pre-determined by the envelope protection system designer.

4.11.1 Reactionary envelope protection method for hub moment limit protection

The NTG based envelope protection method and nonlinear function response method have been successfully tested for preventing hub moment limit exceedances by using force-feedback based tactile cueing to inform the pilot when approaching the estimated longitudinal cyclic control limits. Similar to the NTG approach, control margin calculations in the reactionary envelope protection method are also based upon the adaptive estimate of limit parameter dynamics.

Figure 74 presents the hub moment response during a swoop maneuver with the vehicle in nominal weight configuration. The envelope protection is not active, i.e the softstop locations are not set based on critical control position calculations to cue the pilot. In figure 74 the control limits calculated using the nonlinear function response method are presented in the second subplot. Also, the control limits calculated using the reactionary control margin are presented in the fourth subplot. The safe-response profile settings used in the reactionary control margin calculations are $y_s = y_{lim}$ and $\dot{y}_s = 0.0$. It is observed that the control limits calculated using reactionary envelope

protection method are violated prior to the occurrence of corresponding limit boundary violation. Furthermore, the reactionary control margins go to zero (control limit violation) slightly ahead of the control margins calculated using nonlinear function response method. This signifies that an additional lead time would be available to cue the pilot if reactionary envelope protection method were to be used.

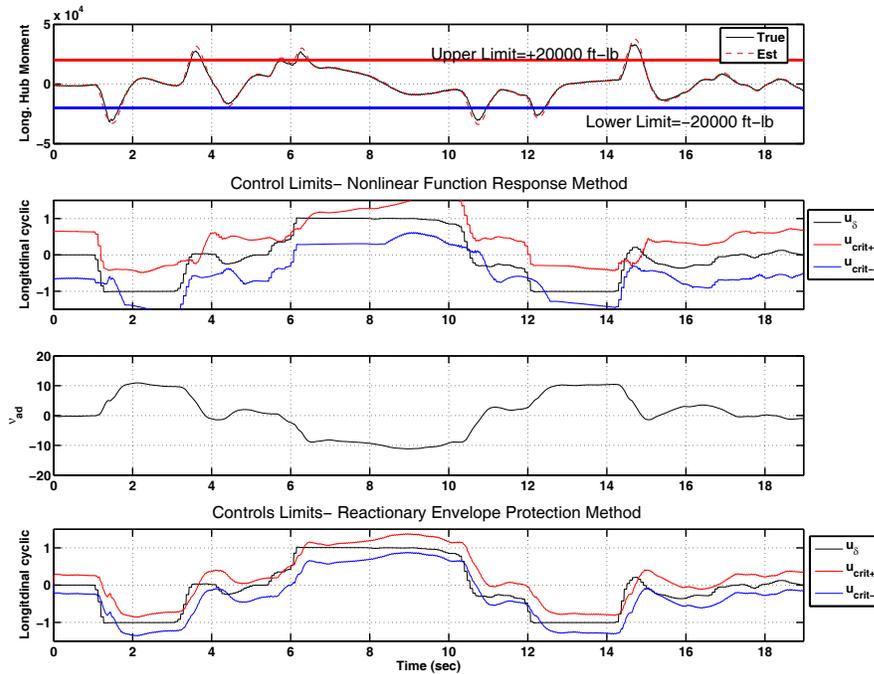


Figure 74: Control limits for hub moment limit protection- Reactionary vs nonlinear function response method

The control limit calculations of the reactionary envelope protection method for the same maneuver are compared to those of the NTG based approach in figure 75. The results presented in figure 75 show that the NTG based approach is comparable to the reactionary envelope protection method in terms of the lead time or time between the control limit violation and the actual limit violation. However, the control limit calculations in the NTG approach are based not only on the aggressive control profile but also the critical time of the limit parameter response. Therefore, when the critical time of the limit parameter response is close to zero the control margins are also nearly zero. The exact control margins are calculated using the smoothing function.

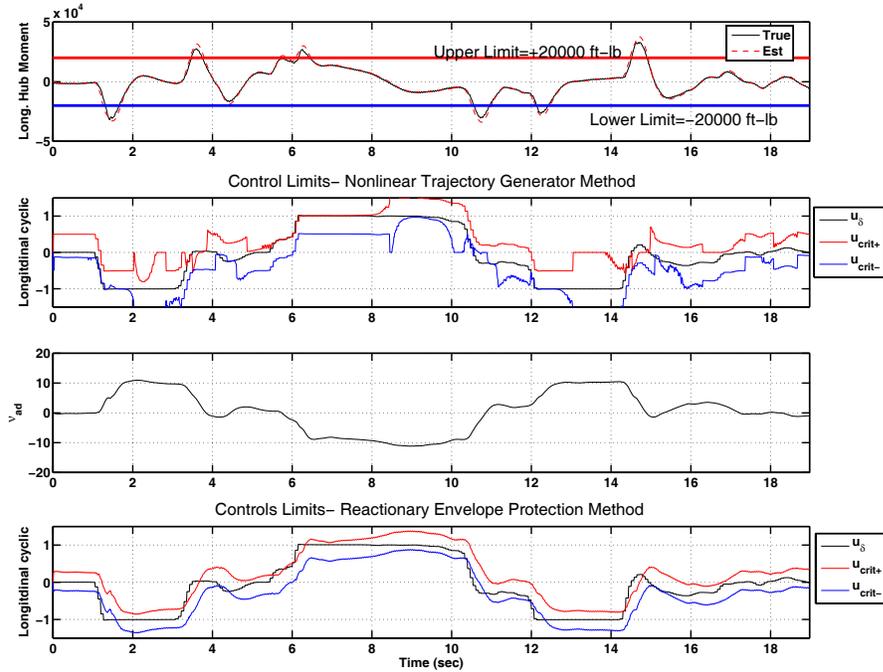


Figure 75: Control limits for hub moment limit protection- Reactionary vs NTG method

Hence, in the NTG case whenever the critical time of the limit parameter response takes very low values the control limits follow the control input closely signifying zero control margin. In an actual test case with envelope protection active the softstop is placed slightly ahead of the critical control position to account for very low values of control margin. The results presented in figures 74 and 75 clearly show that the control margins based upon reactionary envelope protection method are comparable to that obtained using the nonlinear function response method or the NTG approach. Therefore, the proposed extension is a viable envelope protection method for piloted vehicles.

4.11.2 Example simulation- R-22 engine manifold pressure limit protection

As a part of the DARPA Software Enabled Control (SEC) [59] technology transition project Boeing is developing a commercially available R22 Robinson helicopter as an UAV test platform. The idea is to implement, evaluate and demonstrate emerging

technologies on this platform before they are transitioned into other current and future UAV platforms. Envelope protection is one among many other advanced technologies, such as vision-based autonomous landing, aggressive maneuvering etc., that will be implemented and tested using this platform.

A reactionary envelope protection module for engine manifold pressure limit protection is implemented within the Boeing Open Control Platform (OCP). The OCP is a middleware platform that facilitates easy integration and rapid prototyping of advanced control technologies that are being developed under the SEC program [45]. Engine intake manifold pressure is available as a sensor measurement on the vehicle and is a limit parameter monitored by R-22 pilot. A recent safety alert issued by the Robinson helicopter company on December of 2004 warned against premature fatigue failure of R-22 blades. This occurs when engine manifold pressure limits are exceeded producing repeated and excessive over-stressing of the blades [20]. Also, exceeding manifold pressure limits in a nominal R-22 helicopter will be followed by loss of main rotor RPM. If corrective action is not taken quickly and RPM is allowed to drop then rotor stall will occur. During rotor stall, the blades come to a full stop and experience excessive blade flapping can cause the tail boom section to be cut-off. In the current R22 helicopter, being developed as a UAV technology demonstrator, this situation is prevented from occurring by the addition of a turbo to the engine. However, this does not prevent the over-torquing of the transmission and engine resulting in excessive fatigue damages. Therefore, the large excursions of engine manifold pressure, from manufacturer specified limits, must be avoided.

The proposed test architecture will include envelope protection as a mid-level component within the OCP based system architecture as shown in figure 76. The envelope protection system will be designed to modify or input vertical velocity commands to the OCP controller such that the engine manifold pressure is maintained below the prescribed upper limit.

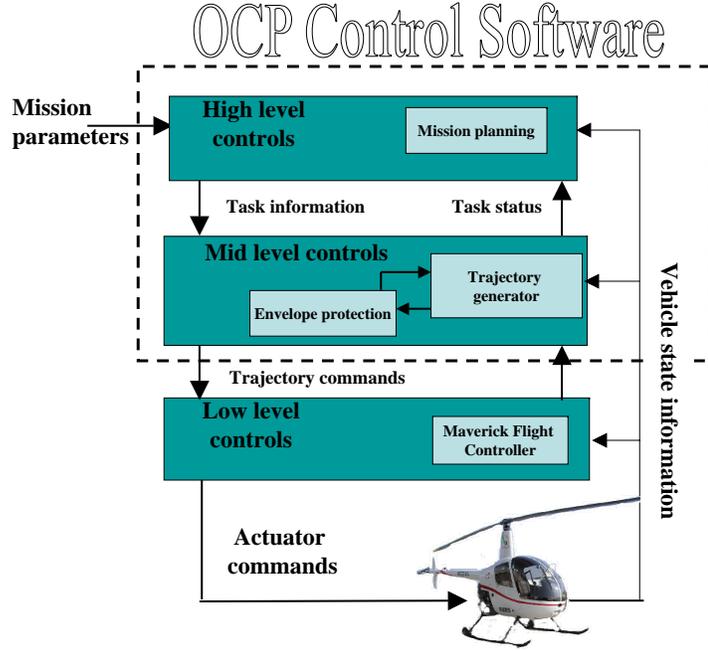


Figure 76: Flight test architecture

A linear model for engine manifold pressure has been developed using MATLAB[®] system identification toolbox. The engine manifold pressure dynamics is identified as a first order system given in equation 182. The input to this system are the collective (δ_{coll}), percentage of main rotor rpm (Ω_{MR}) and the vertical velocity (v_z). As shown in figure 77, the manifold pressure variations predicted by the model match well with the flight test data.

$$\dot{P}_m = -0.3733P_m + 1.579\delta_{coll} - 0.02868\Omega_{MR} + 0.07709v_z. \quad (182)$$

The manifold pressure dynamics represented by equation 182 is included as a part of the R-22 model in the simulation architecture of figure 76. The R-22 vehicle model is obtained from the model parameters generated using Flightlab[®]. The model response has recently been validated against flight test data using inverse simulation [44].

Figure 78 shows a simplified architecture used for Software-in-the-loop simulation evaluations of the engine manifold pressure reactionary envelope protection system.

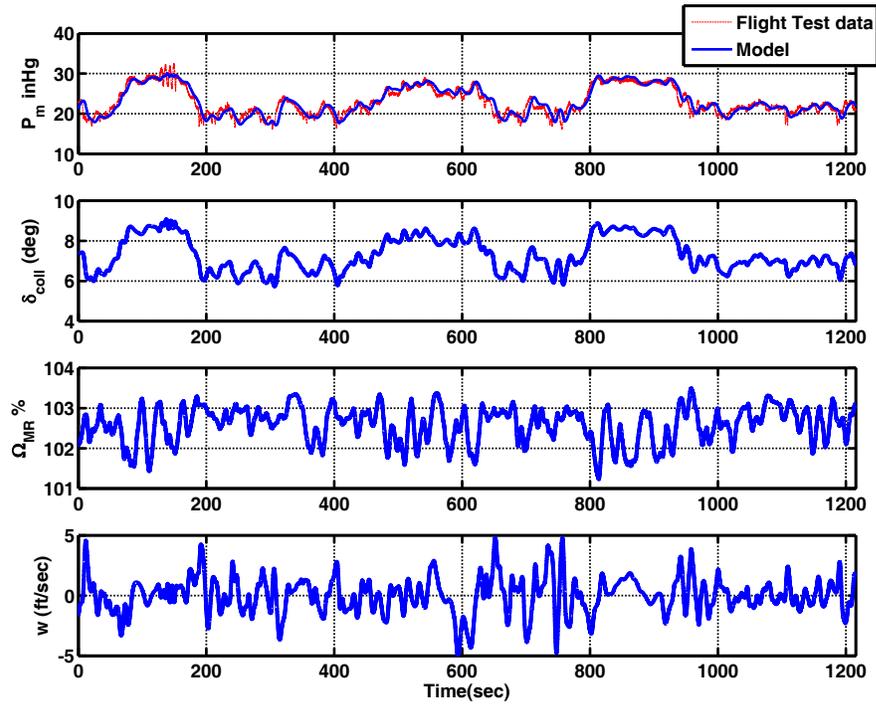


Figure 77: Engine intake manifold pressure model validation

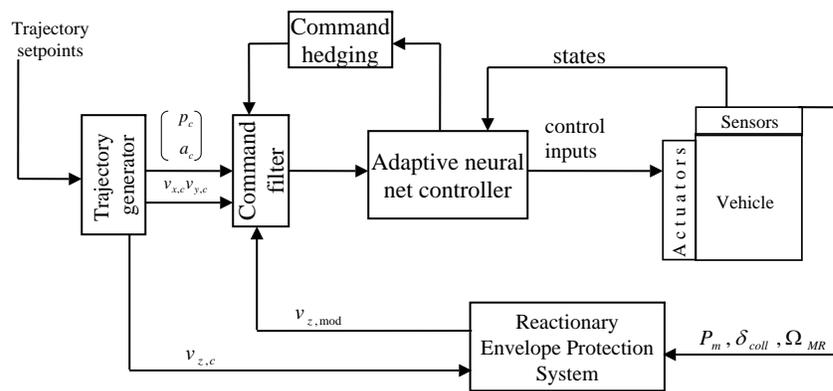


Figure 78: SITL validation of reactionary engine manifold pressure limit protection system

As shown in figure 78 the nominal vertical velocity command from the trajectory generator to the command filter ($v_{z,c}$) is modified by the reactionary envelope protection system in order to keep the manifold pressure below the prescribed upper limit. In simulation, the vehicle is commanded to climb at a constant rate of 5 ft/sec from hover.

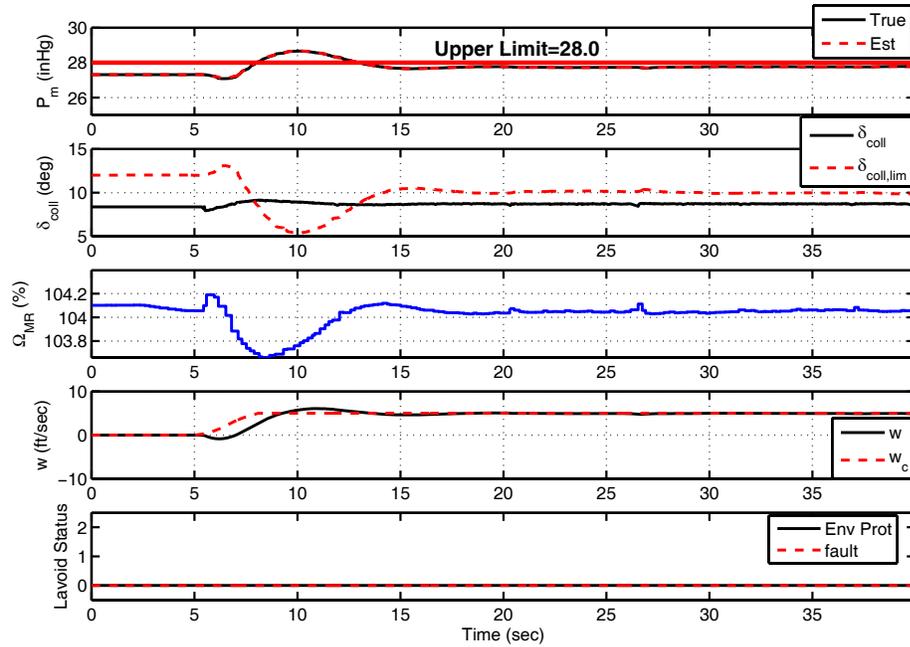


Figure 79: Engine manifold pressure response- Envelope protection OFF

In order to evaluate the reactionary engine manifold limit protection system an artificial limit of 28 inHg is imposed on the response. The engine manifold pressure response with envelope protection switched OFF violates this upper limit as shown in figure 79.

The engine manifold pressure response for the same maneuver with reactionary envelope protection system switched ON is shown in figure 80. Notice the system successfully prevents manifold pressure upper limit violations by modifying the rate of increase of vertical velocity (vertical acceleration) during the maneuver.

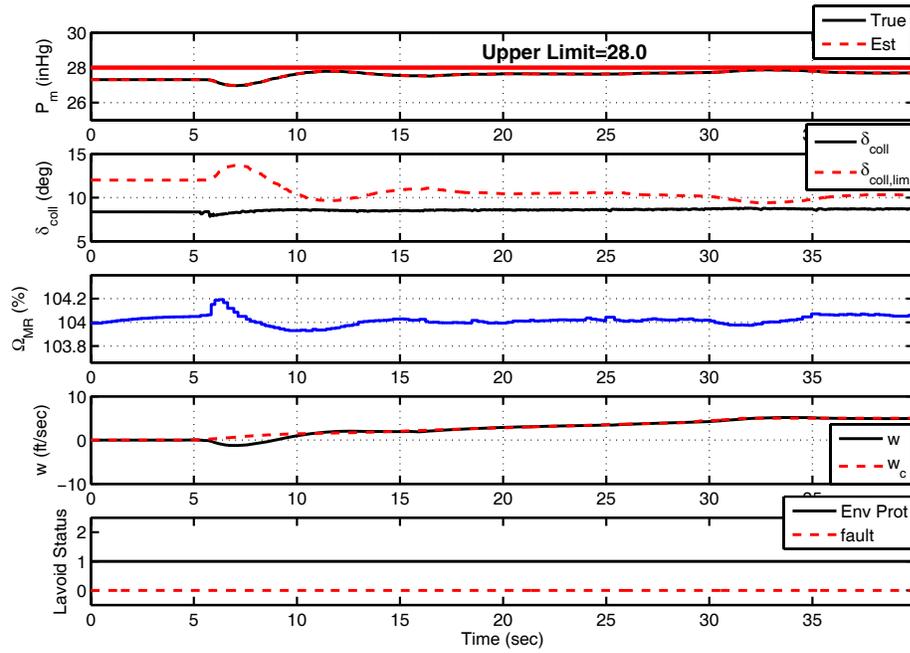


Figure 80: Engine manifold pressure response- Envelope protection ON

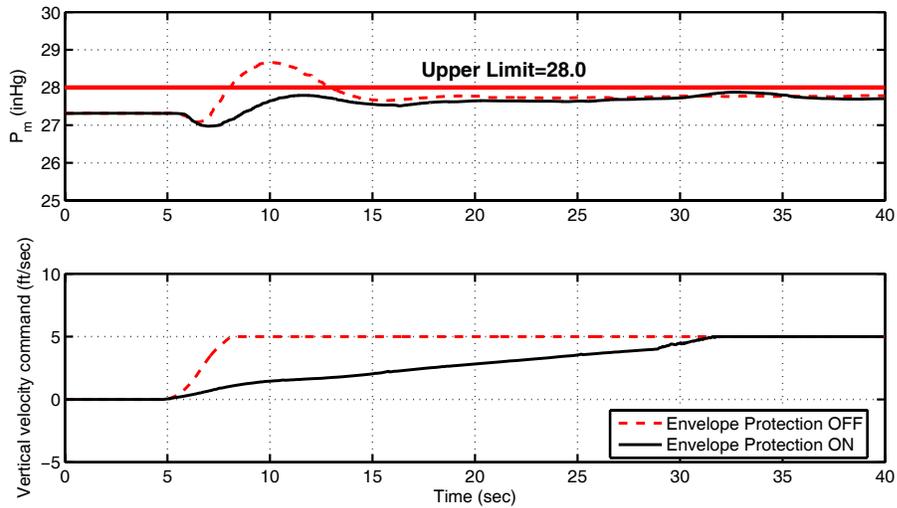


Figure 81: Engine manifold pressure response comparison with and without envelope protection

CHAPTER V

THESIS CONTRIBUTIONS, CONCLUSIONS AND RECOMMENDED FUTURE WORK

5.1 Conclusions and contributions

Envelope protection in manned systems requires timely cues to be conveyed to the pilot/operator regarding limit proximity. Limit proximity information, provided to the pilot using force-feedback tactile cueing, has been found to significantly improve vehicle handling qualities and reduce pilot workload during highly aggressive maneuvers near the limit boundary. Also, when reliable estimates of control limits corresponding to limit boundary are available, force-feedback tactile cueing for envelope protection will allow maximum utilization of the OFE.

Effective envelope protection methods must have the ability to adapt. Adaptive envelope protection methods that are able to capture changes in envelope limit parameter dynamics caused due to variations in flight and or vehicle configurations will in general perform better than static model based methods. Existing envelope protection methods either use simplified models for limit parameter dynamics or require large amounts of data to generate accurate models. These non-adaptive models may not perform well in unmodeled configurations. Adaptive dynamic trim is a recently developed approach that uses an adaptive SHL-NN based architecture to estimate limit parameter dynamics on-line. However, dynamic trim based methods can only be applied to steady-state-response critical limit parameters [see figure 5]. Nonlinear function response method is a recently developed envelope protection method that

overcomes this limitation. However, the approach relies on identifying a static input-output relationship between the current value of limit parameter and other system variables using neural networks. This static relationship is then used to estimate the non-forced maximum future limit parameter response at any given time and subsequently the control margin. This non-adaptive approach will provide a reliable but conservative estimate of the control limits.

In this thesis a new approach is developed for estimating control limits from the adaptive estimate of limit parameter dynamics. This approach is based on obtaining real-time solutions of an optimal control problem and is applicable to both steady-state and transient response critical limit parameters. The optimal control problem is formulated assuming that the pilot is typically striving to complete the maneuver in the shortest possible time (infinite aggressiveness). The objective of an envelope protection system is to curtail this very high value of pilot aggressiveness near the envelope boundary. Therefore, the objective function chosen for the optimal control problem is such that the control solution will take the limit parameter response to the limit boundary in the shortest possible time but with a finite measure of aggressiveness (Control weighting). An optimal trajectory generation package called the NTG is used to obtain real-time solutions for this nonlinear optimal control problem. The proposed approach and the NTG formulation can be used in conjunction with the adaptive estimate of limit parameter dynamics. Analysis and simulation evaluations using linear spring-mass-damper system is used to show that the area norm [see definition 67] of the optimal control solution is truly a reasonable estimate of the control limit. The proposed approach, combined with NTG package for generating real-time optimal solutions, is evaluated for imposing an artificial upper limit on a linear spring-mass-damper system. The results show that the proposed NTG based envelope protection system is effective in maintaining the limit parameter response within the prescribed limits. Finally, the NTG based envelope protection system is

implemented as a limit cueing module within the OPLP architecture for hub moment limit (transient-response critical limit) protection. The envelope protection system is evaluated using pilot-in-the-loop tests at the Georgia Tech Carefree Maneuver Lab. The simulation results showed that the NTG based envelope protection system was successful in maintaining the hub moment response within the prescribed limits. The performance of the NTG based hub moment limit protection system is comparable to that of the non-adaptive nonlinear function response method based hub moment limit protection system. Also, post-inceptor command shaping schemes (DC, FD) were utilized in augmenting the NTG based envelope protection system and were useful in providing high frequency limit protection. Safety and performance comparisons between the NTG limit cue module and existing nonlinear linear function response method implementation for hub moment limit protection is also conducted. The evaluations clearly demonstrate increase in maneuver safety with NTG hub moment protection inspite of higher pilot aggressiveness during the tests.

The second envelope protection method developed in this thesis is an automatic limit protection method proposed mainly for application within UAVs. The approach uses adaptive estimate of limit parameter dynamics and finite-time horizon predictions for detecting impending limit boundary violations. Limit violations are prevented by treating the limit boundary as an obstacle and by correcting nominal control/command inputs to track a limit parameter safe-response profile near the limit boundary. Therefore, the approach allows the designer to choose the prediction horizon for switching between passive estimation and active command/control correction for envelope protection. The approach was first demonstrated using a simple linear example. Reactionary envelope protection method is used to implement an automatic load factor limit protection system within the GTMax integrated flight and simulation architecture. A detailed description of the GTMax integrated simulation and flight test architecture is presented. The design of this envelope protection system is

shown to be independent of the overall system architecture. In the synthesis, design and implementation of reactionary load factor limit protection system, no assumption is made regarding the response type (steady-state or transient response critical). Therefore, the reactionary envelope protection method can be easily applied to both steady-state response critical and transient response critical limit parameters.

This reactionary load factor limit protection system is tested using Software-in-the-Loop and flight evaluations. Load factor upper limit (1.5g) violations during an aggressive turn maneuver are prevented by computing corrections to the nominal pitch rate command. These corrections force the load factor response to follow a safe-response profile near the limit boundary. The inner and outer loop command architecture currently used in the GTMax does not permit directly altering the pitch rate commands in the inner loop without significant changes to the control architecture. Therefore, these pitch rate command corrections are transformed into equivalent acceleration command corrections within the trajectory generator. This form of envelope protection implementation is an example for cases where limit parameter influencing command/control inputs are not directly available in achieving envelope protection. The flight test results show that the reactionary load factor limit protection system is successful in maintaining the vehicle load factor response within prescribed upper limit while executing an aggressive turn maneuver. The simulation and flight test results clearly show that the reactionary envelope protection is a promising new approach for use in future UAV envelope protection systems. The successful flight evaluation of reactionary envelope protection method on the GTMax is also significant because GTMax is a rotary-wing platform and rotorcraft operational envelope, compared to fixed-wing vehicles, is typically constrained by a large number of complex limit parameters.

The reactionary envelope protection method has also been extended to estimate control margins for force-feedback tactile cueing applications in manned vehicles. The

hub moment control margins estimated from this extension are found comparable to that obtained from the NTG and nonlinear function response method. Upon closer inspection, it is observed that the lead time available from the reactionary control margins are comparable to that of the NTG method and slightly higher than that obtained from the nonlinear function response method.

5.2 Recommended future work

1. **Development of probabilistic methods-** In this thesis analytical methods have been used to design limit cue modules. Alternative approaches, such as probabilistic modeling have not been investigated. Development of equivalent or alternative probabilistic methods for limit parameter modeling and envelope protection is a viable research topic.
2. **Adaptive synergetic design-** In manned systems, the pilot is an adaptive element. Recent control technologies, including envelope protection, have increasingly relied on using adaptive technologies for capturing more information about the vehicle and its behavior on-line. On the GTMax for example, the low level controller is an adaptive neural network controller and the load factor limit protection system is also an adaptive system. The presence of these multiple adaptive components makes overall system analysis difficult and brings into focus concerns related to interaction between these adaptive components. Future researchers should investigate how to implement multiple adaptive systems within the same architecture in a synergetic fashion. This investigation can also include issues addressing pilot adaptation and effect of limit cueing on pilot response. In the case of manned vehicles, these studies can help in characterizing the content and form of limit cue information which will help increase pilot confidence in envelope protection and adaptive systems in general.

3. **Development of envelope detection methods for envelope expansion-**

This thesis has focused on the development of new envelope protection methods. Specifically, these methods aid in the timely prevention of large and undesirable violations of the flight envelope by restricting limit parameters within prescribed boundaries. The method has assumed that either sensor measurements or estimates of limit parameters are available. However, there are many other phenomena that define an envelope boundary, for example, retreating blade stall or rotor stall, vortex ring state etc. It is very difficult to determine whether a rotor is experiencing stall based on a single parameter. Empirical studies have correlated rotor stall to a parameter referred to as ERITS (Extended Retreating Indicated Tip Speed). A low value of ERITS (below 300 ft/sec) indicates a rotor stall condition. Therefore, the methods developed in this thesis can be used to limit or prevent ERITS from getting to low values. Even though this approach will successfully limit ERITS above 300ft/sec it cannot guarantee maximum utilization of the available OFE. This is because ERITS is only empirically related to rotor stall. Development of envelope protection methods that can protect the aircraft from undesirable flight conditions such as vortex ring state or stall by detecting their onset and providing solutions for avoidance is a significant research topic.

4. **Range regulation posed as an envelope protection problem-** Recently, development of formation flight control architecture/mode has come into focus and lot of research has been directed towards the development of new control methods consistent with formation flight. The problem of range regulation or regulating subtended angle (when target size is known) can be posed an envelope protection problem with the range or subtended angle is the limit parameter. Maintaining formation flight essentially means regulating range or subtended angle with respect to other aircraft. The low level controller can be made

to accomplish this task by either providing higher level trajectory commands consistent with formation flight [27, 58, 57, 56] or providing estimates of leader maneuver using an estimator [4, 35, 55, 34]. On the other hand range regulation problem can be treated as an envelope protection problem with nominal commands to the low-level controller being modified based on information from a range limiting envelope protection system. This philosophy will avoid the process of re-designing the low-level controller for formation flight mode which in itself is a very expensive and tedious process.

5. **Robust adaptive estimation techniques-** The robustness and system properties of existing adaptive estimation and modeling techniques are not quantifiable by the same measure as current system implementations (phase margin, gain margin etc.). Therefore, there is a need to develop more robust adaptive techniques the performance measure of which can be quantified in either a similar or acceptable way as that of flight certified systems.

APPENDIX A

COMPUTATION OF B-SPLINES AND THEIR APPROXIMATION PROPERTIES

Derivation of B-spline recurrence relation

$$B_{j,k,\bar{\mathfrak{k}}} = \omega_{jk} B_{j,k-1} + (1 - \omega_{j+1,k}) B_{j+1,k-1}$$

where $\omega_{jk} \triangleq \frac{x - t_j}{t_{j+k} - t_j}$

Proof. According to the definition of B-spline-

$$B_{j,k,\bar{\mathfrak{k}}} = (t_{j+k} - t_j) [t_j, t_j + 1, \dots, t_{j+k}] (\cdot - x)_+^{k-1}$$

The Leibniz formula for divided differences can be stated as follows,

$$[t_j, t_{j+1}, \dots, t_{j+k}] gh = \sum_{r=j}^{r=j+k} ([t_j, \dots, t_r] g) ([t_r, \dots, t_{j+k}] h) \quad (183)$$

$$\therefore (t - x)_+^{k-1} = (t - x) (t - x)_+^{k-2}$$

$$[t_j, \dots, t_{j+k}] (t - x)_+^{k-1} = \sum_{r=j}^{r=j+k} ([t_j, \dots, t_r] (t - x)) [t_r, \dots, t_{j+k}] (t - x)_+^{k-2}$$

$$\begin{aligned} \implies [t_j, \dots, t_{j+k}] (t - x)_+^{k-1} &= ([t_j] (t - x)) ([t_j, \dots, t_{j+k}] (t - x)_+^{k-2}) \\ &\quad + ([t_j, t_{j+1}] (t - x)) \cdot ([t_{j+1}, \dots, t_{j+k}] (t - x)_+^{k-2}) \end{aligned} \quad (184)$$

$$\begin{aligned} &= (t_j - x) \frac{[t_{j+1}, \dots, t_{j+k}] (t - x)_+^{k-2}}{(t_{j+k} - t_j)} - \frac{[t_j, \dots, t_{j+k-1}] (t - x)_+^{k-2}}{(t_{j+k} - t_j)} \\ &\quad + 1 \cdot ([t_{j+1}, \dots, t_{j+k}] (t - x)_+^{k-2}) \end{aligned} \quad (185)$$

Substituting equation 185 into equation 90 for B-spline we get

$$\begin{aligned}
B_{j,k,\bar{\mathfrak{t}}} &= (t_{j+k} - t_j) \left(\left(1 + \frac{t_j - x}{t_{j+k} - t_j}\right) [t_{j+1}, \dots, t_{j+k}] (t - x)_+^{k-2} \right. \\
&\quad \left. - \frac{(t_j - x)}{(t_{j+k} - t_j)} [t_j, \dots, t_{j+k-1}] (t - x)_+^{k-2} \right) \\
&= \frac{t_{j+k} - x}{t_{j+k} - t_{j+1}} B_{j+1,k-1,\bar{\mathfrak{t}}} + \frac{x - t_j}{(t_{j+k} - t_j)} B_{j,k-1,\bar{\mathfrak{t}}} \\
&= (1 - \omega_{j+1,k}) B_{j+1,k-1,\bar{\mathfrak{t}}} + \omega_{jk} B_{j,k-1,\bar{\mathfrak{t}}}
\end{aligned} \tag{186}$$

where,

$$\omega_{jk} \triangleq \frac{x - t_j}{t_{j+k} - t_j} \tag{187}$$

□

Marsden's identity allows one to represent any element of $\Pi_{<k}$ in terms of B-splines. According to Marsden's identity for any $\tau \in \mathfrak{R}$

$$(\cdot - \tau)^{k-1} = \sum_j \psi_{jk}(\tau) B_{jk} \tag{188}$$

where

$$\psi_{jk} \triangleq (t_{j+1} - \tau) \dots (t_{j+k-1} - \tau) \tag{189}$$

Any $p \in \Pi_{<k}$ can be expressed in the following form using Taylor series expansion about τ .

$$p = \sum_{\nu=1}^k \frac{(\cdot - \tau)^{k-\nu}}{(k-\nu)!} D^{k-\nu} p(\tau) \tag{190}$$

Differentiating equation 188 $\nu - 1$ times we get

$$(\cdot - \tau)^{k-\nu} (-1)^{\nu-1} (k-1)(k-2)\dots(k-\nu+1) = \sum_j D^{\nu-1} \psi_{jk}(\tau) B_{jk}, \nu > 0 \tag{191}$$

Substituting equation 191 into the expression for $p \in \Pi_{<k}$ given in equation 190 the following is obtained

$$\begin{aligned}
p &= \sum_{\nu=1}^k \left(\frac{\sum_j (-D)^{\nu-1} \psi_{jk}(\tau) B_{jk}}{(k-1)!} \right) D^{k-\nu} p(\tau) \\
p &= \sum_j \left(\frac{\sum_{\nu=1}^k (-D)^{\nu-1} \psi_{jk}(\tau) D^{k-\nu} p(\tau)}{(k-1)!} \right) B_{jk} \\
p &= \sum_j \lambda_{jk} B_{jk} \quad \text{where} \quad \lambda_{jk} = \frac{\sum_{\nu=1}^k (-D)^{\nu-1} \psi_{jk}(\tau) D^{k-\nu} p(\tau)}{(k-1)!}
\end{aligned} \tag{192}$$

In the special case of $p = 1$,

$$\begin{aligned}
\lambda_{jk} &= \frac{(-D)^{k-1} \psi_{jk}(\tau) D^{k-k} p(\tau)}{(k-1)!} = 1 \\
&\quad \therefore \sum_j B_{jk} = 1
\end{aligned} \tag{193}$$

A special property of B-splines can be derived for $p \in \Pi_{<2}$ which is the following:

Proposition A.0.1. *For any $l \in \Pi_{<2}$*

$$\begin{aligned}
l &= \sum_j \lambda_{jk} B_{jk} = \sum_j \left(\frac{\sum_{\nu=1}^k (-D)^{\nu-1} \psi_{jk}(\tau) D^{k-\nu} l(\tau)}{(k-1)!} \right) B_{jk} \\
&= \sum_j \left(\frac{(-D)^{(k-1)-1} \psi_{jk}(\tau) D^{k-(k-1)} l(\tau)}{(k-1)!} + \frac{(-D)^{k-1} \psi_{jk}(\tau) D^{k-k} l(\tau)}{(k-1)!} \right) B_{jk}
\end{aligned} \tag{194}$$

Note that

$$\begin{aligned}
\psi_{jk} &= (t_{j+1} + \dots + t_{j+k-1}) (-1)^{(k-2)} \tau^{(k-2)} + (-1)^{(k-1)} \tau^{k-1} \\
\implies (-D)^{k-2} \psi_{jk}(\tau) &= (k-2)! (t_{j+1} + \dots + t_{j+k-1}) - \tau (k-1)! \\
&\quad \therefore l = \sum_j \left(\frac{t_{j+1} + \dots + t_{j+k-1}}{k-1} - \tau + l(\tau) \right) B_{jk} \\
\implies l &= \sum_j l(t_{jk}^*) B_{jk} \quad \text{where} \quad t_{jk}^* = \frac{t_{j+1} + \dots + t_{j+k-1}}{k-1}
\end{aligned} \tag{195}$$

The sites t_{jk}^* are called **Greville sites**.

A.0.1 Control points and control polygon

B-spline coefficients model the function that they represent. Let B-splines be used to represent a function within an interval $[a, b]$. Then the knot sequence $(\mathbf{t} = (\mathbf{t}_i)_1^{\mathbf{n}+\mathbf{k}})$ is chosen from the breakpoints using the Curry-Schoenberg theorem. Note that the continuity conditions are imposed only at the interior points.

In the knot sequence, $t_1 = t_2 = \dots = t_k = a$, $t_{n+1} = t_{n+2} = \dots = t_{n+k} = b$. Let the function be approximated using B-splines or B-form of the function be given as:

$$f := \sum_j \alpha_j B_{j,k,\bar{\mathbf{t}}} \quad (196)$$

The value of the function at $t_i \leq x \leq t_{i+1}$ will only depend on the non-zero B-splines passing through x .

$$f(x) = \sum_{j=i-k+1}^{j=i} \alpha_j B_{j,k,\bar{\mathbf{t}}} \quad (197)$$

This is because $B_{i-k,k,\bar{\mathbf{t}}}$ is non-zero only in the interval $[t_{i-k}, t_{i-k+k}]$ and $B_{i+1,k,\bar{\mathbf{t}}}$ is non-zero only in the interval $[t_{i+1}, t_{i+1+k}]$. Therefore, the only non-zero B-splines passing through point x are $B_{i-k+1,k,\bar{\mathbf{t}}}, \dots, B_{i,k,\bar{\mathbf{t}}}$. The above equation for $f(x)$ can also be used to conclude the following inequality:

$$\min\{\alpha_{i-k+1}, \dots, \alpha_i\} \leq f(x) \leq \max\{\alpha_{i-k+1}, \dots, \alpha_i\} \quad \text{for } t_i \leq x \leq t_{i+1} \quad (198)$$

The inequality above states that the value of function B-form $f = \sum_j \alpha_j B_{j,k,\bar{\mathbf{t}}}$ on the interval $[t_i, t_{i+1}]$ is bounded, from above and below, by the k B-spline coefficients “nearby”. This close relationship between the value of a spline and the “nearby” B-spline coefficients has led to the definition and use of *control point*. This concept is developed as follows:

$$x = \sum_j t_{jk}^* B_{j,k,\bar{\mathbf{t}}}, \quad x \in [a, b] \quad (199)$$

It was proved earlier that for any $l \in \Pi_{<2}$, $l = \sum_j l(t_{jk}^*) B_{j,k,\bar{\mathbf{t}}}$ where t_{jk}^* are Greville sites. When $f \in \Pi_{<2}$ then $\alpha_j = f(t_{jk}^*)$. In general however, this is not true. Define

the *control sequence* as the following:

$$(P_j := (t_{jk}^*, \alpha_j) \in \mathfrak{R}^2 : j = 1, 2, \dots, n) \quad (200)$$

The *control polygon* $C_{k, \bar{\mathbf{t}}}$ of a spline $f \in \mathfrak{S}_{k, \bar{\mathbf{t}}}$ is the broken line with the spline's control point sequence as vertices. There is a close connection between the spline and its control polygon. The control polygon is an exaggerated version or caricature of the spline which allows one to easily identify certain important features of the spline such as regions of convexity/concavity, zeros etc. It is important however, to investigate the relationship between the value of the spline at Greville site and α_j . They are the same when $f \in \Pi_{<2}$ but otherwise the relationship is quantified as follows:

Proposition A.0.2. *If a spline $f \in \mathfrak{S}$ is continuously differentiable, then*

$$|\alpha_j - f(t_{jk}^*)| \leq \text{const}_k |\bar{\mathbf{t}}|^2 \|D^2 f\|_{\|t_{j+1} \dots t_{j+k-1}\|} \quad (201)$$

Proposition A.0.2 can be used to conclude that, for moderate k the sequence α of B-spline coefficients (or, more precisely the control polygon) for a spline function f gives a fair idea of the graph of f . Proposition A.0.2 and Curry-Schoenberg theorem can be used to deduce that $\mathfrak{S}_{k, \bar{\mathbf{t}}}$ is a subspace of $\mathfrak{S}_{k, \hat{\mathbf{t}}}$ for any knot sequence $\hat{\mathbf{t}}$ that is a *refinement* of $\bar{\mathbf{t}}$, i.e,

$$\bar{\mathbf{t}} \subset \hat{\mathbf{t}} \implies \mathfrak{S}_{k, \bar{\mathbf{t}}} \subset \mathfrak{S}_{k, \hat{\mathbf{t}}} \quad (202)$$

Also, if one were to rewrite $f \in \mathfrak{S}_{k, \bar{\mathbf{t}}}$ as a spline using a *refined* knot sequence $\hat{\mathbf{t}}$ (say by mid-point refinement thereby reducing the mesh size) then, the control polygon $C_{k, \hat{\mathbf{t}}} f$ will move closer to the spline. A refined knot sequence will reduce $|\hat{\mathbf{t}}|$ in proposition A.0.2. This has immediate appeal for generating a computer graph of spline in applications such as Computer Aided Geometric design since most graphing programs only plot broken lines.

A.0.2 Schoenberg's variation diminishing spline approximation

The “shape preserving” spline approximation of a function g on $[a, b]$ is defined by

$$Vg \triangleq \sum_{i=1}^n g(t_{ik}^* B_{i,k,\bar{t}}) \quad \text{on } [a, b] \quad (203)$$

Let $S^{-}\alpha$ denote the number of sign changes in the sequence α . Then according to Schoenberg- “The number of sign changes in the spline function $\sum_j B_{j,k,\bar{t}}$ is not bigger than the number of sign changes in its B-spline coefficient sequence α ”, i.e.,

$$S^{-}\left(\sum_j \alpha_j B_{j,k,\bar{t}}\right) \leq S^{-}\alpha \quad (204)$$

By applying this property deduced by Schoenberg to the “shape preserving” spline approximation and also since the sign changes in $g(t_{ik}^* B_{i,k,\bar{t}})$ will always be less than or equal to the actual number of sign changes in g we get:

$$S^{-}Vg \leq S^{-}(g(t_{ik}^* B_{i,k,\bar{t}})) \leq S^{-}g \quad (205)$$

The case $l \in \Pi_{<2}$ is special for the shape preserving spline approximation because then,

$$Vl = \sum_{i=1}^n l(t_{ik}^* B_{i,k,\bar{t}}) = l \quad \text{for all straight lines } l \quad (206)$$

Using equations 205 and 206 it can be concluded that

$$S^{-}(Vg - l) \leq S^{-}(g - l) \quad \forall \quad l \in \Pi_{<2} \quad (207)$$

Equation 207 can be used to conclude that the maximum number of times that the shape preserving spline approximation Vg crosses any particular line l will be bounded by actual number of crossings by the true function g . Therefore, if the function g is always positive or always negative then so is the shape preserving spline approximation Vg . The derivative of a spline function $\sum_j \alpha_j B_{j,k,\bar{t}}$ can be found using the following formulae:

$$D\left(\sum_j \alpha_j B_{j,k,\bar{t}}\right) = (k-1) \sum_j \frac{\alpha_j - \alpha_{j-1}}{t_{j+k-1} - t_j} B_{j,k-1,\bar{t}} \quad (208)$$

From equations 205, 206 and 208 it can be deduced that if g is a monotone non-decreasing function then

$$\begin{aligned} D(Vg) &= (k-1) \sum_j \frac{\alpha_j - \alpha_{j-1}}{t_{j+k-1} - t_j} B_{j,k-1,\bar{\mathbf{t}}} \\ &= (k-1) \sum_j \frac{g(t_{jk}^*) - g(t_{j-1,k}^*)}{t_{j+k-1} - t_j} B_{j,k-1,\bar{\mathbf{t}}} \geq 0 \quad \because g(t_{jk}^*) \geq g(t_{j-1,k}^*) \end{aligned} \quad (209)$$

Therefore, Vg is also monotone non-decreasing. Similarly, if g is a convex function then,

$$\begin{aligned} D^2(Vg) &= D(D(Vg)) = D\left((k-1) \sum_j \frac{\alpha_j - \alpha_{j-1}}{t_{j+k-1} - t_j} B_{j,k-1,\bar{\mathbf{t}}}\right) \\ &= D\left((k-1) \sum_j \frac{g(t_{jk}^*) - g(t_{j-1,k}^*)}{t_{j+k-1} - t_j} B_{j,k-1,\bar{\mathbf{t}}}\right) \\ &= D\left((k-1) \sum_j \beta_j B_{j,k-1,\bar{\mathbf{t}}}\right) \\ &= (k-1)(k-2) \sum_j \frac{\beta_j - \beta_{j-1}}{t_{j+k-2} - t_j} B_{j,k-2,\bar{\mathbf{t}}} \\ &= \sum_j \frac{(k-1)(k-2)}{t_{j+k-2} - t_j} \left(\frac{g(t_{jk}^*) - g(t_{j-1,k}^*)}{t_{j+k-1} - t_j} - \frac{g(t_{j-1,k}^*) - g(t_{j-2,k}^*)}{t_{j-1+k-1} - t_{j-1}} \right) B_{j,k-2,\bar{\mathbf{t}}} \\ &\geq 0 \quad \because \beta_j \geq \beta_{j-1} \end{aligned} \quad (210)$$

Therefore, if g is a convex function so is the shape preserving spline approximation Vg . Note that Vg the shape preserving spline approximation is only a *linear* approximation of the original spline approximation and therefore cannot provide as good approximation to a smooth function as splines are capable of providing. In fact, even if g has m continuous derivatives for some $m \geq 2$,

$$\|g - Vg\| \leq \text{const}_{g,k} |\mathbf{t}|^2 \quad (211)$$

The order of approximation cannot be improved further by using the shape preserving spline approximation.

APPENDIX B

TRACE IDENTITIES

Prove that $tr(\mathbf{AB}) \leq \|\mathbf{A}\|_F \|\mathbf{B}\|_F$

Proof. Let $\mathbf{A} \in \mathfrak{R}^{m \times n}$ and $\mathbf{C} = \mathbf{A}^T \mathbf{A}$. Then,

$$tr(\mathbf{A}^T \mathbf{A}) = tr(\mathbf{C}) \tag{212}$$

$$= \sum_{i=1}^n c_{ii} \tag{213}$$

$$= \sum_{i=1}^n \sum_{k=1}^m a_{ki} a_{ki} \tag{214}$$

$$= \|\mathbf{A}\|_F^2 \tag{215}$$

Show that $tr(\mathbf{AB}) \leq \|\mathbf{A}\|_F \|\mathbf{B}\|_F$ where $\mathbf{B} \in \mathfrak{R}^{n \times m}$. Let $\mathbf{C} = \mathbf{AB}$.

$$tr(\mathbf{AB}) = tr(\mathbf{C}) \tag{216}$$

$$= \sum_{i=1}^m c_{ii} \tag{217}$$

$$= \sum_{i=1}^m \sum_{k=1}^n a_{ik} b_{ki} \tag{218}$$

$$\leq \sum_{i=1}^m \left(\sum_{k=1}^n a_{ik}^2 \right)^{\frac{1}{2}} \left(\sum_{k=1}^n b_{ki}^2 \right)^{\frac{1}{2}} \quad \text{Cauchy-Schwartz inequality} \tag{219}$$

$$\leq \left[\sum_{i=1}^m \sum_{k=1}^n a_{ik}^2 \right]^{\frac{1}{2}} \left[\sum_{i=1}^m \sum_{k=1}^n a_{ik}^2 \right]^{\frac{1}{2}} \quad \text{Cauchy-Schwartz inequality} \tag{220}$$

$$= \|\mathbf{A}\|_F \|\mathbf{B}\|_F \tag{221}$$

Therefore, $tr(\mathbf{AB}) \leq \|\mathbf{A}\|_F \|\mathbf{B}\|_F$. □

APPENDIX C

ANALYSIS OF SPRING-MASS-DAMPER SYSTEM FOR APPLICATION OF NONLINEAR FUNCTION RESPONSE METHOD

Consider a general linear spring-mass-damper given below:

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\omega_n^2 & -2\zeta\omega_n \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ b \end{bmatrix} u \quad (222)$$

$$y_p = x_1 = \mathbf{C}\mathbf{x} = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} x_1 & x_2 \end{bmatrix} \quad (223)$$

Taking Laplace transform

$$\begin{aligned} s\mathbf{X}(s) - \mathbf{X}(0) &= \mathbf{A}\mathbf{X}(s) + \mathbf{B}u(s) \\ \implies (s\mathbf{I} - \mathbf{A})\mathbf{X}(s) &= \mathbf{X}(0) + \mathbf{B}u(s) \\ \therefore \mathbf{X}(s) &= (s\mathbf{I} - \mathbf{A})^{-1} \left(\mathbf{X}(0) + \mathbf{B}u(s) \right) \end{aligned} \quad (224)$$

Now,

$$\begin{aligned} (s\mathbf{I} - \mathbf{A})^{-1} &= \left(\begin{bmatrix} s & 0 \\ 0 & s \end{bmatrix} - \begin{bmatrix} 0 & 1 \\ -\omega_n^2 & -2\zeta\omega_n \end{bmatrix} \right)^{-1} \\ &= \begin{bmatrix} s & -1 \\ \omega_n^2 & s + 2\zeta\omega_n \end{bmatrix}^{-1} \\ &= \frac{1}{s^2 + 2\zeta\omega_n s + \omega_n^2} \begin{bmatrix} (s + 2\zeta\omega_n) & -\omega_n^2 \\ 1 & s \end{bmatrix}^T \end{aligned}$$

$$\therefore (s\mathbf{I} - \mathbf{A})^{-1} = \frac{1}{s^2 + 2\zeta\omega_n s + \omega_n^2} \begin{bmatrix} (s + 2\zeta\omega_n) & 1 \\ -\omega_n^2 & s \end{bmatrix}. \text{ Also,}$$

$$\begin{aligned} y_p(s) &= \mathbf{C}\mathbf{X}(s) \\ &= \mathbf{C}(s\mathbf{I} - \mathbf{A})^{-1} \left(\mathbf{X}(0) + \mathbf{B}u(s) \right) \end{aligned}$$

$$\begin{aligned} \mathbf{C}(s\mathbf{I} - \mathbf{A})^{-1} &= \begin{bmatrix} 1 & 0 \end{bmatrix} \frac{1}{s^2 + 2\zeta\omega_n s + \omega_n^2} \begin{bmatrix} (s + 2\zeta\omega_n) & 1 \\ -\omega_n^2 & s \end{bmatrix} \\ &= \frac{1}{s^2 + 2\zeta\omega_n s + \omega_n^2} \begin{bmatrix} s + 2\zeta\omega_n & 1 \end{bmatrix} \end{aligned}$$

$$\begin{aligned} \mathbf{C}(s\mathbf{I} - \mathbf{A})^{-1}\mathbf{B} &= \frac{1}{s^2 + 2\zeta\omega_n s + \omega_n^2} \begin{bmatrix} s + 2\zeta\omega_n & 1 \end{bmatrix} \begin{bmatrix} 0 \\ b \end{bmatrix} \\ &= \frac{b}{s^2 + 2\zeta\omega_n s + \omega_n^2} \end{aligned}$$

$$\therefore y_p(s) = \frac{s + 2\zeta\omega_n}{s^2 + 2\zeta\omega_n s + \omega_n^2} x_1(0) + \frac{1}{s^2 + 2\zeta\omega_n s + \omega_n^2} x_2(0) + \frac{b}{s^2 + 2\zeta\omega_n s + \omega_n^2} u(s)$$

Note the following Laplace transforms,

$$\begin{aligned} \mathcal{L}^{-1} \left(\frac{1}{s^2 + 2\zeta\omega_n s + \omega_n^2} \right) &= \mathcal{L}^{-1} \left(\frac{1}{(s + \zeta\omega_n)^2 + \omega_n^2(1 - \zeta^2)} \right) \\ &= \frac{1}{\omega_d} \exp(-\zeta\omega_n t) \sin \omega_d t \\ \omega_d &\triangleq \omega_n \sqrt{(1 - \zeta^2)} \end{aligned}$$

$$\mathcal{L}^{-1} \left(\frac{\omega_d}{(s + \zeta\omega_n)^2 + \omega_d^2} \right) = \exp(-\zeta\omega_n t) \sin \omega_d t$$

Similarly,

$$\begin{aligned} \mathcal{L}^{-1} \left(\frac{s + 2\zeta\omega_n}{s^2 + 2\zeta\omega_n s + \omega_n^2} \right) &= \mathcal{L}^{-1} \left(\frac{s + \zeta\omega_n}{s^2 + 2\zeta\omega_n s + \omega_n^2} \right) + \mathcal{L}^{-1} \left(\frac{\zeta\omega_n}{s^2 + 2\zeta\omega_n s + \omega_n^2} \right) \\ &= \exp(-\zeta\omega_n t) \cos \omega_d t + \frac{\zeta\omega_n}{\omega_d} \exp(-\zeta\omega_n t) \sin \omega_d t \\ &= \exp(-\zeta\omega_n t) \left[\cos \omega_d t + \frac{\zeta\omega_n}{\omega_n \sqrt{(1 - \zeta^2)}} \exp(-\zeta\omega_n t) \sin \omega_d t \right] \\ &= \frac{\exp(-\zeta\omega_n t)}{\sqrt{(1 - \zeta^2)}} \sin(\omega_d t + \phi) \end{aligned}$$

where,

$$\begin{aligned}\tan \phi &= \frac{\sqrt{(1-\zeta^2)}}{\zeta} \\ \therefore y_p(t) &= \frac{\exp(-\zeta\omega_n t)}{\sqrt{(1-\zeta^2)}} \sin(\omega_d t + \phi)x_1(0) + \frac{1}{\omega_d} \exp(-\zeta\omega_n t) \sin(\omega_d t)x_2(0) \\ &\quad + \frac{b}{\omega_d} \exp(-\zeta\omega_n t) \sin(\omega_d t)u(t)\end{aligned}\quad (225)$$

Equation 225 is of the form

$$y_p(t) = Q(x, t) + H(x, t)u(t)$$

where $Q(x, t) = \frac{\exp(-\zeta\omega_n t)}{\sqrt{(1-\zeta^2)}} \sin(\omega_d t + \phi)x_1(0) + \frac{1}{\omega_d} \exp(-\zeta\omega_n t) \sin(\omega_d t)x_2(0)$ and $H(x, t) = \frac{b}{\omega_d} \exp(-\zeta\omega_n t) \sin(\omega_d t)$.

The maximum and minimum values of $\frac{\exp(-\zeta\omega_n t)}{\sqrt{(1-\zeta^2)}} \sin(\omega_d t + \phi)$ are $\frac{\exp(-\zeta\omega_n t_{1,max})}{\sqrt{(1-\zeta^2)}} \sin(\omega_d t_{1,max} + \phi)$ and $\frac{\exp(-\zeta\omega_n t_{1,min})}{\sqrt{(1-\zeta^2)}} \sin(\omega_d t_{1,min} + \phi)$ respectively where $t_{1,max} = 0.0$ and $t_{1,min} = \frac{\pi}{\omega_d}$. Similarly, the maximum and minimum values of $\frac{1}{\omega_d} \exp(-\zeta\omega_n t) \sin(\omega_d t)$ are $\frac{1}{\omega_d} \exp(-\zeta\omega_n t_{2,max}) \sin(\omega_d t_{2,max})$ and $\frac{1}{\omega_d} \exp(-\zeta\omega_n t_{2,min}) \sin(\omega_d t_{2,min})$ respectively where $t_{2,max} = \frac{\phi}{\omega_d}$ and $t_{2,min} = \frac{\pi + \phi}{\omega_d}$.

C.0.3 Maxima and minimas of : $Q(x, t)$

The necessary condition at extremum of $Q(x, t)$ is:

$$\begin{aligned}\dot{Q}(x, t) &= 0 \\ &= \frac{d}{dt} \left(\frac{\exp(-\zeta\omega_n t)}{\sqrt{(1-\zeta^2)}} \sin(\omega_d t + \phi)x_1(0) + \frac{1}{\omega_d} \exp(-\zeta\omega_n t) \sin(\omega_d t)x_2(0) \right) \\ &= \frac{x_1(0)}{\sqrt{(1-\zeta^2)}} \left[-\zeta\omega_n \exp(-\zeta\omega_n t) \sin(\omega_d t + \phi) + \omega_d \exp(-\zeta\omega_n t) \cos(\omega_d t + \phi) \right] \\ &\quad + \frac{x_2(0)}{\omega_d} \left[-\zeta\omega_n \exp(-\zeta\omega_n t) \sin(\omega_d t) + \omega_d \exp(-\zeta\omega_n t) \cos(\omega_d t) \right] \\ \implies &\frac{x_1(0)}{\sqrt{(1-\zeta^2)}} \left[-\zeta \sin(\omega_d t + \phi) + \sqrt{(1-\zeta^2)} \cos(\omega_d t + \phi) \right] \\ &\quad + \frac{x_2(0)}{\omega_d} \left[-\zeta \sin(\omega_d t) + \sqrt{(1-\zeta^2)} \cos(\omega_d t) \right] = 0\end{aligned}$$

$$\because \tan \phi = \frac{\sqrt{(1-\zeta^2)}}{\zeta} \quad \cos \phi = \zeta \text{ and } \sin \phi = \sqrt{(1-\zeta^2)}.$$

$$\begin{aligned} \therefore \frac{x_1(0)}{\sqrt{(1-\zeta^2)}} & \left[-\zeta \left(\sin(\omega_d t) \cos \phi + \cos(\omega_d t) \sin \phi \right) + \sqrt{(1-\zeta^2)} \left(\cos(\omega_d t) \cos \phi - \sin(\omega_d t) \sin \phi \right) \right] \\ & + \frac{x_2(0)}{\omega_d} \left[-\zeta \sin(\omega_d t) + \sqrt{(1-\zeta^2)} \cos(\omega_d t) \right] = 0 \end{aligned}$$

Collecting $\sin \omega_d t$ and $\cos \omega_d t$ terms we get,

$$\begin{aligned} & \sin(\omega_d t) \left[-\zeta \frac{x_2(0)}{\omega_d} + \frac{x_1(0)}{\sqrt{(1-\zeta^2)}} \left(-\zeta \cos \phi - \sqrt{(1-\zeta^2)} \sin \phi \right) \right] \\ & + \cos(\omega_d t) \left[\frac{x_2(0)}{\omega_d} \sqrt{(1-\zeta^2)} + \frac{x_1(0)}{\sqrt{(1-\zeta^2)}} \left(-\zeta \sin \phi + \sqrt{(1-\zeta^2)} \cos \phi \right) \right] = 0 \\ \implies & \sin(\omega_d t) \left(-\zeta \frac{x_2(0)}{\omega_d} + \frac{x_1(0)}{\sqrt{(1-\zeta^2)}} \right) + \cos(\omega_d t) \frac{x_2(0)}{\omega_d} \sqrt{(1-\zeta^2)} = 0 \\ \therefore & -\zeta \cos \phi - \sqrt{(1-\zeta^2)} \sin \phi = 1 \\ & -\zeta \sin \phi + \sqrt{(1-\zeta^2)} \cos \phi = 0 \end{aligned}$$

Let $\tan \theta \triangleq \frac{\frac{x_2(0)}{\omega_d} \sqrt{(1-\zeta^2)}}{-\zeta \frac{x_2(0)}{\omega_d} + \frac{x_1(0)}{\sqrt{(1-\zeta^2)}}}$ then,

$$\begin{aligned} \dot{Q}(x, t) = 0 & \quad \text{when} \quad \sin(\omega_d t_e + \theta) = 0 \\ \implies \omega_d t_e + \theta & = n\pi, \quad n = 0, 1, 2, \dots \\ \therefore t_e & = \frac{n\pi - \theta}{\omega_d}, \quad n = 0, 1, 2, \dots \end{aligned}$$

However, the above extremum time values do not include the $t = 0$ which could happen to be global maxima or minima. Therefore, the search for $\max[Q(x, t)]$ or $\min[Q(x, t)]$ should include points $t_e = 0, \frac{n\pi - \theta}{\omega_d}, \quad n = 0, 1, 2, \dots$ until $\frac{n\pi - \theta}{\omega_d} > 5.0$ seconds.

REFERENCES

- [1] AIKEN, E. W., ORMISTON, R. A., and YOUNG, L. A., “Future directions in rotorcraft technology at ames research center,” in *Proceedings of American Helicopter Society 56th Annual Forum*, vol. 1, (Virginia Beach, VA), pp. 10–30, May 2000.
- [2] BATEMAN, A. J., WARD, D. G., BARRON, R. L., and WHALLEY, M. S., “Piloted simulation evaluation of neural network limit avoidance system for rotorcraft,” *AIAA*, May 1999. A99-36928,AIAA 99-4252.
- [3] BRYSON JR, A. E., *Dynamic Optimization*. Addison-Wesley, 1999.
- [4] CALISE, A., JOHNSON, E., SATTIGERI, R., WATANABE, Y., and MADYASTHA, V., “Estimation and guidance strategies for vision based target tracking,” *Proceedings of the American Control Conference*, vol. 7, pp. 5079 – 5084, June 2005. Portland, OR.
- [5] CORPS, S. G., “Airbus a320 side stick and fly by wire - an update,” in *Proceedings - Society of Automotive Engineers*, pp. 149–161, 1987. Long Beach, CA.
- [6] DE. BOOR, C., *Practical Guide to Splines*. Springer New York, 1978.
- [7] EINTHOVEN, P. and CHAN, M., “Energy management,” in *Proceedings of the AHS Flight Control and Crew Design Technical Specialists Meeting*, (Philadelphia, PA), Oct. 2002.
- [8] FUNAHASHI, K., “On the approximate realization of continuous mappings by neural networks,” *Neural Networks*, vol. 2, pp. 183–192, 1989.
- [9] HORN, J., CALISE, A. J., and PRASAD, J. V. R., “Flight envelope cueing on a tilt-rotor aircraft using neural network limit prediction,” in *Proceedings of American Helicopter Society 54th Annual Forum*, vol. 2, (Washington, D.C), pp. 1093–1104, May 1998.
- [10] HORN, J., CALISE, A. J., and PRASAD, J. V. R., “Flight envelope limiting systems using neural networks,” in *Proceedings of AIAA Atmospheric Flight Mechanics Conference*, (Boston, MA), May 1998. AIAA-1998-4459,A98-37204 10-08.
- [11] HORN, J., CALISE, A. J., and PRASAD, J. V. R., “Development of envelope protection systems for rotorcraft,” in *Proceedings of American Helicopter Society 55th Annual Forum*, vol. 2, (Washington, D.C), pp. 2025–2036, May 1999.

- [12] HORN, J., CALISE, A. J., and PRASAD, J. V. R., “Flight envelope cueing on a tiltrotor aircraft using neural network limit prediction,” *Journal of the American Helicopter Society*, vol. 46, pp. 23–31, January 2001.
- [13] HORN, J., CALISE, A. J., and PRASAD, J. V. R., “Flight envelope limit detection and avoidance for rotorcraft,” *Journal of the American Helicopter Society*, vol. 47, pp. 253–262, October 2002.
- [14] HORN, J., *Flight Envelope Limit Detection and Avoidance*. PhD thesis, Georgia Institute of Technology, School of Aerospace Engineering, May 1999.
- [15] HORN, J. F. and SAHANI, N., “Detection and avoidance of main rotor hub moment limits on rotorcraft,” *Journal of Aircraft*, vol. 41, pp. 372–379, March/April 2004.
- [16] HOVAKIMYAN, N., CALISE, A. J., and KIM, N., “Adaptive output feedback control of a class of multi-input multi-output systems using neural networks,” *International Journal of Control*, vol. 77, no. 15, pp. 1318–1329, 2004.
- [17] HOVAKIMYAN, N., NARDI, F., CALISE, A. J., and KIM, N., “Adaptive output feedback control of uncertain systems using single hidden layer neural networks,” *IEEE Transactions in Neural Networks*, vol. 13, pp. 1420–1431, Nov. 2002.
- [18] HOWITT, J., “Carefree handling for super-agility,” in *American Helicopter Society Aeromechanics Specialist Conference*, (Fairfield County, CT), pp. 4–58 4–69, October 1995.
- [19] IDAN, M., CALISE, A., and HOVAKIMYAN, N., “An adaptive output feedback control methodology: Theory and practical implementation aspects,” in *AIAA Guidance, Navigation, and Control Conference and Exhibit*, August 2002. Monterey, CA.
- [20] INTERNATIONAL, H. A., “Robinson helicopter company issues r-22 safety alert.” Helicopter Association International available online @ <http://www.rotor.com/article.php?sid=1563&mode=thread&order=0>, date accessed May 16th 2006.
- [21] JERAM, G., “Open design for helicopter active control systems,” in *Proceedings of American Helicopter Society 58th Annual Forum*, vol. 2, (Montreal, Canada), pp. 2435–2454, June 2002.
- [22] JERAM, G. J., SAHANI, N., PRASAD, J., and HORN, J., “Distributing limit protection between autonomous restraint and voluntary tactile cues,” in *43rd AIAA Aerospace Sciences Meeting and Exhibit*, (Reno, Nevada), January 2005.
- [23] JERAM, G. J. J., *Open Platform for Limit Protection with Carefree Maneuver Application*. PhD thesis, Georgia Institute of Technology, School of Aerospace Engineering, December 2004.

- [24] JOHNSON, E. N. and KANNAN, S. K., “Adaptive flight control for an autonomous unmanned helicopter,” in *AIAA Guidance, Navigation, and Control Conference and Exhibit*, (Monterey, CA), August 2002.
- [25] JOHNSON, E. N. and SCHRAGE, D., “The georgia tech unmanned aerial research vehicle:gtmax,” in *AIAA Guidance, Navigation, and Control Conference and Exhibit*, (Austin, TX), August 2003.
- [26] KANNAN, S. K. and JOHNSON, E. N., “Adaptive trajectory based flight control for autonomous helicopters,” in *Proceedings of the 21st Digital Avionics Systems Conference*, (Irvine, CA), October 2002.
- [27] KIM, B., CALISE, A., and SATTIGERI, R., “Adaptive, integrated guidance and control design for line-of-sight based formation flight,” in *AIAA Guidance, Navigation, and Control Conference and Exhibit*, (Keystone, CO), August 2006.
- [28] KIM, N., *Improved Methods in Neural Network-Based Adaptive Output Feedback Control with Applications to Flight Control*. PhD thesis, Georgia Institute of Technology, School of Aerospace Engineering, November 2003.
- [29] K.KHALIL, H., *Nonlinear Systems*. Prentice Hall, 2002.
- [30] KUTAY, A. T., CALISE, A. J., and HOVAKIMYAN, N., “Adaptive output feedback control with reduced sensitivity to sensor noise,” in *Proceedings of the American Control Conference*, (Denver, CO), June 2003.
- [31] KUTAY, A. T., *Neural Network Based Adaptive Output Feedback Control: Applications and Improvements*. PhD thesis, Georgia Institute of Technology, School of Aerospace Engineering, December 2005.
- [32] LAVRETSKY, E. and HOVAKIMYAN, N., “Reconstruction of continuous-time dynamics using delayed outputs and feedforward neural networks,” *IEEE Transactions on Automatic Control*, 2002. Submitted.
- [33] LOY, K., “Carefree handling and its applications to military helicopters and mission,” in *Proceedings of American Helicopter Society 53rd Annual Forum*, vol. 1, (Virginia Beach, VA), pp. 489–494, May 1997.
- [34] MADYASTHA, V., *Adaptive Estimation for Control of Uncertain Nonlinear Systems with Applications to Target Tracking*. PhD thesis, Georgia Institute of Technology, School of Aerospace Engineering, December 2005.
- [35] MADYASTHA, V. and CALISE, A., “An adaptive filtering approach to target tracking,” *Proceedings of the American Control Conference*, vol. 2, pp. 1269 – 1274, June 2005. Portland, OR.
- [36] MASSEY, C., “Carefree handling systems for helicopters,” *Journal of Aerospace Engineering. Part G*, no. 206, pp. 37–46, 1992.

- [37] MAYO, J. R. and IANNACCI, V. N., “Rah-66 comanche flight envelope cueing system,” in *Proceedings of American Helicopter Society 55th Annual Forum*, vol. 2, (Montreal, Canada), pp. 2010–2024, May 1999.
- [38] MENON, P. K., IRAGAVARAPU, V. R., and WHALLEY, M. S., “Estimation of rotorcraft limit envelopes using neural networks,” in *Proceedings of American Helicopter Society 52nd Annual Forum*, vol. 2, (Washington, D.C), pp. 1423–1431, June 1996.
- [39] MILAM, M. B., *Real-Time Optimal Trajectory Generation for Constrained Dynamical Systems*. PhD thesis, California Institute of Technology, Control and Dynamical Systems, May 2003.
- [40] MILAM, M. B., MUSHAMBI, K., and MURRAY, R. M., “A new computational approach to real-time trajectory generation for constrained mechanical systems,” in *Proceedings of the 39th IEEE Conference on Decision and Control*, vol. 1, pp. 845–851, December 2000. http://www.cds.caltech.edu/%7Emurray/papers/2000f_mmm00-cdc.html, date accessed May 16th 2006.
- [41] MILLER, D. G., G.EINTHOVEN, P., MORSE, C. S., and WOOD, J., “Hact flight control system (hfcs) control law overview,” in *Proceedings of American Helicopter Society 58th Annual Forum*, vol. 1, pp. 867–884, June 2002. Montreal, Canada.
- [42] NATIONAL RESEARCH COUNCIL, *Aviation Safety and Pilot Control- Understanding and Preventing Unfavorable Pilot-Vehicle Interactions*. National Academy Press, 1997.
- [43] NICHOLAS, J. S., CLARK, B., WARD, D., JERAM, G., and PRASAD, J. V. R., “Development of artificial neural networks for the hact flight control system,” in *Proceedings of American Helicopter Society 58th Annual Forum*, vol. 2, (Montreal, Canada), June 2002.
- [44] PADFIELD, G. D., *Helicopter Flight Dynamics: The Theory and Application of Flying Qualities and Simulation Modeling*. AIAA Education Series, 1996.
- [45] PAUNICKA, J., MENDEL, B., and CORMAN, D., “The ocp - an open middleware solution for embedded systems,” *Proceedings of the American Control Conference*, vol. 5, pp. 3445–3450, June 2001. Arlington, VA.
- [46] PRASAD, J. V. R., YAVRUCUK, I., and UNNIKRISHNAN, S., “Adaptive limit prediction and avoidance for rotorcraft,” in *Proceedings of 28th European Rotorcraft Forum*, (Bristol, UK), September 2002.
- [47] PRITCHETT, A. R., “Reviewing the role of cockpit alerting systems,” *Human Factors and Aerospace Safety*, pp. 47–79, 2001.

- [48] REYNOLD, C., “Not bumping into things.” Notes on obstacle avoidance for the course on Physically Based Modeling at SIGGRAPH 88 available online @ <http://www.red3d.com/cwr/nobump/nobump.html>, date accessed May 16th 2006.
- [49] SAHANI, N., *Envelope Protection Systems for Piloted and Unmanned Rotorcraft*. PhD thesis, The Pennsylvania State University, Department of Aerospace Engineering, December 2005.
- [50] SAHANI, N., HORN, J., JERAM, G., and PRASAD, J. V. R., “Hub moment limit protection using neural network prediction,” *Annual Forum Proceedings - American Helicopter Society*, vol. 1, pp. 775–786, May 2004. Baltimore, MD.
- [51] SAHANI, N. and HORN, J. F., “Neural network based algorithms for comprehensive collective axis limit avoidance for rotorcraft,” *Journal of Aerospace Computing, Information and Communication*, vol. 1, pp. 432–451, November 2004.
- [52] SAHANI, N. and HORN, J. F., “Command limiting for full-envelope guidance and control of rotorcraft,” in *AIAA Guidance, Navigation, and Control Conference and Exhibit*, (San Francisco, California), August 2005.
- [53] SAHANI, N. and HORN, J. F., “Adaptive model inversion control of a helicopter with structural load limiting,” *Journal of Guidance Control and Dynamics*, vol. 29, pp. 411–420, March/April 2006.
- [54] SAHASRABUDHE, V., SPAULDING, R., FAYNBERG, A., HORN, J., and SAHANI, N., “Simulation investigation of a comprehensive collective-axis tactile cueing system,” in *Proceedings of American Helicopter Society 58th Annual Forum*, vol. 1, (Montreal, Canada), pp. 559–568, June 2002.
- [55] SATTIGERI, R. and CALISE, A., “Neural network augmented kalman filtering in the presence of unknown system inputs,” in *AIAA Guidance, Navigation, and Control Conference and Exhibit*, (Keystone, CO), August 2006.
- [56] SATTIGERI, R., CALISE, A., and EVERS, J., “An adaptive approach to vision-based formation control,” in *AIAA Guidance, Navigation, and Control Conference and Exhibit*, (Austin, TX), August 2003.
- [57] SATTIGERI, R., CALISE, A., and EVERS, J., “An adaptive vision-based approach to decentralized formation control,” in *AIAA Guidance, Navigation, and Control Conference and Exhibit*, (Providence, RI), August 2004.
- [58] SATTIGERI, R., CALISE, A., and KIM, B., “6-dof nonlinear simulation of vision-based formation flight,” in *AIAA Guidance, Navigation, and Control Conference and Exhibit*, (San Fransico, CA), August 2005.
- [59] SCHRAGE, D. and VACHTSEVANOS, G., “Software enabled control(sec) for intelligent uavs,” in *Ist UAV Conference*, (Portsmouth, Virginia), May 2002.

- [60] UNNIKRISHNAN, S., JERAM, G., and PRASAD, J. V. R., “Tactile limit avoidance cueing using adaptive dynamic trim,” in *Proceedings of American Helicopter Society 60th Annual Forum*, vol. 1, (Baltimore, MD), pp. 763–774, June 2004.
- [61] WHALLEY, M. and ACHACHE, M., “Joint u.s/france investigation of helicopter flight envelope limit cueing,” in *Proceedings of American Helicopter Society 52nd Annual Forum*, vol. 2, (Washington, D.C), pp. 1589–1617, June 1996.
- [62] WHALLEY, M., BUCKANIN, R., KELLER, J. F., and ROOS, J., “Helicopter active control technology demonstrator program,” in *Proceedings of American Helicopter Society 57th Annual Forum*, vol. 1, (Washington, D.C), pp. 795–804, May 2001.
- [63] WHALLEY, M., HINDSON, W., and THIERS, G., “A comparison of active side-stick and conventional inceptors for helicopter flight envelope tactile cueing,” in *Proceedings of the American Helicopter Society 56th Annual Forum, Virginia Beach*, vol. 1, (VA), pp. 181–204, May 2000.
- [64] WHALLEY, M. S., “A piloted simulation investigation of helicopter limit cueing,” USAATCOM Technical report 94-A-020, Aeroflightdynamics Directorate, U.S.Army Aviation and Troop Command, Ames Research Center, Moffet Field, California, October 1994.
- [65] YANG, B.-J., *Adaptive Output Feedback Control of Flexible Systems*. PhD thesis, Georgia Institute of Technology, School of Aerospace Engineering, April 2004.
- [66] YAVRUCUK, I. and PRASAD, J. V. R., “Adaptive limit margin prediction and control cueing for carefree maneuvering of vtol aircraft,” in *Proceedings of the AHS Flight Control and Crew Design Technical Specialists Meeting*, (Philadelphia, PA), Oct. 2002.
- [67] YAVRUCUK, I., PRASAD, J. V. R., and CALISE, A. J., “Adaptive limit detection and avoidance for carefree maneuvering,” in *Proceedings of AIAA Atmospheric Flight Mechanics Conference*, (Montreal), August 2001.
- [68] YAVRUCUK, I., PRASAD, J. V. R., and CALISE, A. J., “Carefree maneuvering using adaptive neural networks,” in *Proceedings of the AIAA Atmospheric Flight Mechanics Conference*, (Monterey, CA), Aug. 2002.
- [69] YAVRUCUK, I., PRASAD, J. V. R., CALISE, A. J., and UNNIKRISHNAN, S., “Adaptive limit and control margin prediction and limit avoidance,” in *Proceedings of the American Helicopter Society 58th Annual Forum*, vol. 1, (Montreal, Canada), pp. 856–866, June 2002.
- [70] YAVRUCUK, I., UNNIKRISHNAN, S., and PRASAD, J. V. R., “Envelope protection in autonomous unmanned aerial vehicles,” in *Proceedings of the American Helicopter Society 59th Annual Forum*, vol. 2, (Phoenix AZ), pp. 2000–2010, May 2003.

- [71] YAVRUCUK, I., *Adaptive Limit Margin Detection and Limit Avoidance*. PhD thesis, Georgia Institute of Technology, School of Aerospace Engineering, July 2003.

VITA

Suraj Unnikrishnan was born on May 30th 1979, in Mumbai (Bombay), Maharashtra, India. In 1996 he secured admission into India's premier technology institute, IIT after clearing the all India Joint Entrance Examination (IIT-JEE). He was awarded a Bachelor of Technology in Aerospace Engineering from Indian Institute of Technology, Bombay, in June of 2000. In August 2000 he joined the graduate studies program in Aerospace Engineering at Georgia Tech. He was awarded a Master of Science (M.S) in Aerospace engineering in December 2002 and a Doctor of Philosophy (Ph.D) in Aerospace Engineering in August 2006. His research interests include adaptive control and estimation, optimal control and flight mechanics.