CRANFIELD UNIVERSITY

Matthew Osborne

Safe Online Learning for Nonlinear Dynamical Systems Using Control Contraction Metrics

SCHOOL OF AEROSPACE, TRANSPORT AND MANUFACTURING
Transport Systems

PhD
Academic Year: 2019–2023

Supervisors: Prof. Hyo-Sang Shin, Dr Dmitry Ignatyev, Prof. Antonios Tsourdos

March 2023

CRANFIELD UNIVERSITY

SCHOOL OF AEROSPACE, TRANSPORT AND MANUFACTURING
Transport Systems

PhD

Academic Year: 2019–2023

MATTHEW OSBORNE

Safe Online Learning for Nonlinear Dynamical Systems Using Control Contraction Metrics

Supervisors: Prof. Hyo-Sang Shin, Dr Dmitry Ignatyev, Prof. Antonios Tsourdos

March 2023

This thesis is submitted in partial fulfilment of the requirements for the degree of PhD.

© Cranfield University 2023. All rights reserved. No part of this publication may be reproduced without the written permission of the copyright owner.

Abstract

This thesis aims to develop an online learning framework for a military fixed-wing aircraft that can adapt a control policy to unforeseen changes in the airframe's flight dynamics. This is an active area of research and a significant challenge for high dimensional nonlinear systems due to the inherent safety risks and computational challenges of solving exponential time algorithms.

The research achieves this aim by providing an extensive survey of safe online learning approaches for nonlinear systems by assessing each technique with the aid of key performance metrics. Two critical performance metrics are the reliability and time complexity of the approach taken. To support the survey a benchmarking study of salient techniques provides further evidence to support the findings of the literature survey to identify promising avenues of research.

A generic safe learning process is defined and a convex optimisation learning pipeline is developed to handle nonlinear system identification and online controller synthesis via control contraction metrics. The developed pipeline is applied to a longitudinal simulation of an F-16 aircraft using high-fidelity wind tunnel data. The gap in knowledge around the application of control contraction metrics to aircraft designed to meet flying qualities requirements based on linear time invariant theory is bridged.

A novel cascaded two loop algorithm is developed to explicitly place the eigenvalues of the inner and outer loop of a differential feedback controller. Further a parameterisation of a robust controller is shown to better optimise the performance relative to flying qualities specifications. Conditions for a hybrid linear sum of controllers is shown to provide a stability guarantee for a mixed controller that enables a performance trade-off

of each approach. The performance of the developed controllers is demonstrated on six damaged aircraft profiles to assess the robustness and transient characteristics for each method based on a forty second flight trajectory.

The variation in nonlinear damage profiles illustrates the limitations of a linear approximating function for three nonlinear deviations. We show that the robust quadratic regulator controller generates a smoother transient response compared to the exponential contraction controllers. The two-loop contraction metric controller improves the rise-time performance compared to a single loop but is less robust to damage variations. The outcome of the research is a greater understanding of the application of contraction-based controllers and the effect of tuning parameters for a robust controller with potential for a reinforcement learning algorithm. Further a method to hybridise control policies is proposed and a loop shaping method using contraction based linear matrix inequalities developed with potential application to cascaded systems.

Keywords

Safe Learning; Model Based Reinforcement Learning; Adaptive Control; Nonlinear Systems; Control Contraction Metrics; Fixed-Wing Control; Convex Optimisation.

Acknowledgements

I would like to thank the Engineering and Physical Sciences Council (EPSRC) and BAE Systems for funding this PhD and making the research possible. I am also grateful to my supervisors Professor Antonios Tsourdos for his kind support and Professor Hyo-Sang Shin for giving me the technical direction and opportunity to explore this area of research as well as my thanks to Dr Dmitry Ignatyev for his valuable contribution and support in the technical details. Despite my set backs and frustrations they have supported and been on hand to keep me on track to finish my PhD for which I am truly grateful.

Thanks also to my industrial supervisor Professor Nick Colissimo for his enthusiam and motivational support throughout the PhD despite delays and changes of direction. Our meetings and discussions with the team at BAE have helped focus my attention and objectives for the research and provided a reality check for some of the proposals that would otherwise have set me off on the wrong track.

Much thanks go to David Torres Ocaña for his support in helping me to develop the aircraft simulator based on his previous MSc work and his kind words of advice.

A special thanks to all the researchers that contributed to my learning during my PhD including in the initial stages discussions with Dr Sumeet Singh from Stanford University, Professor Johan Löfberg from Linköping University for his expert knowledge in numerical techniques, Professor Ian Manchester from Sydney University for his expertise in the field of contraction metrics and guidance from Professor Steve Brunton from Washington University for his excellent tutorials and guidance in control systems learning techniques.

Thanks also to my colleagues in the Aerospace department, George and Saki for their kind words of support and motivation to keep on track and for discussions around the research technical details. And my fellow PhD colleagues and researchers in SATM for their support and chats over a coffee to bounce ideas off.

And finally and most importantly to my family much appreciation for their patience and support in keeping me level headed over the years and to my close friend Kami for her words of encouragement and patience during times of stress she convinced me to not give up by being the voice of reason for which I am grateful.

Contents

Abstract iii

Acknowledgements v

Contents vi

List of Figures ix

List of Tables xii

List of Abbreviations xiv

1 Introduction 1

1.1 Background and Context 1
1.2 Motivation 3
1.3 Aims and Objectives 4
1.4 Organisation of the Thesis 5
1.5 Contributions 6
1.6 List of Publications 8
References 9

2 Safe Online Learning 13

2.1 Introduction 13
2.2 The Generic Control Learning Framework 15
2.3 Performance Metrics 19
2.4 Approaches to Safe Online Learning of Nonlinear Dynamic Systems . . . 27
2.5 Convex Optimisation 33
2.6 LQR-Trees and Funnels 36
2.7 Nonconvex Techniques 38
2.8 Neural Network Approaches 39
2.9 Lyapunov Based Sampling Techniques 40
2.10 Nonlinear Model Predictive Control Techniques 40

2.11 Contraction Theory Based Techniques 43
2.12 Function Approximation Techniques 54
2.13 Benchmarking Evaluation of Approaches 66
2.14 Comparison of Techniques 76
2.15 Gaps and Challenges 81
2.16 Future Work 82
2.17 Conclusion 84
References 85

3 Safe Control Learning Pipeline via Static CCM 99
3.1 Introduction 99
3.2 Overview 103
3.3 The Learning Problem 107
3.4 The Learning Pipeline 113
3.5 F-16 Longitudinal Modelling and Simulation 119
3.6 LQR technique 126
3.7 The CCM Technique 127
3.8 Results and Discussion 141
3.9 Conclusion 149

Appendices 153
3.A Finsler's Lemma 153
References 154

4 Cascaded 2-Loop CCM and Hybrid RCCM 161
4.1 Introduction 161
4.2 Definition of the Aircraft Longitudinal Attitude Hold Controller 166
4.3 Review of Control Contraction Metrics (CCM) 166
4.4 Solution Methodology for CCM Loop Shaping and Hybrid Approaches . 170
4.5 System Analysis Using the Root Locus Technique 187
4.6 Step Response Analysis of Controller Performance 190
4.7 Results 196
4.8 Conclusions and Future Work 198
Acknowledgements 199

Appendices 200
4.A Aerodynamic Coefficient Deviations from Baseline Dynamics 200
4.B Linear Approximations for The Short Period Dynamics 202

4.C Comparison of Feedback Gain Matrices for Each Controller Design for Each Aircraft Condition . . . 203
4.D Step Responses for Each Controller Design for the 6 Damaged Aircraft Linear Approximations . . . 204
References . . . 205

5 Longitudinal Controller Performance 208
5.1 Introduction . . . 208
5.2 Testing Methodology Damage Profiles and Simulation Technique . . . 212
5.3 CCM and RCCM Linear Matrix Inequality Optimisation and Algorithmic Controller Designs . . . 215
5.4 Results . . . 220
5.5 Discussion . . . 224
5.6 Conclusions and Future Work . . . 233
Acknowledgements . . . 234
References . . . 235

6 Discussion on Learning via CCMs 237
6.1 Uncertainty Handling for Polytopic Uncertainty . . . 237
6.2 Nonlinear control synthesis . . . 238
6.3 Full Flight Envelope Scheduling . . . 240
6.4 Controller Augmentation . . . 240
References . . . 244

7 Conclusions and Future Works 245
7.1 Conclusion on original hypothesis . . . 247
7.2 Future Works . . . 248
Complete Bibliography . . . 249

List of Figures

1.1 Thesis Overview 6

2.1 NASA's Insight Lander, [90] 13
2.2 Lockheed Martin F-35 Lightning II, [6] 13
2.3 Lillium VTOL Air Taxi, [40] 13
2.4 A Phase Plane Diagram of a System's RoA Approximated by Three Function Classes (SoS -Yellow, LQR-Blue,NN-Orange) vs the True RoA (Green) from Richards et al. [96] 16
2.5 Generic Safe Online Learning Control System Framework 17
2.6 Safe Set Iterations and Zero Level Set Barrier Safety Function 18
2.7 Safety Certificate Computation Times vs Number of State Dimensions 30
2.8 Learning Based Control Lyapunov Function Approach 33
2.9 A convex optimisation function compared to a nonconvex function . . 34
2.10 An Example of the Positivestellensatz use of a positive polynomial . . 35
2.11 LQR-Trees Approach 37
2.12 Illustration of Bilinear Optimisation of SoS Lyapunov Equation 38
2.13 Learning Based - NMPC Approach 43
2.14 Contraction theory diagram, Adapted from [70]. 44
2.15 Learning Based - RCCM 47
2.16 HJI-Reachability with Gaussian Process Learning Based Safety Function 50
2.17 Uncertainty Aware Bayesian Optimisation Safe Learning Framework . 52
2.18 Bayesian optimisation over three iterations 53
2.19 Sparse Identification of Nonlinear Dynamics with Control SINDYc . . 58
2.20 A 3D input, 2D output, neural network with one hidden layer 60
2.21 Illustrative representation of a 2D neural network 60
2.22 Illustrative example of a 1D Kernel with 4 support vectors 62
2.23 Example of a Gaussian process function 64
2.24 Examples of Sparse Gaussian Process Regression 68
2.25 Exampes of Neural Network with 2 Hidden Nodes 69
2.26 Safety Certificate Learning Times Comparison 73
2.27 Safety Cerificate Learning Times - LQR,Lyapunov,CCM 73
2.28 Controller Synthesis Times 74
2.29 Function Approximation Learning Times 75
2.30 Function Approximation Mean Squared Errors 75

3.1 CCM Learning Framework 106
3.2 Benchmark 2-Loop Proportional Feedback Controller 106
3.3 F-16 Aircraft 120

3.4 The 3-DOF F-16 Reference Axes . . . 121
3.5 F-16 Longitudinal Dynamics Learning Simulator . . . 122
3.6 Aerodynamic force coefficient wind tunnel data . . . 123
3.7 Aerodynamic force coefficient data modified to simulate damage . . . 125
3.8 Contraction theory diagram . . . 127
3.9 Example of Worst Case Robustness Tube Bounds . . . 133
3.10 Flight Trajectory Data Compared to a Learned Model . . . 146
3.11 Baseline Optimised Gains Compared to Design Specification Limits . . . 146
3.12 Optimised Gains for the Damaged Aircraft Compared to Design . . . 146
3.13 Grid Search Disturbance Rejection for Baseline vs Damaged Aircraft . 150
3.14 Disturbance Rejection of the Nonlinear Baseline vs Damaged Aircraft . 150
3.15 Disturbance Rejection of a Linear Model for Baseline vs Damaged Aircraft . . . 151

4.2.1 Two-Loop 3rd Order Full State Proportional Feedback Controller . . . 167
4.4.1 MIL-STD Time Domain Tracking Performance Parameters, [13] . . . 176
4.4.2 Hybrid Controller with a = 1 . . . 182
4.4.3 Hybrid Controller with a = 0.8 . . . 183
4.4.4 Hybrid Controller with a = 0.5 . . . 183
4.4.5 Hybrid Controller with a = 0.2 . . . 184
4.4.6 Negative Feedback Poles for a Linear Sum Controller vs Controllers K1 and K2 . . . 187
4.5.1 Diagram for Root Locus Analysis of Transfer Functions . . . 189
4.5.2 Root Locus Analysis of Theta Feedback Gain . . . 192
4.5.3 Bode Plots for Varying Inner Loop feedback Gains in Ka and Kq . . . 192
4.5.4 Magnitude of Theta Feedback Gain to Meet Gain and Phase Margin Requirements Using a Root Locus Technique. (Error bars indicate that exact PM limits were not used but were within a few degrees of error to sample the Kt values) . . . 193
4.5.5 Step Response Effect of Increasing Outer Loop Pole Magnitude . . . 194
4.5.6 Inner Loop Damping Analysis p3 = -3 . . . 194
4.5.7 Inner Loop Damping Analysis p3 = -6 . . . 195
4.6.1 Root Locus Controller Subject to Damaged Aircraft Simulation for a Step Size of 5 degrees . . . 196
4.7.1 Controller Step Responses for the Baseline Case . . . 198
4.A.1 Pitching Moment and Axial Aerodynamic Force Coefficients for Varying Angles of Attack (AoA) and Varying Elevator Deflection Angles. Baseline Dynamics and Damage Profiles Baseline to D2 . . . 200
4.A.2 Pitching Moment, and Axial Aerodynamic Force Coefficients for Varying Angles of Attack (AoA) and Varying Elevator Deflection Angles. Baseline Dynamics and Damage Profiles D3 to D6 . . . 201

5.2.1 Simulator Block Diagram . . . 215
5.2.2 Two-Loop Attitude Hold Feedback Controller in Simulator Setup with Trim Correction . . . 216
5.4.1 Baseline Controllers with CCM contraction rate of 2.1 saturating actuator 224
5.4.2 Baseline Controllers with CCM contraction rate of 1.5 . . . 225

5.4.3 Baseline CCM Double and Single increasing lambda 225
5.4.4 Root Locus Baseline Controller for All Damaged Profiles 225
5.4.5 Pole Placement Controller for All Damaged Profiles 226
5.4.6 CCM Single Loop with $\lambda = 1.5$ Controller for All Damaged Profiles . 226
5.4.7 CCM Double Loop with $\lambda = 1.5$ Controller for All Damaged Profiles . 226
5.4.8 CCM Single Loop Controller with $\lambda = 1.2$ for All Damaged Profiles . 227
5.4.9 CCM Double Loop Controller with $\lambda = 1.2$ for All Damaged Profiles . 227
5.4.10 RCCM [1 0016 1] Controller for All Damaged Profiles 227
5.4.11 RCCM [1 0016 16] Controller for All Damaged Profiles 228
5.4.12 Hybrid Linear Sum for All Damaged Profiles 228
5.4.13 Baseline Hybrid CCM Double Loop and RCCM with mixing parameter 228
5.4.14 Baseline Hybrid Linear Sum CCM Double Loop and RCCM with mixing parameter 229
5.4.15 Damaged Case 6 Hybrid CCM Double Loop and RCCM with mixing parameter 229
5.4.16 Damaged Case 6 Hybrid Linear Sum CCM Double Loop and RCCM with mixing parameter 229

6.3.1 Example of an F-16 flight envelope and controller scheduling points. Red points show updated scheduling parameters learned from a flight trajectory. 241
6.3.2 Longitudinal Controller CCM Scheduling Example 242
6.4.1 Theta integral error controller augmentation 242

List of Tables

2.1 Metric Weighting 20
2.2 Time Complexity for common components 21
2.3 Response Time Ranking 22
2.4 Safety Certificate Learning Time Ranking 23
2.5 Conservatism Table 24
2.6 Reliability Ranking 26
2.7 Explainability Table 27
2.8 Subset of Safe Learning Techniques 28
2.9 Overview of Salient Techniques in Safe Learning of Control Systems with Key Attributes 31
2.10 Recent Studies Using Convex Optimisation Techniques 36
2.11 Recent Studies Using NMPC Techniques 43
2.12 Recent Studies Using Reachability Techniques 51
2.13 Recent Studies Using Uncertainty Aware Based Techniques 55
2.14 Function Approximation Techniques 57
2.15 NMPC Synthesis Times and Errors 71
2.16 Chebyshev Pseudospectral Newton Descent Hyperparameters 71
2.17 Overview of Advantages and Disadvantages 77
2.18 Summary of Learning and Response Times for Salient Techniques 78
2.19 Summary of Techniques 78

3.1 Stages of the Learning Pipeline 104
3.2 Index for Library Regressors for the Polynomial Model functions 115
3.3 Aircraft Properties 122
3.4 Simulation Trim Settings 124
3.5 Control Limits 124
3.6 CAT A Level 1 Flying Qualitites for Class IV Aircraft 125
3.7 Examples of Metric Lower Bound Certification and Feedback Parameters 136
3.8 Learned Parameters for Library Regressors for the Polynomial Model Approximation 142
3.9 Percentage Errors of Learned Model Tracking Tests 143
3.10 Gains Optimised for the Baseline and Damaged Aircraft for Each Technique 145

4.4.1 Definitions - Flying Qualities Specifications [21] 174
4.4.2 Short-Period Damping Ratio and Natural Frequency Limits for Class IV Aircraft, Cat A flight phases [21] 175
4.4.3 MIL-STD Time Domain Tracking Performance Parameters, [13] 176

4.4.4 MIL-STD Time Domain Tracking Specifications at 15,000 ft and 500 ft/s Airspeed 176
4.5.1 Analysis of Level 1 Flying Qualities Pass/Fail Criteria for the Inner Loop Damping Ratio and Natural Frequency and Outer Loop Pole Placment . . 191
4.5.2 Root Locus Design Options for Controller Gains 191
4.6.1 Summary of the Aircraft Aerodynamic Coefficient Modifications between 0 and 20 degrees alpha 195
4.7.1 Baseline 198
4.C.1 Damage Case 1 203
4.C.2 Damage Case 2 203
4.C.3 Damage Case 3 203
4.C.4 Damage Case 4 203
4.C.5 Damage Case 5 203
4.C.6 Damage Case 6 203

5.2.1 Summary of the Aircraft Aerodynamic Coefficient Modifications between 0 and 20 degrees alpha 213
5.2.2 Aircraft Properties 214
5.2.3 Simulation Trim Settings 215
5.3.1 Parameters Used in Controller Designs 217

List of Abbreviations

ARE	Algebraic Ricatti Equation
BO	Bayesian Optimisation
CAP	Control Anticipation Parameter
CBF	Control Barrier Function
CCM	Control Contraction Metric
CFD	Computational Fluid Dynamics
CLF	Control Lyapunov Function
CPA	Continuous Piecewise Affine
DNN	Deep Neural Network
DSOS	Diagonally Dominant Sum-of-Squares
EA	Eigenvalue Assignment
FCS	Flight Control System
GP	Gaussian Process
HJI	Hamilton-Jacobi Isaacs
LASSO	Least Absolute Shrinkage Selection Operator
LCB	Lower Confidence Bound
LEF	Leading Edge Flaps
LMI	Linear Matrix Inequality
LQR	Linear Quadratic Regulator
LTI	Linear Time Invariant
MIMO	Multi Input Multi Output

MPC	Model Predictive Control
MRAC	Model Reference Adaptive Control
NCM	Neural Contraction Metric
NDI	Nonlinear Dynamic Inversion
NN	Neural Network
NMPC	Nonlinear Model Predictive Control
OCP	Optimal Control Problem
PID	Proportional Integral Derivative
PIO	Pilot Induced Oscillation
PDE	Partial Differential Equation
QP	Quadratic Programming
RCCM	Robust Control Contraction Metric
RL	Reinforcement Learning
RLS	Recursive Least Squares
RKHS	Reproducing Kernel Hilbert Space
RNN	Recurrent Neural Network
SAL	Safe Active Learning
SDP	Semi-Definite Programming
SDSOS	Sparse Diagonally Dominant Sum-of-Squares
SBO	Safe Bayesian Optimisation
SINDY	System Identification of Nonlinear Dynamic Systems
SISO	Single Input Single Output
SOS	Sum-of-Squares
SQP	Sequential Quadratic Programming
TVLQR	Time Varying Linear Quadratic Regulator
UA	Uncertainty Aware
UCB	Upper Confidence Bound

Nomenclature

α	Angle of attack
α	Parameter vector
$\bar{c}$	Aerodynamic Mean Chord
$\bar{q}$	Dynamic Pressure
β	Parameter vector
δ_u	Differential control vector
δ_x	Differential state vector
$\dot{x}_e$	Error Dynamics
η	Control Input Command
γ	Parametric curve
$\hat{\mathcal{D}}$	Disturbance Set Estimate
$\hat{f}(x)$	Drift function approximation
$\hat{F}(x, u, d)$	Plant Function Approximation
κ	Kernel function
$\kappa_x(\cdot)$	Kernel function with fixed first coordinate
λ	Contraction Rate
λ_θ	Regularisation parameter
$\mathbb{K}$	Gram matrix
$\mathcal{D}$	Disturbance Set
$\mathcal{G}$	Unsafe Set
$\mathcal{K}$	Constraints Set

$\mathcal{S}$ Safe Set

$\mathcal{U}$ Control Input Set

$\mathcal{X}$ State Space Set

μ_* Posterior Mean

μ_{f_*} Predicted mean

ω Weights vector

ω_n Natural Frequency

$\phi(x)$ Feature space

$\Phi(x, u)$ Library or feature space

π Controller Policy

ρ A constant

$\rho(x, t)$ Closed loop gain function

Σ_* Posterior Covariance

σ_{f_*} Predicted std dev

θ Pitch Attitude

θ_c Pitch Attitude Command

θ_{err} Pitch Attitude Error

ξ Weight matrix

ζ Damping Ratio

A A Matrix

b Bias vector

$B(x)$ Control Matrix Function

$B_d(x)$ Disturbance Control Function

C Output Matrix

$c(x, u, d)$ Running cost

$d \in \mathbb{R}^{n_d}$ Disturbance vector

$E(s, \gamma)$ Curve Energy

$e(s, \gamma)$ Curve energy function

$F(x, u, d)$ Plant Dynamics Function

H Hamiltonian

I Identity Matrix

$J(x)$ Control Barrier Function

$J(x, u)$ Cost Function

$K(x)$ Gain Matrix

K_α Alpha Feedback Gain

K_θ Pitch Attitude Feedback Gain

K_q Pitch Rate Feedback Gain

$l(\gamma)$ Curve length

$L(x)$ Positive Polynomial

m No. of basis functions

$M(x, t)$ Riemannian Metric

P P matrix

Q Q matrix

q Pitch rate

R R matrix

r Range

s Geodesic parameter

s Sensitivity

T Time Horizon

$u \in \mathbb{R}^m$ Control Input Vector

v Pilot or Distubance Input

$V(x)$ Control Lyapunov Function

$V(x, \delta x)$ Differential CLF

$W(x, t)$ Inverse or Dual Metric

$x \in \mathbb{R}^n$ State Vector

x_0 Initial State

x_r	Reference State
y	Output vector
$f(x)$	Drift dynamics

Chapter 1

Introduction

1.1 Background and Context

Adaptive control has been of interest since the 1950s and modern approaches span many fields of study. In a recent survey by Annaswamy et al. [2] these fields include iterative learning, reinforcement learning, and linear parameter varying techniques. Adaptive control traditionally stems from model reference adaptive control (MRAC) and the MIT-rule in 1958 where a parameterised reference model is updated in real time based on tracking error dynamics, [2]. Later nonlinear techniques developed since the publication by Krstic [19] when nonlinear adaptive techniques such as backstepping, passivity [13], feedback linearisation [15] and tuning functions [28] became more widely researched. The L_1 adaptive control approach is a modern extension of MRAC. This method separates model updates into a robust solution and a faster solution, which is then filtered by a low-pass filter [14].

Recently reinforcement learning (RL) techniques have become of great interest for online learning for control systems leveraging training data in an offline learning and online adaptation framework via a safety filter, [12], [3],[5]. This is coupled by the rise in data-driven adaptive control methods and safe learning for aerospace applications,[16],[25]. Model-based RL presents a more efficient learning approach, especially when compared

to approximate dynamic programming and model-free learning, as discussed by Recht et al. [22] and Lewis et al. [20].

Despite these advancements, the application of an online learning pipeline in adaptive control still presents significant challenges. These mainly arise from the need for robust and reliable learning of expressive and tractable nonlinear functions that can provide efficient, real-time control synthesis solutions [8, 1, 4].

Moreover, the need to adapt the control policy to maintain performance and stability requirements in highly nonlinear flight envelopes or during mechanical failures adds time-dependent constraints. This requires a level of robustness for certification and guaranteed safety measures. Recent research is focused on nonlinear adaptive methods with safety guarantees [17, 10, 23, 9, 18, 21, 6, 24, 27].

Currently there are no commercially deployed online adaptive control systems for aircraft other than the use of Nonlinear Dynamic Inversion (NDI) in use with the F-35 as a control allocation scheme for its fly-by-wire flight control system [11]. While advanced, NDI is not fully autonomous online adaptive control because the system is not learning a complex plant model online. NDI requires the precise modeling of the aerodynamics of the aircraft and the ability to invert the control effectiveness matrix. Higher-order aerodynamic effects in reality are difficult to accurately model resulting in a mismatch of the true dynamics and derived controller.

There is also a gap between traditional engineering methods and practical nonlinear learning approaches which have not yet been accepted by certification agencies like the FAA [29].

The aim of this research is to develop a learning pipeline for adaptive control, with the goal of optimising the performance of fixed-wing aircraft within a safe and certifiable framework. The end goal therefore is a tractable method for safety certification that can improve aircraft performance subject to nonlinear plant dynamics.

1.2 Motivation

Modern combat aircraft and unmanned autonomous systems are typically augmented by a Flight Control System (FCS) computer. This is to reduce the pilot workload and maintain stable flight conditions or in the autonomous case adapt to varying operating conditions. The FCS is therefore a safety critical feature of the aircraft. The FCS can respond at a much higher rate to disturbances in flight than a pilot and can be programmed to match the changing flight dynamics and sensitivity of the control surfaces to different operating conditions within the flight envelope. The sensitivity of the control surfaces and the aircraft's flight dynamics are functions of the shape of the airframe and the operating Mach number, altitude, and weather conditions. Typically changes in flight dynamics are accounted for by employing a gain schedule at set operating points as functions of the Mach number and altitude. These operating points are used to design the FCS actuator gains via wind tunnel testing, Computational Fluid Dynamics (CFD), and extensive flight testing which have a very high production cost. The software developed for the FCS is effectively hard coded and goes through extensive certification procedures to ensure robust and safe control of the aircraft in all conditions. In service however changes occuring to the payload or structure of the aircraft due to damage need to be accounted for in the control policy by re-optimising the control gains to maintain safe flying and handling qualities. Due to the time and cost associated with re-optimising the FCS systems control policy this is practically not possible until a repair or design revison is planned in advance. There is therefore much interest in the use of real-time adaptation and learning in flight to maintain safe aircraft control policies without the need for offline testing.

Recent advancements in machine learning and control theory are moving towards intelligent learning of control via adaptive AI techniques. Many researchers have adopted model approximations combined with a Gaussian Process learning element that aims to learn the unknown system dynamics from sample data. The most significant challenge however is that of computation to provide safety guarantees for greater than 10 dimensional systems, which is currently not possible for highly nonlinear and coupled dynam-

ics. Adaptive control techniques like model predictive control are also computationally constrained for real time implementation whereas iterative methods like backstepping and sliding mode control require accurate models derived offline to be safely used for flight control systems. Recent studies adopting control contraction metrics and a linear differential analysis of dynamics offer a broader class of systems that can be characterised in reasonable online learning time frames.

The gap in the current research is that of proving stability for high dimensional systems online for more than ten dimensions for a nonlinear system where the dynamics need to be learned in flight. Current adaptive techniques rely on an accurate model to track the residual errors in the dynamics or rely on linear approximations. Nonlinear learning for high dimensional systems typically employ the use of neural networks [26], as in the case of Reinforcement Learning [12] but require offline training and are not suitable for safety critical applications. There is a need to learn from data in a safe incremental manner that is both explainable and reliable for it to be suitable for industry and application to flight control systems [7].

1.3 Aims and Objectives

1.3.1 Aims

This thesis aims to develop an online learning architecture for a flight controller for a military aircraft. This has not been done before in the context of safe learning in the absence of a reference model for an unstable nonlinear system.

1.3.2 Hypothesis

The main hypothesis as a result of the benchmarking analysis is that Control Contraction Metrics offer a fast and expressive framework for stability certification for adaptive flight controllers for reasonable online learning timeframes.

1.3.3 Objectives

The project is broken down into the the following objectives to:

- **O1** Review the state of the art in safe learning of nonlinear systems.
- **O2** Benchmark safe learning techniques for their suitability for high dimensional and safety critical applications.
- **O3** Develop key metrics to assess the different techniques found within the literature.
- **O4** Develop a realistic testbed simulation for a nonlinear F-16 aircraft.
- **O5** Implement a learning algorithm using the CCM technique and convex optimisation.
- **O6** Develop an algorithm that can learn the system dynamics in the context of design limitations from LTI theory.
- **O7** Show that the aircraft can achieve increased performance characteristics as a result of the learning method.
- **O8** Show how the developed architecture can handle faults and deviations from the baseline dynamics of the aircraft to re-learn the optimal control gains.

1.4 Organisation of the Thesis

This thesis is compiled in paper format. Chapters 2 to 5 are based on the papers submitted or in review. Each chapter therefore contains a self-contained introduction, aims and overview of the research and literature referenced by the individual paper.

The overall structure of the chapters and the objectives aligned to the relevant papers are shown in the Fig. 1.1

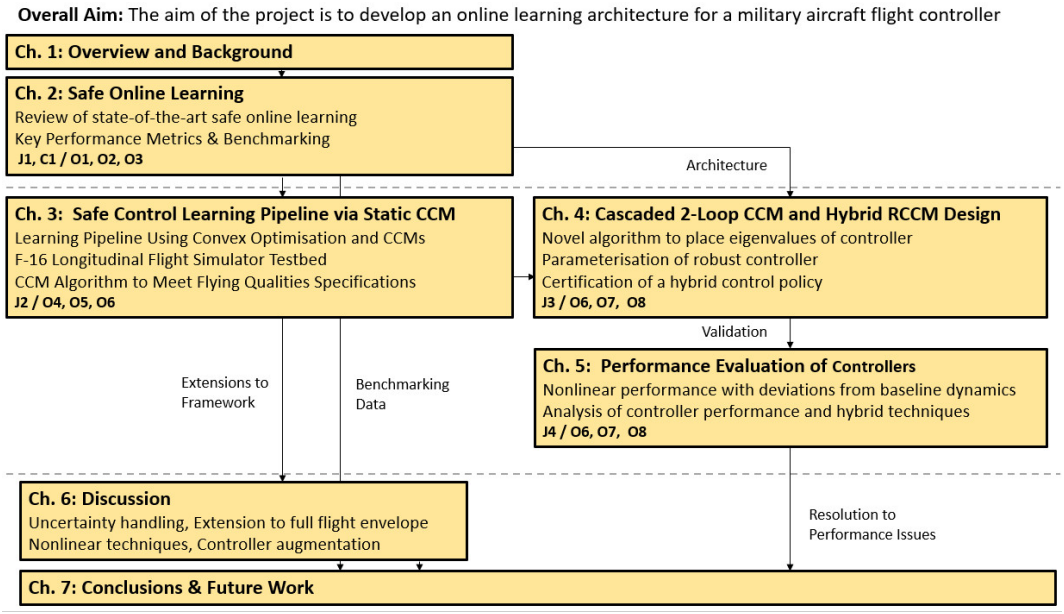


Figure 1.1 – Thesis Overview

1.5 Contributions

The following list of contributions made in this thesis has been broken down into major contributions and secondary contributions respectively.

1.5.1 Major Contributions

- An extensive literature review of the main approaches for safe learning of nonlinear dynamic systems is provided. The literature review compares the current state-of-the-art approaches to safe online learning for nonlinear dynamic systems. A description of a generic safe learning framework for feedback control systems is described and each technique compared to the generic framework. The learning pipeline is further broken down into algorithms that are used in each step of the process of learning to compare key performance metrics. Performance metrics as an heuristic ranking is given based on the review with proposals developed as a first look analysis so that the most promising techniques can be investgated for online learning for nonlinear control systems relevant to aerospace systems.

- A learning pipeline is developed adopting convex optimisation as an online batch learning algorithm to learn the system dynamics from trajectory data. The learned models are used to certify stability via the CCM technique and update the control policy within a proportional feedback architecture.

- We bridge the gap in the existing literature between nonlinear CCM theory and LTI theory to explore the practical implemention of aircraft flight controllers by integrating flying qualities into the design of a CCM controller.

- We develop a novel cascaded two-loop control contraction metric to explicitly optimise inner and outer loop contraction properties.

- A study exploring a hybrid LMI approach, which combines robust and exponential constraints with a mixing parameter, is conducted to assess its efficacy. Conditions for stability guarantees of a linear sum hybrid control policy are identified under certain conditions.

1.5.2 Secondary Contributions

- A benchmarking exercise is carried out for some salient techniques to compare the performance of algorithms on systems up to 10 dimensions. The scaleability in terms of processing speed is compared for safety certificate optimisations, function approximations, and control synthesis for a linear quadratic regulator, nonlinear model predictive controller, convex optimisation using a Lyapunov function and control contraction metrics. Function approximations compared include, sparse gaussian processes, neural networks, least squares regressions and convex optimisations. The sensitivity of processing times to polynomial degree and linearity was also carried out to validate the choice of polynomial functions for different dimensions.

- A nonlinear longitudinal F-16 simulator is developed and validated with previous models derived from NASA wind tunnel data. The simulator is constructed so that the kinematic equations are built as a block diagram so that the model can be interrogated with lookup tables for aerodynamic coefficients. The model includes six damaged aerodynamic coefficient profiles that can be automatically activated to simulate faults or deviations from the baseline.

- We analyse six novel controllers, a CCM, cascaded 2-loop CCM, an RCCM with scaling and prioritisation parameters, a Hybrid Linear Sum and Hybrid LMI controllers, within the nonlinear simulator with six damaged aircraft profiles to test the robustness and tracking performance of each controller.

- A comparison to a baseline root locus designed controller and pole placement approach offers a benchmark for evaluating the efficacy of the new control strategies.

- We explore the application of the developed learning pipeline to the full flight envelope and propose a method to adopt a reinforcement style algorithm.

1.6 List of Publications

In the list of publications, entries designated as 'Under Embargo' have not yet received publication clearance due to commercial considerations imposed by the sponsoring company. Consequently, a request is made for a six-month embargo on the thesis to facilitate the processing of intellectual property rights and patent applications.

C1 – M. Osborne, H.-S. Shin, A. Tsourdos, "A Review of Safe Online Learning Approaches for Nonlinear Control Systems" – In 2021 International Conference on Unmanned Aircraft Systems (ICUAS) 2021 Jun 15 (pp. 794-803) IEEE.

J1 - M. Osborne, H.-S. Shin, D. Ignatyev, A. Tsourdos, "A Review of the State-of-the-Art in Safe Online Learning for Nonlinear Control Systems" – JAE Online (In revision to submit)

J2 - M. Osborne, H.-S. Shin, D. Ignatyev, A. Tsourdos, "Longitudinal Control Learning via Static Contraction Based Metric" – 01/2024 (Under Embargo)

J3 - M. Osborne, H.-S. Shin, D. Ignatyev, A. Tsourdos, "Cascaded 2-Loop CCM and Hybrid RCCM Adaptive Controller Design for Longitudinal Flight Control" – 01/2024 (Under Embargo)

J4 - M. Osborne, H.-S. Shin, D. Ignatyev, A. Tsourdos, "Performance Evaluation of 2-Loop CCM and Hybrid RCCM Adaptive Controller" – 01/2024 (Under Embargo)

References

[1] A. A. Ahmadi, A. Chaudhry, V. Sindhwani, and S. Tu. Safely learning dynamical systems from short trajectories. pages 498–509, 2021.

[2] A. M. Annaswamy and A. L. Fradkov. A historical perspective of adaptive control and learning. *Annual Reviews in Control*, 52:18–41, 2021.

[3] F. Berkenkamp, M. Turchetta, A. P. Schoellig, and A. Krause. Safe model-based reinforcement learning with stability guarantees. *Advances in Neural Information Processing Systems*, 2017-Decem(Nips):909–919, 2017.

[4] N. M. Boffi, S. Tu, N. Matni, J. J. E. Slotine, and V. Sindhwani. Learning stability certificates from data. *arXiv*, pages 1–30, 2020.

[5] S. Chen, M. Fazlyab, M. Morari, G. J. Pappas, and V. M. Preciado. Learning lyapunov functions for piecewise affine systems with neural network controllers. *arXiv*, 2020.

[6] T. Cimen. Survey of state-dependent riccati equation in nonlinear optimal feedback

control synthesis. *Journal of Guidance, Control, and Dynamics*, 35(4):1025–1047, 2012.

[7] A. I. S. T. Committee et al. Roadmap for intelligent systems in aerospace. *Reston: AIAA*, 2016.

[8] M. Fisher, V. Mascardi, K. Y. Rozier, B.-H. Schlingloff, M. Winikoff, and N. Yorke-Smith. Towards a framework for certification of reliable autonomous systems. *Autonomous Agents and Multi-Agent Systems*, 35(1):1–65, 2021.

[9] S. Gros. Towards safe reinforcement learning using nmpc and policy gradients: Part ii - deterministic case. *arXiv*, pages 1–12, 2019.

[10] M. Hachicha, R. B. Halima, and A. H. Kacem. Formal verification approaches of self-adaptive systems: a survey. *Procedia Computer Science*, 159:1853–1862, 2019.

[11] J. J. Harris. F-35 flight control law design, development and verification. In *2018 Aviation Technology, Integration, and Operations Conference*, page 3516, 2018.

[12] S. Heyer, D. Kroezen, and E.-J. Van Kampen. Online adaptive incremental reinforcement learning flight control for a cs-25 class aircraft. In *AIAA Scitech 2020 Forum*, page 1844, 2020.

[13] D. Hill and P. Moylan. The stability of nonlinear dissipative systems. *IEEE transactions on automatic control*, 21(5):708–711, 1976.

[14] N. Hovakimyan, C. Cao, E. Kharisov, E. Xargay, and I. M. Gregory. L 1 adaptive control for safety-critical systems. *IEEE Control Systems Magazine*, 31(5):54–104, 2011.

[15] S. Jiffri, J. Mottershead, and J. Cooper. Adaptive feedback linearisation and control of a flexible aircraft wing. In *Topics in Modal Analysis, Volume 7: Proceedings of the 31st IMAC, A Conference on Structural Dynamics, 2013*, pages 683–699. Springer, 2014.

[16] E. Kaiser, J. N. Kutz, and S. L. Brunton. Sparse identification of nonlinear dynamics for model predictive control in the low-data limit. *Proceedings of the Royal Society A*, 474(2219):20180335, 2018.

[17] M. Kaminski. *A rapid robust fault detection algorithm for flight control reconfiguration*. PhD thesis, Technische Universität München, 2019.

[18] B. Kiumarsi, K. G. Vamvoudakis, H. Modares, and F. L. Lewis. Optimal and autonomous control using reinforcement learning: A survey. *IEEE Transactions on Neural Networks and Learning Systems*, 29(6):2042–2062, 2018.

[19] M. Krstic, P. V. Kokotovic, and I. Kanellakopoulos. *Nonlinear and adaptive control design*. John Wiley & Sons, Inc., 1995.

[20] F. L. Lewis and D. Liu. *Reinforcement learning and approximate dynamic programming for feedback control*. John Wiley & Sons, 2013.

[21] L. Nepomuceno, É. A. Moura, M. A. Morales, R. G. Silva, and L. C. Góes. An lqr-lmi longitudinal stability augmentation system for a subscale fighter aircraft with variable center of gravity position. In *AIAA AVIATION 2022 Forum*, page 4059, 2022.

[22] B. Recht. A tour of reinforcement learning: The view from continuous control. *Annual Review of Control, Robotics, and Autonomous Systems*, 2(1):253–279, 2018.

[23] C. Rouff, R. Buskens, L. Pullum, X. Cui, and M. Hinchey. The adaptiv approach to verification of adaptive systems. In *Proceedings of the fifth international c* conference on computer science and software engineering*, pages 118–122, 2012.

[24] R. Self, M. Harlan, and R. Kamalapurkar. Online inverse reinforcement learning for nonlinear systems. In *2019 IEEE conference on control technology and applications (CCTA)*, pages 296–301. IEEE, 2019.

[25] M. Selim, A. Alanwar, M. W. El-Kharashi, H. M. Abbas, and K. H. Johansson. Safe reinforcement learning using data-driven predictive control. In *2022 5th International Conference on Communications, Signal Processing, and their Applications (ICCSPA)*, pages 1–6. IEEE, 2022.

[26] L. Sonneveldt, E. R. Van Oort, Q. P. Chu, C. C. De Visser, J. A. Mulder, and J. H. Breeman. Lyapunov-based fault tolerant flight control designs for a modern fighter aircraft model. *AIAA Guidance, Navigation, and Control Conference and Exhibit*, (August):1–23, 2009.

[27] J. Umenberger, M. Ferizbegovic, T. B. Schön, and H. Hjalmarsson. Robust exploration in linear quadratic reinforcement learning. *Advances in Neural Information Processing Systems*, 32, 2019.

[28] N. Wahid and N. Hassan. Self-tuning fuzzy pid controller design for aircraft pitch control. In *2012 Third International Conference on Intelligent Systems Modelling and Simulation*, pages 19–24. IEEE, 2012.

[29] C. Wilkinson, J. Lynch, R. Bharadwaj, and K. Woodham. Verification of adaptive systems. *Federal Aviation Administration, DOT/FAA/TC-16/4*, 2016.

Chapter 2

Safe Online Learning

2.1 Introduction

Intelligent autonomous control systems are of great interest in a wide array of aerospace applications, from remote space exploration, to operation in highly nonlinear flow fields, to safety critical assurance cases as shown in Figs.2.1-2.3. With the rise in the use of machine learning techniques and autonomous systems, there have been great efforts to develop data driven self certification techniques for full autonomy [44, 43, 103, 23].

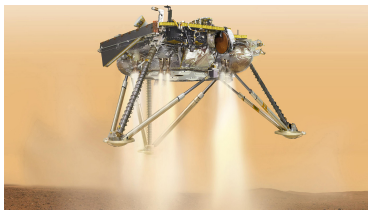


Figure 2.1 – NASA's Insight Lander, [90]

Figure 2.2 – Lockheed Martin F-35 Lightning II, [6]

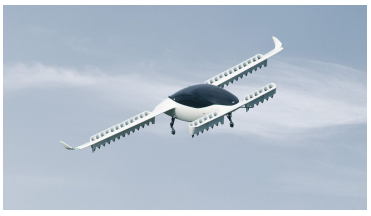


Figure 2.3 – Lillium VTOL Air Taxi, [40]

The F35 Lightening, Fig.2.2, is a good example of the development of state-of-the art control systems for safety critical applications. The aircraft developed by Lockheed Martin has seen the progression from traditional Proportional Integral Derivative (PID) Gain Scheduling control design to Nonlinear Dynamic Inversion (NDI) flight control systems that represent feedback linearisable dynamics covering 3 million data points [52]. The design of the system is an example of the complexity and expensive nature of nonlinear aerodynamic challenges in resolving highly complex flight envelopes at high resolution for discrete airframe configurations. It is anticipated that the next paradigm of

flight control systems might see the aircraft learning its own dynamics and stable control policies in flight [37]. Currently however, safe online learning of nonlinear dynamic systems is a significant challenge [103, 23, 87, 127, 119] and attempts to realise a practical online learning algorithm are incredibly challenging [17, 54, 46]. To learn by trial and error, there is the risk of entering unsafe states which is unacceptable for safety critical systems. Furthermore, traditional analytical techniques like Lyapunov theory used in the control systems field to guarantee stability become intractable for highly nonlinear systems which are prevalent in real world dynamics [77, 71]. State-of-the-art learning frameworks bridge the gap between traditional adaptive and optimal control techniques and reinforcement learning (RL) within the computer science field [64, 61, 47, 51, 93]. Traditional control theory is model-based whereas RL is classically model-free learning of discrete state-action spaces arising from dynamic programming and machine learning techniques. Reinforcement learning techniques [109] are described by a policy, synonymous with a controller, and a value function which is typically termed as a cost function and represents the sum of rewards over a time horizon. The interaction of the system with the environment and reward structure adapts the control policy over time to optimise performance. RL for control systems like the actor-critic network are currently reliant on offline simulation using high capacity computing resources [121]. RL based approaches are also dependant upon the skill of the designer in hand crafting a reward function [42] and there are also no guarantees that trained controllers operating in the real world will act safely in the presence of unseen system changes or environmental disturbances [115, 65, 94]. Continuous and high-dimensional spaces in control applications also lead to exponential growth in state and actions spaces [57] that make real time computations intractable. Safe learning for nonlinear control systems has predominantly relied on model-based control theoretic proofs for certification of stability. For dynamic systems certifying stability relies on proofs relating to operation of the system within its region of attraction (RoA). Nonlinear systems can be analytically or otherwise sampled for certification of the RoA. This is a region bounded around an equilibrium

point where all phase plane trajectories converge to the equilibrium point, ensuring safe operation and control of the system. The computation of the RoA is however a difficult and computationally expensive process; [89] states that they are typically NP-hard due to the typically nonconvex and high dimensionality of the systems' state space dynamics. Solutions to calculate the safe regions for continuous time systems include polynomial candidate functions and Sum-of-Squares (SoS) techniques, Continuous Piecewise Affine (CPA) candidate Lyapunov functions (LFs), sampling, state-space partitions and convex programming techniques [24]. Fig. 2.4 shows a recent study be Richards et al. [96] that estimates the RoA via a neural network, LQR and SoS technique that highlights the conservative nature of SoS when control saturation is a nonlinear constraint on learning. Safe regions can also be calculated explicitly via the use of partial differential system equations (PDEs) and differential game theory as in the case of the Hamilton-Jacobi-Isaacs (HJI) reachability technique where a grid of the state space is resolved into safe or unsafe reachable sets via worst case disturbances. Other library based techniques include LQR-trees where safe trajectories are computed based on time evolution of safe sets robust to distubances and CPA parameter varying techniques where optimised linear system equations are computed over the entire state space grid. One of the biggest challenges, however, for grid-based or tree-based algorithms such as reachability and funnels has been the *curse of dimensionality* for generating state space safety certificates of modelled behaviour and the tradeoff of conservatism in dynamics modelling vs tractable controller synthesis under safety constraints [39, 17, 23, 112].

2.2 The Generic Control Learning Framework

The generic safe learning control system framework, subject to environmental disturbances is shown in Fig. 2.5. The general control-affine equation is given by:

$$\dot{x} = F(x, u, d) = f(x) + B(x)\, u + B_d\, (d(x)) \tag{2.1}$$

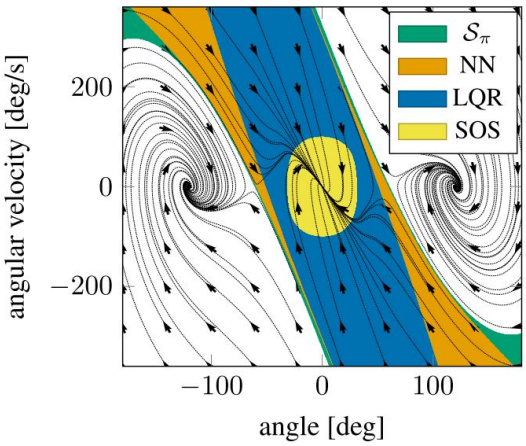


Figure 2.4 – A Phase Plane Diagram of a System's RoA Approximated by Three Function Classes (SoS -Yellow, LQR-Blue,NN-Orange) vs the True RoA (Green) from Richards et al. [96]

where the state vector is $x \in \mathcal{X} \subset \mathbb{R}^n$ and control input $u \in \mathcal{U} \subset \mathbb{R}^m$, with disturbance $d \in \mathcal{D} \subset \mathbb{R}^{n_d}$. The functions are mappings, $f : \mathcal{X} \to \mathbb{R}^n$, and $B : \mathcal{X} \to \mathbb{R}^{n \times m}$ and the model function approximation is represented as $\hat{F}(x, u, d) = \hat{f}(x) + \hat{B}(x)\, u + \hat{B}_d(d(x))$. The main goal of the learning algorithm is to update the feedback control policy, $\pi_i(x, t)$, where i is the i^{th} policy iteration, via optimisation of a cost function under dynamic constraints. Data is collected online at discrete time steps k as $\{x_k, x_k^*, u_k, \dot{x}_k\}$ where x^* is the set point or reference trajectory and model learning is performed on batches of trajectory data typically online. The system is initialised with an a priori control policy π_0 and the approximation of the system dynamics is improved over time as the system explores the state space and environmental disturbance bounds. Disturbance bounds are defined as the magnitude of the maximum L_2 norm of exogenous inputs, assumed to arise from aerodynamic effects such as wind gusts, turbulence or flow fields caused by foreign bodies.

The disturbance $d(x)$ is unknown, but assumed to be bounded at each state by a compact set $\hat{\mathcal{D}}(x) \subseteq \mathcal{D}$. The disturbances are learned as the system explores the environment. The system dynamics are subject to constraints defined by a constraint set $x \in \mathcal{K} \subset \mathbb{R}^n$. Safe limits are those states and actions which keep the system within controllable states $\forall x(t),\ \forall u(t),\ t \geq 0$. The safe states are stored as the safe set denoted $\mathcal{S}_i$ which is grown over time incrementally in i steps as the system learns more about the dynamics

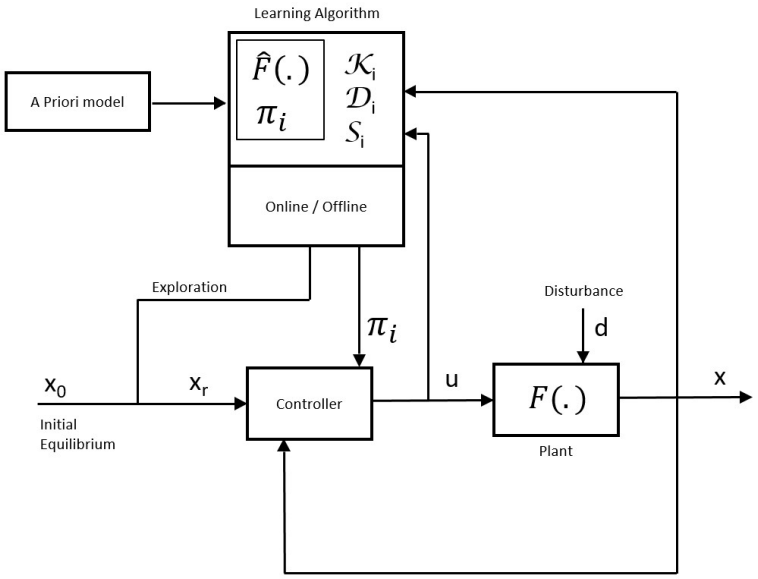


Figure 2.5 – Generic Safe Online Learning Control System Framework

and disturbance bounds. A key feature of safety critical learning is that under uncertain conditions or in states and disturbances outside of the safe limits, the system will either fall back to a previous more cautious safe set or a more cautious policy until confidence has again been made in the stability of the system. The typical method to determine safe and unsafe regions of the state-space is via Control Barrier Functions (CBFs), also known as safety functions, described in [10]. They provide a method for pruning the state space of unsafe states by defining an acceptable set that is invariant under the control actions available to the system. CBFs can be added to cost functions to avoid unsafe regions of the state space as certificates. The certificates define an unsafe set $\mathcal{G}_0$ and an initial safe set as $\mathcal{S}_0$ with a function $J(x) \geq 0$, $\forall x \in \mathcal{S}_0$ and $J(x) < 0$, $\forall x \in \mathcal{G}_0$, Fig. 2.6. Safe actions are then defined as preserving the inequality $\dot{J}(x, u) \geq -\gamma J(x)$, where $\gamma \geq 0$, by enforcing actions to remain inside the set at the boundary whilst allowing all actions inside the safe set. This is because outside of the zero level set of $J(x)$ the function changes sign and forces the gradient $\dot{J}(x)$ to also change sign causing the motion of the system to go from travelling downhill on the function to uphill on the function returning to the region of the safe set. Once inside the safe set both positive and negative gradients are allowed, scaled by the positive factor γ, enabling free exploration. Control barrier

functions are a generalisation of Control Lyapunov Functions (CLFs), where Lyapunov functions define level sets for stable trajectories and the function remains positive and centered at an equilibrium point at $x = 0$, $\dot{V}(x, u) \leq -\alpha V(x)$. Fig. 2.6 shows both the Lyapunov function and the barrier function definitions in the same plot with a second iteration of a safe set $\mathcal{S}_2$.

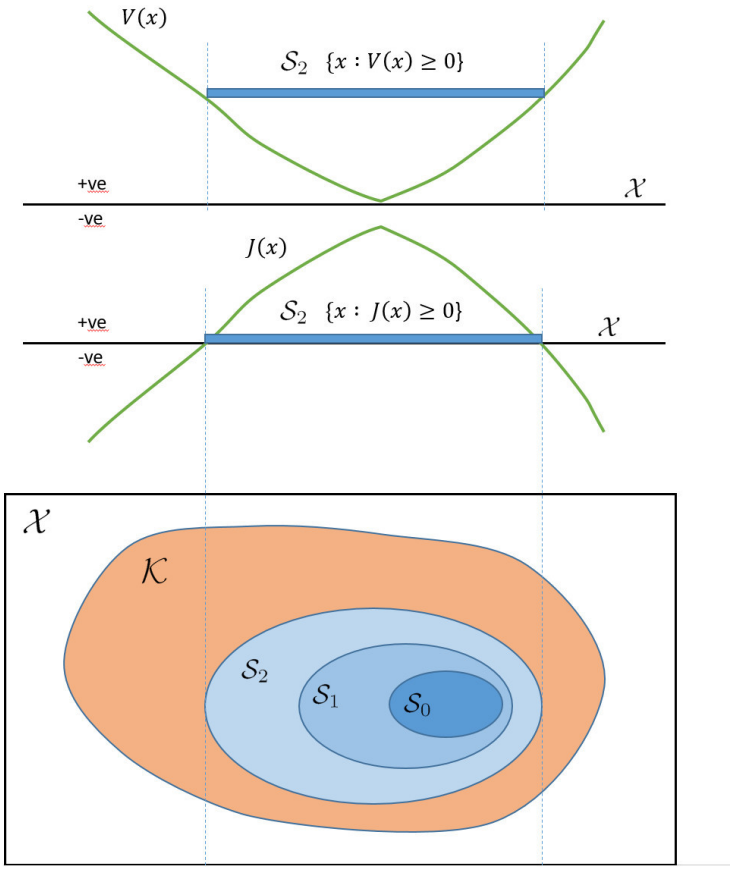


Figure 2.6 – Safe Set Iterations and Zero Level Set Barrier Safety Function

In Lyapunov's direct method for the function $V(x)$, stability is guaranteed if the derivative $\dot{V}(x) \leq 0$. The system is said to be asymptotically stable if a strict inequality is found. Typically in control systems however exponential stability is desired with additional convergence properties as shown by the following conditions [62, 110]:

$$V(x) > 0, \tag{2.2a}$$
$$\forall x \in \mathcal{B} \setminus \{0\}, \quad V(0) = 0$$
$$\dot{V}(x) = \frac{\partial V}{\partial x}\hat{F}(x,u) \leq -\alpha V(x), \tag{2.2b}$$
$$\forall x \in \mathcal{B} \setminus \{0\}$$
$$\dot{V}(0) = 0,$$
$$where \quad \alpha > 0$$

The stable region around the origin is described by a ball of states forming an open subset of $\mathbb{R}^n$ called set $\mathcal{B}$ where $\setminus$ means not included in the set. $\hat{F}(x,u)$ is the learned approximation of the system dynamics as a function of the state x and control inputs u and $\frac{\partial V}{\partial x}\hat{F}(x,u)$ in the Lie derivative of the Lyapunov function in the direction of the dynamics. α is an exponential scaling factor greater than zero. Global stability is guaranteed if $\hat{F}(x,u)$ is continuous and $V(x)$ is continuously differentiable, where $V(x) \to \infty$ whenever $\|x\| \to \infty$.

2.3 Performance Metrics

Learning systems for safety critical control will require challenging performance measures to be acceptable as safe for online implementation. It is our intention to aid comparisons of techniques to highlight the major implementation challenges and promote the development of solutions for real world applications. A non exhaustive set of key performance criteria are detailed in the next section. The metrics are initially a qualitative ranking with the intention of providing quantitative measures in the future. The metrics provide a first approximation rather than a detailed scientific analysis. Each metric is also weighted depending upon the perceived criticality of the requirements for safety critical applications in Table 2.1.

Table 2.1 Metric Weighting

Metric (s)	**Attributes** (%)	**Weighting** (%)
Scaleability	No. of Dimensions possible	15%
Response Time	Time to Synthesise control policy	20%
Learning Time	Time to Learn a safe policy	15%
Conservatism	Limitation of Safe Set	10%
Reliability	Solution Guarantees	25%
Explainability	Level of Analytical Abstraction	15%

It is important to note that direct comparisons of current research is not the intention as learning and response times will be dependant upon hardware and implementations of each approach and are typically not benchmarking exercises. We discuss each performance criteria considered in the following:

2.3.1 Scaleability

Scaleability relates to the time complexity of an algorithm in the computer science field and is a measure of how the number of operations an algorithm must compute, scales with the number of inputs that must be processed. We approximate complexity to the highest order where the focus is on the safety certification techniques. If an algorithm steps through a sequence of N operations by a fixed time per operation, it is termed linear time in complexity, $\mathcal{O}(N)$. If there are nested loops within the algorithm then the number of operations scale polynomially with the input $\mathcal{O}(D^k)$, where k is the number of loops. For matrix multiplications an $[n \times m]$ matrix times an $[m \times n]$ matrix is $\mathcal{O}(nmn)$, n rows of addition by (m elements $\times$ n) columns of multiplications. To approximate complexity for matrix operations in dimension D, this simplifies to $\mathcal{O}(D^3)$. In the case of Gaussian Process Regression the Matrix inverse must be found from an $(N + I)^2$ Matrix for each dimension of the state space, resulting in complexity $\mathcal{O}(D\,(N + I)^3)$ where

N is the number of reference points and I the number of inference points in the state space. Sparse Gaussian Processes reduce the complexity by only sampling the function at sparse Pseudo points P of complexity $\mathcal{O}(D\,(N+I)^3)$. Neural networks are essentially matrix multiplications with the forward inference operation including an activation function for each element which increases complexity to $\mathcal{O}(D^4)$. Neural network training complexity is further complicated by backpropogation with a gradient measure per iteration ε and so has complexity $\mathcal{O}(\varepsilon D^5)$ subject to the number of iterations required for convergence. For a more detailed analysis of neural network complexity refer to, [51]. For algorithms that use a gridded search space, discretised over each input dimension, the number of operations scales exponentially with complexity, $\mathcal{O}(kN^D)$, where k is a constant time operation and N is the number of discrete sampling points per dimension. Convex optimisation can be approximated as a polynomial time algorithm D^k, although the complexity is dependant upon the chosen algorithm, ellipsoid methods and interior point methods are commonly used, [116]. For a more recent in depth analysis of convex optimisation and complexity see [30]. We rank the algorithmic components typical in safe learning in Table 2.2.

Table 2.2 Time Complexity for common components

Algorithm	**Complexity**	**Score**
Convex Optimisation	$\mathcal{O}(D^k)$	••••
Sparse Gaussian Process Training	$\mathcal{O}(DP\,(N+I)^2)$	•••
Gaussian Process Training	$\mathcal{O}(D\,(N+I)^3)$	•••
Neural Network Training	$\mathcal{O}(\varepsilon(MD)^5)$	••
Grid Search	$\mathcal{O}((kN)^D)$	•

* N = number of reference points, I = number of inference points, P = number of pseudo points M = number of hidden nodes, ϵ = number of training iterations, k = a constant, L = Library dimension D = the dimensionality of the state variables.)

2.3.2 Response Time

Response time refers to the time taken to synthesise a control policy online, given tracking error data and a safety certificate. To process controller updates to maintain stability and control, response time is a critical feature for highly dynamic systems. For example, a military aircraft must not have a controller delay of more than 0.1s, [106], as when combined with a pilot reaction time of approximately 0.2s, a limit of 0.3s is reached after which Pilot-Induced Oscillation (PIO) becomes problematic, [80]. Given that for aircraft a response time of less than 0.1 seconds is required we define rankings for algorithm response times above and below this threshold as being in the millisecond range, 10 millisecond range and above the threshold as in the less than 0.5 second range and as a worst case greater than 0.5 second range. It is important to note that the reported response times where available are from different hardware systems and with differing dimensionality constraints so an order of magnitude is reported as a first approximation for each algorithms performance attributes. Table 2.3 lists some of the salient techniques for safe online learning with an approximate comparison of processing times where cited in the literature.

Table 2.3 Response Time Ranking

Time (s)	**Score**
< 0.001	•••••
< 0.01	••••
< 0.1	•••
< 0.5	••
> 0.5	•

2.3.3 Safety Certificate Learning Time

The online learning time required to generate a safety certificate is dependent upon the time complexity of an algorithm and number of dimensions of a system and also the hardware used to process the algorithm. In order to estimate the required learning times it is important to consider the situation in which learning is required and the level of urgency. For example a pre-determined test flight schedule may last for hours whereas recovery of the aircraft from loss of control authority due to a change in the system dynamics would require decisions to be made in seconds. We define therefore these windows of time for 5 scenarios for a military aircraft in table 2.4 and the corresponding ranking for a score for learning time based on these scenarios. The flight envelope test window refers to the situation where the dynamics over a section of the flight envelope are sampled during repeated manoeuvres which may last between 30 seconds and 5 minutes. A mission flight is proposed as a window of time in which a new environment, airframe loadout or airframe damage may occur during the limited time of an operation where the dynamics of the airframe are not critically affected but performance can be improved for the remainder of the flight.

Table 2.4 Safety Certificate Learning Time Ranking

Scenario	Time Frame	**Score**
Recovery Low Altitude	$0 \leq t \leq 5s$	•••••
Recovery High Altitude	$5s \leq t \leq 30s$	••••
Flight Envelope Test	$30s \leq t \leq 5m$	•••
Mission Flight	$5m \leq t \leq 30m$	••
Test Flight	$30m \leq t$	•

2.3.4 Conservatism

Conservatism refers to the extent to which the algorithm explores the true safe set, S_T as opposed to the approximated safe set, S_i, determined by a safety certificate function. Conservatism can be thought of as the volumetric ratio $\frac{Vol(S_0)}{Vol(S_T)}$, where an unconservative function will tend to one. Although analytical data for conservatism is not usually available some intuition can be gained from the study by Richards et al. [96]. In the study, it was shown that a NN function approximation can better represent the true safe set of a system when compared to LQR and SoS techniques. Although the SoS procedure is typically similar to the LQR approach as they are both quadratic functions, in [96] the SoS optimisation had additional control saturation constraints emposed, reducing the safe region further. For each function approximation the expressiveness is also determined by the number of basis, features or nodes in the function so comparing different functions would require an information budget of some description, which we do not propose here but note that NNs generally express more complex relationships per node and layer due to the combinations possible between connections. Whereas GPs and other linear techniques require the nonlinear features to be decided in advance to some extent. SoS techniques are clearly limited by nature of the geometry arising from ellipses. We therefore rank techniques approximately in Table 2.5.

Table 2.5 Conservatism Table

Conservatism	**Score**
Neural Networks	•••••
Gaussian Processes	••••
Monomial Kernel Methods	••••
Lyapunov LQR	•••
Sum of Squares	•••
Sparse SoS	••

2.3.5 Reliability

Reliability is a ranking of whether a solution is guaranteed to exist. To synthesise a control policy a chain of events must occur, a function regression, the generation of a safety certificate and the synthesis of the control policy. In this chain of events the reliability is dependent on the weakest link in the chain. To compare techniques therefore we rank reliability for each component and summarise the final rank for algorithms based on the weakest link, Table 2.6. Some techniques combine steps into one step, for example NMPC combines safety certification and control synthesis in one optimisation step. The most reliable analytical proofs are from model based linear approximations which are guaranteed to generate solutions from any training data although the solution may not be accurate the fact that a solution is available may be more critical for time dependant decision making. Convex optimisation is considered a technology [26], and the most reliable optimisation technique today. A solution is guaranteed to be found accurately or within a small distance to the global optimum provably. Although expert knowledge is sometimes required for particularly flat convergence regions where a small uncertainty in a threshold may produce significant offsets in the solutions. We rank linear search techniques like Bayesian Optimisation, Bilinear optimisation, and Game Theory third because although solutions can be found efficiently there are no formal guarantees for these techniques and solutions potentially diverge or stagnate for some functions. In sparse GP representations or overly conservative approximations, searching for a desired optimum may suffer from local maxima or overly conservative results, subject to the acquisition function, hyperparameters, and kernel function of choice. In general HJI solutions generated over a grid space require careful tailoring of the toolbox options [85], grid spacing and function approximations to determine the accuracy of solutions. HJI theory is predominantly aimed as a pedagogical research tool, [86] and is still the subject of research for tractable closed form solutions [8]. NMPC we rank 4th because although the online synthesis is highly dependant upon the choice of time horizon and trajectory complexity there are no formal guarantees for a stable solution. We then rank Nonlinear Programming and Deep Neural

Networks 5th, even though they excel at high dimensional function approximations the reliability can only be certified through extensive training and testing. Research is active in the area of certifying DNNs for safety critical applications, [125, 128] but currently for online learning no guarantees can be made for unseen data.

Table 2.6 Reliability Ranking

Technique	**Ranking**
LQR ARE	•••••
Convex Optimisation	••••
Bilinear Techniques	•••
GP Bayesian Optimisation	•••
HJI Differential Game Theory	•••
NMPC Multiple Shooting	••
Nonlinear Programming	•
Deep Neural Network SGD	•

Safety related systems are required to be verifiable, deterministic, and traceable. These attributes are a requirement for managing and accounting for system failures. Formal methods provide a way to guarantee performance, but formal approaches can suffer from over conservatism in maintaining tractable expressions. Where expressive function approximations are used, the complexity of the mapping of inputs to outputs can reduce explainability. An overview of the levels of AI explainability is discussed in, [14]. For this review, Table 2.7 has been used to classify numerical complexity and hence explainability as a result of being interpretable.

Table 2.7 Explainability Table

Algorithm Structure	**Explainability**	**Score**
Rule Based, Formal Logic	Logical Proofs	•••••
Linear Systems Tools	Analytical	••••
Nonlinear Model Based	Moderate Complexity	•••
Model-Based Hybrid Techniques	”Grey Box”	••
Deep Learning and Model Free	”Black Box”	•

2.4 Approaches to Safe Online Learning of Nonlinear Dynamic Systems

In this section we give an overview of the main approaches adopted in safe learning based on the generic framework and function approximation techniques discussed in the previous section. Noting that we are not focusing on techniques that arise from dynamic programming and deep learning but more on learning in the context of traditional control theory. With this in mind safe learning techniques can broadly be broken down into two categories:

1. Offline library-based techniques with online uncertainty handling.

2. Online trajectory optimisation techniques with safety filtering.

Table 2.8 provides a subset of recent studies that consider robust learning under disturbances. The techniques are typically hybrid models with nominal model-based and affine model-free learning components. Offline techniques to determine a library of safe states and actions include explicit Hamilton Jacobi Isaacs (HJI), [43, 33, 124], calcualtions over a discretised state space or precomputed robust trajectories in the form of funnel libraries, [74] or linearised CPA techniques that adopt linear quadratic regulator (LQR) parameter varying controllers, [19]. Online techniques adopt some form of MPC over a time horizon either directly via a nonlinear model and sequential quadratic programming (SQP),

[49, 97] or as in the case of the Control Contraction Metric (CCM) technique via optimisation of a stabilizing control trajectory using a Riemannian metric, [103, 77, 114]. Uncertainty aware (UA) techniques define algorithms that employ Bayesian Optimisation (BO) for exploration, [23, 43, 55].

Table 2.8 Subset of Safe Learning Techniques

	LMPC	**RCCM**	**HJI**	**UA-BO**
Incremental	[49, 97]	[103, 77]	-	[23, 96]
Online	[56, 119]	[113, 114, 71]	-	[59, 56]
Library Based	[19, 74]	[104]	[43, 33]	[43, 20]
Offline	[73]	-	[123, 9]	-

We compare recent studies' learning times for safety certificates, Fig. 2.7, overlaid on to two function spaces for exponential time algorithms and polynomial time algorithms. The figure is not a direct comparison of computation times due to each study being processed on separate hardware but is meant to highlight ballpark processing times for low dimensional systems and the computational challenges due to dimensionality. Also each technique varies in sophistication, for example in Singh et al. 2019 the algorithm acutally performs both a regression and safety certificate optimisation as a bilinear problem instead of just verifying safety of a known function. Also overlaid on the figure is data collected as part of this papers' study on benchmarking safety certificates using SoS, LQR, CCM and SDSoS, DSoS techniques. Data from Ahmadi and Majumdar 2019 study [5] is also shown. Note that the Yalmip implementation uses the Matlab Timit function as well as the Optimizer function in Yalmip which removes any high level overheads in the processing, more closely related to embedded timings, whereas the DSoS/SDSoS is highlevel Matlab code. For flight controllers a test procedure to optimise control in flight must be certified in a reasonable time frame to both maximise fuel efficiency and minimise the test flight envelope. We define approximate time ranges for online processing as less than 0.1s as the threshold for robotic systems updating control policies. An incremental batch processing

upper limit is set to 500s as test flights lasting hours would be expected to perform several learning manoevers in one flight, 500s would allow 7 learning loops to be peformed per hour as a worst case. Offline learning greater than 500s is where data would be collected in flight and processed at a ground station before updating the control policy for further testing. For all algorithms the challenge is to remain within the polynomial time space but the learning complexity means that they typically lie within the exponential region by adopting partial gridding or discretisation of the state space in some form.

In the following sections we briefly discuss some of the key approaches in safe learning. We start with linearisation techniques and then progress onto more complex extensions of linear techniques such as convex approaches and trajectory validation via robust tubes. We then discuss non-convexity and nonlinear approaches via neural networks and bounded regions of attraction using a Lyapunov function. We futher discuss online optimisation using nonlinear model predictive control and explicit library based techniques using HJI reachability before briefly discussing probability based approaches using Bayesian optimisation.

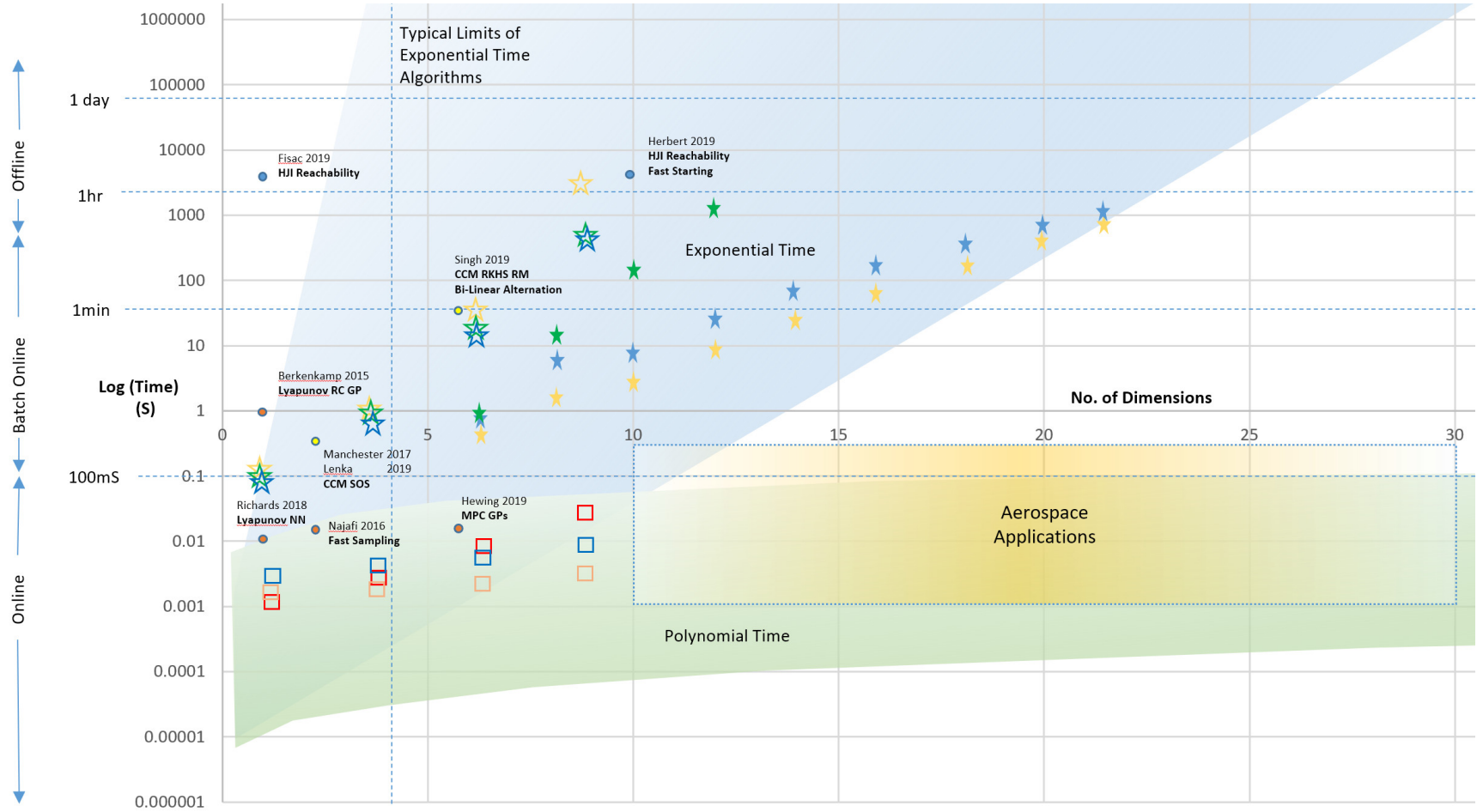


Figure 2.7 – Safety Certificate Computation Times vs Number of State Dimensions. Overlaid is benchmark data from this paper: Squares represent the Yalmip Optimizer timings for Blue - Convex Optimisation Lyapunov Technique, Red - CCM SoS technique, Orange - LQR ARE technique. Stars represent a standard Matlab implementation of SPOT code for Green - SoS technique, Yellow - DSoS technique, Blue - SDSoS technique. Solid stars are from Ahmadi and Majumdar 2019 paper on robotic safety certificates using DSoS and SDSoS [5]

* Note: computation times are not directly comparable between studies as they are processed on different hardware specifications. The plot is a rough guide to help explain the computational challenges.

Table 2.9 Overview of Salient Techniques in Safe Learning of Control Systems with Key Attributes

Approach	Offline\Online	Local \Global	Uncertainty Aware	Optimisation Technique
NMPC	Online	Local	No	NLP Solvers & SDPs
SQP	Offline & Online	Local & Global	No	SDP
CCM	Offline & Online	Local & Global	No	SDP, Alternation & Gridding
UA	Offline & Online	Local & Global	Yes	Bayesian Optimisation
HJI	Offline	Global	No	Differential Game Theory
CPA	Offline	Global	No	Mixed Integer Programming
LQR-F	Offline	Local	Yes	SDP
RL	Offline	Global	No	Gradient Descent
IRL	Offline	Global	No	Algebraic Riccati Equation

2.4.1 Linearisation Techniques

The fundamental linearised control framework is LQR with feedback control and is used widely for systems where nonlinearities can be approximated locally by linear equations. To accomodate approximations over the entire state space parameter varying techniques are used. Standard techniques to linearise system dynamics include Taylor series expansion [3] up to a suitable degree of accuracy or taking the Jacobian of the system at piecewise locations to prove stability via quadratic functions. A standard linear feedback technique with a CLF safety certificate is shown in Fig. 2.8. Nonlinear dynamics that can be approximated by linear piecewise varying functions are termed Continuous Piecewise Affine (CPA) [19, 92, 36]. In CPA techniques the state space is partitioned into polyhedral regions where system constraints and optimal linear approximations are stored as a lookup table. The solution is generated offline due to the typical scale of the programs that employ mixed integer programming (MIP). These techniques are extensions of traditional gain scheduling or linear parameter varying (LPV) techniques. The generic robust

linear control equation is given by:

$$\dot{x} = A(\alpha)x + Bu \tag{2.3}$$

where A and B are stabilisable and controllable system matrices via the optimal control action $u^* = -Kx$. K is the feedback gain matrix and rearranging $\dot{x} = A(\alpha) + B(-Kx)$ forms the closed-loop solution, $\dot{x} = (A(\alpha) - BK)x$. The cost or performance function $J(x, u)$ is quadratic in x and u via postive semi-definite symmetric weighting term Q and positive definite term R, $J(x, u) = \frac{1}{2}\int_0^\infty (x^T Qx + u^T Ru)dt$. Linear systems can be solved in real time via the Algebraic Riccati Equation (ARE), where $-Q = PA + A^T P - PBR^{-1}B^T P$, which generates a positive definite P matrix that certifies the system as stable over a region of the state space for a bounded uncertainty in α. The gain term K can then be calculated from the equation $K = R^{-1}B^T P$. The CLF, also termed the safety function or storage function, $V(x, t)$ is the quadratic state equation weighted by the ARE P matrix, $V(x, t) = x^T Px$. The linearised equation with uncertainty can also be converted from a nonconvex problem to a convex one where a congruence transformation can be used as described in [25].

$$\begin{aligned} &\min_{P,K} \quad tr(P) \\ &\text{s.t. } (A + BK)^T P + P(A + BK) \prec -Q - K^T RK \end{aligned} \tag{2.4}$$

The problem is transformed via $Y = P^{-1}$, and $L = KY$, applying a Schur complement to obtain 2.5.

$$\begin{aligned} &\max_{Y,L} \quad tr(Y) \\ &\text{s.t.} \begin{bmatrix} -(AY+BL)^T-(AY+BL) & Y & L^T \\ Y & Q^{-1} & 0 \\ L & 0 & R^{-1} \end{bmatrix} \geq 0 \end{aligned} \tag{2.5}$$

Yalmip is used with the Mosek SDP solver to find the optimal P matrix as opposed to using the standard ARE formulation. Once P is found a gain matrix K is obtained as before from $K = R^{-1}B^TP$.

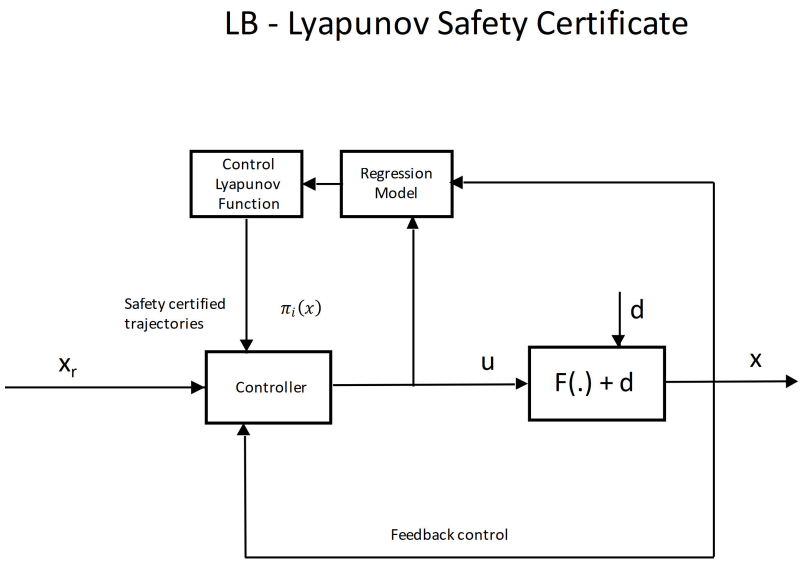


Figure 2.8 – Learning Based Control Lyapunov Function Approach

2.5 Convex Optimisation

Convex optimisation makes use of the convex property of functions 2.9, defined by a connecting line between two points on the function being above their end points at all intermediate locations, and can be solved efficiently by interior point methods or active set methods, [26]. The general formulation for a convex optimisation is:

$$\begin{aligned} \min \quad & J(x) \\ \text{s.t.} \quad & f_i(x) \leq b_i, \quad i = 1, ..., m, \end{aligned} \tag{2.6}$$

where $J(x)$ and all $f_i(x)$ functions are convex.

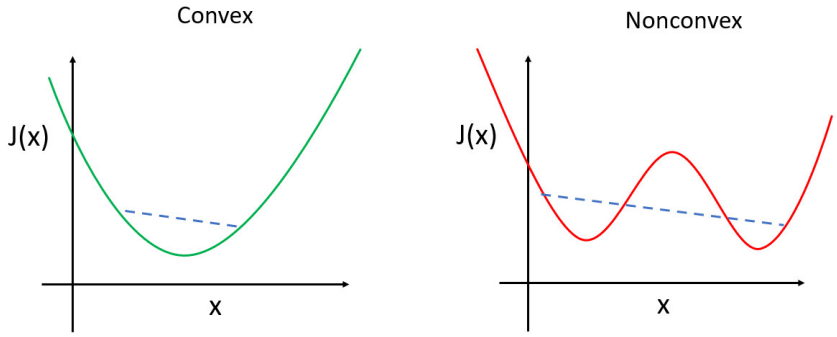


Figure 2.9 – A convex optimisation function compared to a nonconvex function. A line segment connecting two points remains above the function for the convex case but is intersected by the function in the nonconvex case.

One such function is the SoS polynomial and has become popular for proving positivity of polynomial systems in control theory [69, 72, 3, 25]. Some recent studies into SOS convex optimisation are given in Table 2.10. A multivariate polynomial is a SoS if there are a set of polynomials $f_i(x)$, for $i = 1, ..., m$, such that:

$$p(x) = \sum_{i=1}^{m} f_i^2(x) \tag{2.7}$$

The standard method involves enforcing symmetric positive definite matrix properties on Q for the quadratic equation $p(x) = x^T Q x$, which is itself a SoS of the state vector x. The technique is extended to nonlinear systems via the use of monomial functions of the state vector, $p(x) = m(x)^T Q m(x)$. The SoS formulation is therefore reduced to a linear matrix inequality (LMI) problem. Semi-definite programming (SDP) parsers and solvers such as SOSTOOLS [91], Yalmip/Mosek [68, 12] or SPOT [81] can be used to find solutions to LMI problems efficiently. In the case where positivity is not possible

globally, which is typical in systems with constraints, a local relaxation is used via the *Positivestellenzatz* or S-procedure whereby an extra polynomial term, $L(x)$ relaxes the function derivative positivity-constraint, (2.15), [72]. Fig. 2.10 is an example of this technique adapted from [111]. The optimisation is a SoS formulation, Eqs. 2.8-2.12.

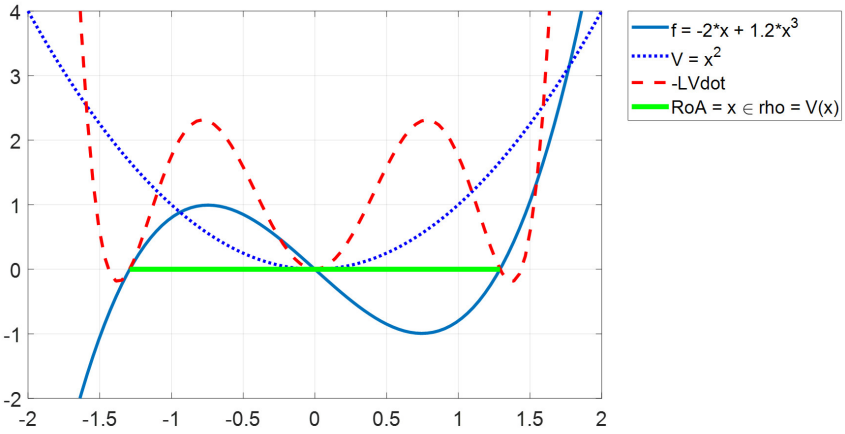


Figure 2.10 – An Example of the Positivestellensatz use of a positive polynomial in one dimension to discriminate between regions of a function that are stable and maximises the level set ρ of the safety function $V(x)$.

$$\max_{L(x),V(x)} \quad \rho \tag{2.8}$$

$$s.t \quad V(x) = x^2 \tag{2.9}$$

$$x^T x(V(x) - \rho) - L(x)\dot{V}(x) \quad SoS \tag{2.10}$$

$$L(x) \qquad\qquad SoS \tag{2.11}$$

$$f(x) = -2x + 1.2x^3 \tag{2.12}$$

Recently, inner approximations of the SoS matrix equations have been developed [3]. By pruning the matrix equations to prioritise the diagonal entries, a diagonally dominant algorithm DSoS and scaled diagonally dominant SDSoS algorithm have been shown to optimise solutions in faster time frames at the expense of more conservatism. DSoS and SDSoS are faster to process compared to SoS optimisation above 10 dimensions and significantly faster above 30 dimensional system optimisation. Currently the implementation

is available in SPOT, [81]. Pre and post-processing of SoS has also been carried out by Löfberg, [69] in the Yalmip parser to improve optimisation times. [126] recently developed a sparse SoS (SSoS) formulation of SoS polynomials, which offers a middle ground between full SoS formulation and DSoS/SDSoS formulation and in some cases offers faster computation times as an SDP with less conservatism.

Table 2.10 Recent Studies Using Convex Optimisation Techniques

Author	Technique	Dim.	Safety Cert.	Soln Time	Solver
[4], (Ahmadi & Majumdar 2019)	SDSoS	30D	CLF	22.5m	Mosek
[3], (Ahmadi & Majumdar 2015)	SDSoS	16D	CLF	15m	Mosek
[105], (Smith et al. 2019)	SLS & SoS	4D	CLF	<5m	Mosek
[84], (Meng et al. 2019)	SoS & H_∞ Cont.	-	H_∞	-	-

2.6 LQR-Trees and Funnels

Funnels can be thought of as the forward or backward propagation of RoAs in time. At discrete points in a state space trajectory, as long as the system is confined to a trajectory within the funnel, the system is verified as stable. Tedrake et al. [112], introduced the Linear Quadratic Regulator Tree (LQR-Tree) algorithm for feedback motion planning that computes a tree of LQR stabilised trajectories with verified stable RoAs using SoS techniques. A diagram of the LQR-Tree library based approach is shown in Fig. 2.11. The exploration of the state space as a sequence of funnels forms a tree-like structure and avoids the curse of dimensionality resulting from discrete state exploration techniques adopted in dynamic programming. The LQR-trees algorithm is limited with respect to the number of dimensions a system has however, due to the computational requirements for verification using SoS. Tedrake et al [112], have applied LQR-Trees to systems of up to five dimensions. For many systems, LQR-Trees are limited to offline computation of the funnels, but this however makes them unsuitable for unknown environments or time-

varying conditions as they rely on the a priori model [73]. Majumdar et al [73] propose a library of funnel systems computed offline that can be adopted online to over come the computation of large SoS problems. The library can be used for stability and obstacle avoidance by verifying the geometric overlap of the funnels with trajectory obstacles in real time. Branicky et al [27], developed a hybrid controller using LQR funnels at each level of the controller to drive the trajectories towards a goal state within a RoA. Branicky et al. term the LQR, Time-Varying-LQR (TVLQR). A hybrid controller switches between control policies via an indexing parameter that changes the controller at fixed regions within the state space or via threshold constraints. In this manner, highly non-linear systems or robotic systems with discrete behaviour can be controlled more easily by adopting localised controller behaviour. The switching of controllers however is not without problems due to the lack of control theoretic proofs for stability and as in the case of sliding mode control can induce oscillations at the boundary.

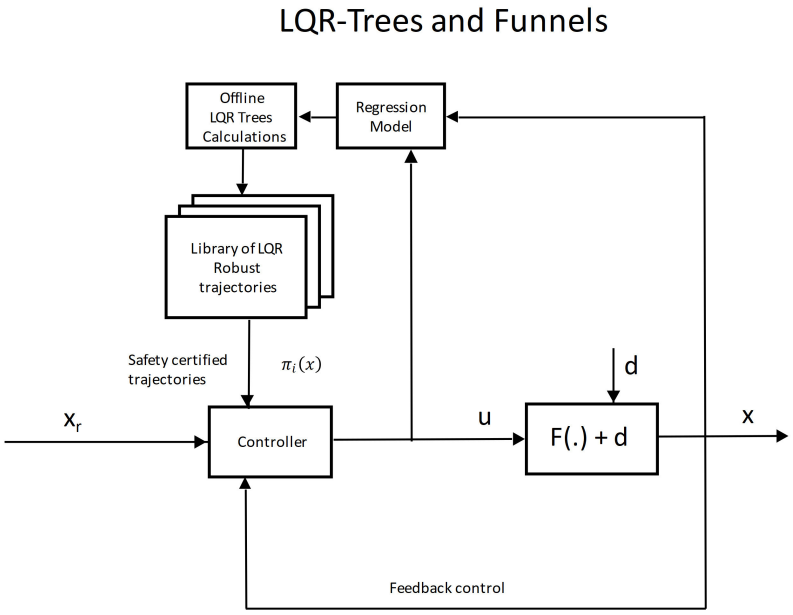


Figure 2.11 – LQR-Trees Approach

2.7 Nonconvex Techniques

Nonconvex techniques are needed for problems with multiple optimisation parameters or when the system is highly nonlinear. In the first case Lyapunov techniques using SoS optimisation to find a stabilising controller where the objective includes maximising the region of attraction are bilinear [72]. The SoS optimisation equations 2.13-2.16 optimise the region of attraction level set value ρ while searching for the Lyapunov function $V(x)$ and the condition $\dot{V}(x) = \frac{\partial V}{\partial x}(f(x) + B(x)u(x))$, is negative where $L(x)$ is a positive polynomial enforcing slackness of the constraints on $V(x)$. Eq. 2.15 is linear in the decision parameters $L(x), u(x)$ when $V(x)$ is fixed and linear in $V(x)$ when $L(x), u(x)$ are fixed and can solved via 2 convex subproblems, alternating between the two [72], 2.12.

$$\max_{\rho, L(x), V(x), u(x)} \quad \rho \tag{2.13}$$

$$s.t \quad V(x) \qquad\qquad SoS \tag{2.14}$$

$$-\dot{V}(x) + L(x)(V(x) - \rho) \quad SoS \tag{2.15}$$

$$L(x) \qquad\qquad SoS \tag{2.16}$$

$$\text{step 1} - \frac{\partial V}{\partial x} f(x) - \frac{\partial V}{\partial x} B(x)u(x) + L(x)(V(x) - \rho)$$
$$\text{step 2} - \frac{\partial V}{\partial x} f(x) - \frac{\partial V}{\partial x} B(x)u(x) + L(x)(V(x) - \rho)$$

Figure 2.12 – Expansion of Eq. 2.15, a bilinear optimisation using alternation between convex functions [72]. In each step the black functions are fixed, blue functions are fixed decision variables, red decision variables are found in the optimisation step, and green is the optimisation objective.

Several authors have adopted bilinear optimisation approaches to solve these types of problems, [103, 2, 73]. The downside is the loss in efficieny in finding a solution due

to the increased complexity and that no formal guarantees can be made on the solution. The techniques also require feasible initialisations to promote convergence to local optima typically via LQR approximations [72]. Nonconvex problems can adopt nonlinear solvers that use gradient descent methods which are commonly adopted in neural networks (NNs). For safety critical applications however, the use of nonlinear solvers is generally avoided due to both latency and reliability in finding a solution, [110].

2.8 Neural Network Approaches

Neural networks (NN) are popular for their speed in handling high dimensional data as a universal function approximator and for real time inference. They are also becoming more prevalent in control systems in research but yet to be considered suitable for safety critical applications due to their inherent non-determinism, [37]. Richards et al [96], present a NN method to learn accurate safety certificates for nonlinear, closed-loop dynamical systems. The NN technique offers the least conservatism over the true safe set when compared to LQR and SoS techniques. [32, 35] use NNs with 2 steps to both calculate the maxmium RoA for a CLF and then to verify that the RoA satisfies the Lyapunov conditions. The solution sets are checked in a falsifier or verifier step with counter examples to verify the RoA. The NN approaches produce larger regions of attraction compared to SoS and LQR techniques. The disadvantage of using neural networks is that they typically require large datasets from which to learn from and the solutions are not explainable in terms of tractable algebra. Also the bias-variance tradeoff is prone to overfitting without system expert knowledge. On fast linear model using deep neural networks (DNNs) is explored in [16] for online learning. The technique uses a Modified Relay Feedback Test (MRFT) to design controller gains and learns the system dynamics in seconds for a quadcopter without prior knowledge of the system. Stability is guaranteed by the design of the system in the identification and control phases. Further work is needed to understand application to nonlinear controllers as a gain schedule.

2.9 Lyapunov Based Sampling Techniques

In many systems, the online estimation of a region of attraction is intractable. Research has been made into sampling-based techniques where the exploration of a region of attraction is made by pointwise analysis of the system behaviour to approximate the RoA for real-time applications [89]. Bobiti et al [24], propose an automated sampling based Lyapunov technique to define a bounded set of states using multi resolution sampling and hyper rectangles as basic sampling blocks. They demonstrate the performance via benchmark examples. [89] proposed a fast sampling technique for the RoA for real-time applications. The authors exploit the local continuity of sampled points to generalise the bounds and use patching theorems to bind validated Lyapunov functions to a local function. Simulation and experimental results showed that the sampling technique quickly estimated RoAs within seconds for a second-order dynamical system. The limitations of the sampling based techniques are the lack of formal guarantees on the solution and that knowledge of the function approximation must be known to choose hyperparameters suitable for efficient sampling.

2.10 Nonlinear Model Predictive Control Techniques

Nonlinear model predictive control (NMPC) uses an accurate nonlinear system model to predict trajectories over a time horizon. A diagram of the learning approach is shown in Fig. 2.13. By using the computation of trajectories constraining the start and end points of the solutions an optimal control policy can be iterated over by using shooting methods. NMPC has recently been employed for both trajectory optimisation in learning algorithms [49], and for safety certification in combination with stochastic uncertainty, [119, 55]. A recent review of MPC techniques is detailed in [56] and some recent studies are shown in Table 2.11. The NMPC scheme is based on solving an Optimal Control Problem (OCP) at every time step and then implementing the first control input iteratively over a time horizon, T. The cost function also serves as the quadratic Lyapunov function $V(x,t)$ to

ensure stability asymptotically for each trajectory solution, 2.17. By nature, the NMPC problem becomes a zero order hold discrete time problem, where x and u are indexed iteratively by subscripts k so that signals x_0, x_k, u_k are shifted in time at each iteration. A detailed description can be found in [83].

$$\begin{aligned} \min_{u \in U_T(x_k)} \quad & J_T(x_k, u) \;=\; \int_{t_k}^{t_k+T} \left(x^T Q x + u^T R u\right) \\ s.t \quad & \dot{x} \;=\; f(x) + B(x)u \\ & \forall t \in [t_k, t_k + T] \\ & x \in \mathcal{X} \quad \forall t \in [t_k, t_k + T] \\ & u \in \mathcal{U} \quad \forall t \in [t_k, t_k + T] \end{aligned} \tag{2.17}$$

The advantage of optimisation techniques is that constraints can be included as control saturation inputs, dynamic constraints, and any state constraints as linear or nonlinear equality or inequality equations. Stochastic techniques are a significant challenge for control theory proofs however, due to the nonlinear propagation of Gaussian distributions, [56]. Recently Grandia et al. [49] have integrated CLFs with MPC for a 4D Segway robot by either enforcing exponential convergence of the Lyapunov cost function at every MPC iteration or by restricting each iteration to a Lyapunov level set. The algorithm uses SQP with a modified Hessian. They show that compared to NMPC no tuning parameters are required and the algorithm performs smoothly to set point changes, whereas a myopic CLF-QP solver causes abrupt changes in control for changes to set point commands. Rosolia et al. [97] implement a Learning Model Predictive Controller (LMPC) that linearises the dynamics locally and uses quadratic programming to minimise the lap time of a racing car. A finite time OCP is solved in 10ms with a sampling time of 10Hz and time horizon of 12s. [119] use MPC as a safety filter. [83] show how a relaxed Lyapunov asymptotic stability criteria can be used with NMPC for a holonomic 2 wheeled robot to ensure trajectory tracking to a set point. The multiple shooting algorithm is able

to sythesize a controller for a prediction horizon of 2 or 3 seconds and 10 Hz sampling time for a reasonable accuracy. [38] implement a NMPC technique via the use of Recurrent Neural Networks (NMPC/RNN) to synthesize a controller at up to 30Hz for a model helicopter autorotation maneuver. Although there are no stability certifcates the technique uses quadratic cost functions amenable to Lyapunov analysis. The technique updates the controller after a suitable number of epochs have evolved from the RNN. The possibly nonconvex optimisation is augmented with random sampling to increase convergence guarantees. Speed of optimisation is also achieved via parallel computation of algorithm equations.

Wang et al. [122] compute control actions for a problem with 12 states and 3 controls with a time horizon of 30 time steps in around 5 ms by using convex optimisation and QP for a linear system. It is thought that the algorithm using Newton descent and line search techniques could be extended to nonlinear systems, either via linearisastion at each time step or larger numbers of iterations.

[50] design a fault-tolerant control 6-degrees of freedom (6-DOF) scheme based on a fast NMPC for real-time implementation for extreme maneuvers. Using a direct multiple-shooting approach and the ACADO Toolkit with SQP. Control synthesis times of 30 ms were achieved for two examples of extreme maneuvers.

[1] use a linear internal model with a reduced number of prediction points that are not evenly distributed along the prediction horizon. Trajectory tracking control was validated through both simulation and experimental testing with the Qball-X4 UAV with reduced onboard computing capcity a standard MPC algorithm failed whereas the optimised algorithm was successful although only for simple trajectories and without stability gaurantees.

NMPC has the potential to optimise control policies most effectively if the requirements are met that the model represents the system acurately enough over the time horizon and that the control policy has an adequate response time for the dynamics problem being solved. For highly dynamic and high dimensional systems this approach continues to

pose real time computational constraints that may become less problematic as technology evolves in the future.

Table 2.11 Recent Studies Using NMPC Techniques

Author	Technique	Dim.	Synth. Time	Solver
[83], (Mehrez 2020)	Multiple Shooting	3D	10s	CasADi
[67], (Leung 2017)	Multiple Shooting	3D	8.8s	ACADO
[122], (Wang & Boyd 2010)	Convex Opt	12D	5ms	-
[38], (Dalamagkidis 2011)	RNN	6D	30Hz	-
[50], (Gros 2012)	Multiple Shooting	6D	30ms	ACADO
[97], (Rosolia 2019)	Linear QP	2D	10ms	OSQP
[55], (Hewing 2019)	Sparse Sampling GP	2D	20ms	FORCESPro

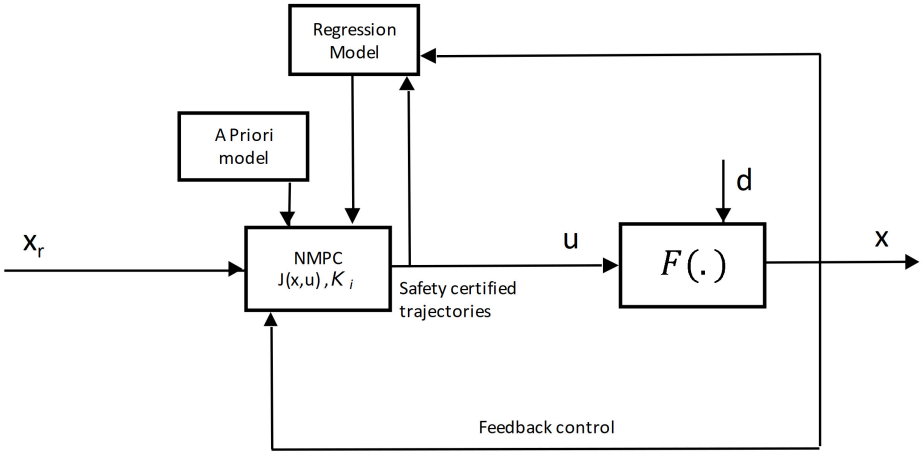


Figure 2.13 – Learning Based - NMPC Approach

2.11 Contraction Theory Based Techniques

Contraction theory has been developed as a differential dynamics extension of the more established CLF techniques, which exploit linear eigenvalue analysis [70]. A detailed description of a contraction metric formulation using sum-of-squares techniques is given in [15] and its extension to CCMs is described in [75]. A diagram of the appraoch is

shown in Fig. 2.15. Contraction metrics refer to the vector field differential, $\delta_{x(t)} = (x(t) - x^*(t))$ of the system trajectories under the control action $u(x,t)$. A CCM forms a safety certificate for stability in the sense of a differential Lyapunov function, $V(x, \delta x)$, to ensure convergence of system trajectories to an equilibrium state via online controller synthesis. A metric is computed offline or incrementally online from sample data by learning the system dynamics.

The solution is computed in two stages, first the computation offline of the solution to an LMI, to derive the metric inverse $W(x,t) = M(x,t)^{-1}$ and a closed loop gain function $\rho(x,t)$ that will be used as an input for the feedback control synthesis. Then an online control policy is calculated using the Metric to find a minimising geodesic between the current state and a reference state on a Riemannian manifold. The technique bridges the gap between conservative but fast linearisation methods like LQR and computationally expensive but explicit NMPC techniques [67].

$M(x,t)$ is a Riemannian metric commonly used in robotics and high dimensional control problems. The metric defines a vector field as a manifold in the state space where the tangent space describes the flow of the system trajectories. The length $\sqrt{\delta x^T M(x,t) \delta x}$ provides a local distance measurement and notion of orthogonality.

In contraction theory, an initial virtual displacement, δx_0, of a pair of flow lines or trajectories, Fig. 2.14, is measured downstream, δx_t, over the time derivative, dt.

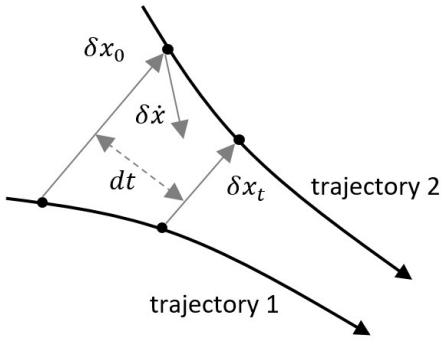


Figure 2.14 – Contraction theory diagram, Adapted from [70].

The squared distance between two trajectories can be defined as $\delta x^T \delta x$. Where the rate of change of the distance can be expressed as:

$$\frac{d}{dt}({\delta_x}^T\delta_x) = 2{\delta_x}^T\dot{\delta}_x = 2{\delta_x}^T\frac{\partial f}{\partial x}\delta_x \tag{2.18}$$

The largest eigenvalue of the symmetric part of the Jacobian $\frac{\partial f}{\partial x}$, the largest eigenvalue of $\frac{1}{2}\left(\frac{\partial f}{\partial x} + \frac{\partial f}{\partial x}^T\right)$, can be substituted to give a scalar form of the squared rate of change in displacement:

$$\frac{d}{dt}({\delta_x}^T\delta_x) = 2\lambda_{max}\delta x^T\delta x \tag{2.19}$$

As a result, if the largest eigenvalue, λ_{max}, of the differential dynamics, remains negative by a certain threshold amount $-\beta$, for all x and t then stability can be inferred, 2.20, [70].

$$\|\delta x\| = \|\delta x_0\|\, e^{\int_0^t \lambda_{max}(x,t)dt} \tag{2.20}$$

Furthermore, if a system's local set of trajectories can be shown to be contracting, then the system is said to be locally dynamically stable over the time horizon t. An important generalisation of the virtual displacement vector to an eigenvector basis via a transformation, $\delta z = \Theta\delta x$, gives a generalised distance framework via a Metric function $M(x,t)$, (i.e. $\delta z^T\delta z = \delta x^T\Theta^T\Theta\delta x = \delta x^T M\delta x$). Where $M(x,t)$ is a symmetric and continuously differentiable metric and positive definite for exponential convergence of the system, [70]. $M(x,t)$ is a Riemannian metric commonly used in robotics and high dimensional control problems. The metric defines a manifold in high dimensional space with an associated tangent space of the inner product $\delta x^T M(x,t)\delta x$. This provides local length and orthogonality. A smooth curve $\gamma : [0,1] \to \mathbb{R}^n$ with $\frac{\partial\gamma}{\partial s} \neq 0\ \forall s \in [0,1]$. Define $e(s,\gamma) = \gamma_s^T(s)M(\gamma(s))\gamma_s(s)$ where $\gamma_s = \frac{\partial\gamma}{\partial s}$. Description taken from [67]. The length of the curve with the end points at $\gamma(0)$ and $\gamma(1)$ is defined as:

$$l(\gamma) = \int_0^1 \sqrt{e(s,\gamma)}ds \tag{2.21}$$

The shortest path between two points on a manifold is called the geodesic. This is

the infimum of curves defined $l(\gamma)$. The squared length is termed the energy $E(\gamma)$ and is more conveniently computed by:

$$\underset{\gamma \in c[0,1]}{\operatorname{argmin}} \int_0^1 e(s,\gamma) ds \tag{2.22}$$
$$\gamma(0) = x^*(t),\ \gamma(1) = x(t)$$

The solution to the CCM is found from a Linear Matrix Inequality (LMI).

$$-\dot{W} + WA^T + AW - \rho BB^T < -2\lambda W \tag{2.23}$$

where A is the system Jacobian and B is the control matrix. The inequality is a convex optimisation in $W(x,t)$ and $\rho(x,t)$, where the contraction rate, λ, is either fixed or optimised using bisection. W and ρ are approximated by choosing suitable basis functions and then enforcing the constraints by gridding over a region of the state space [102]. The solution is also found when it is possible to structure the dynamics function as a sum-of-squares to certify Lyapunov stability under the control constraints [78]. The resulting $W(x)$ and $\rho(x)$ form the stabilising differential control equation:

$$u(t) - u^*(t) = -K(x(t) - x^*(t)) \tag{2.24}$$

$$\textit{where} \quad K = -\frac{1}{2}\rho(x) B^T W(x)^{-1} \tag{2.25}$$

In [71], the authors extend the use of CCMs to a direct adaptive control approach for stabilisable nonlinear systems with parametric matched or extended matched uncertainty. The authors in [77], extend the use of CCMs to Robust control (RCCM) using an L_2 gain bound and quadratic cost where set invariance is enforced in the LMI via the use of Lagarange multipliers. An LMI is solved for a 2-dimensional Moore-Greitzer system in 1.3s using Yalmip and Mosek. Singh et al. [103] use a CCM approach to learn stabilisable

dynamics by regularising a dynamics optimisation problem online from demonstration data. The dynamics and metric subproblems are solved simultaneously via alternation over the biconvex problem formulation using semidefinite programming, line search, and localised gridding over a number of training samples. In [102] an online solution to a 6-state planar quadrotor trajectory optimisation is found using a robust CCM with gridding over 4 state space constraints taking 60s offline to solve. The online controller optimisation is performed via pseudospectral collocation using the SNOPT solver in around 4ms, adequate for a 1s resolve time online. [71] use a control contraction metric technique using SoS optimisation in Yalmip for the Moore-Greitzer jet engine system. The metric was computed in under 0.45s for up to degree 6 polynomials. They highlight the global solution properties as being favourable over LQR solutions.

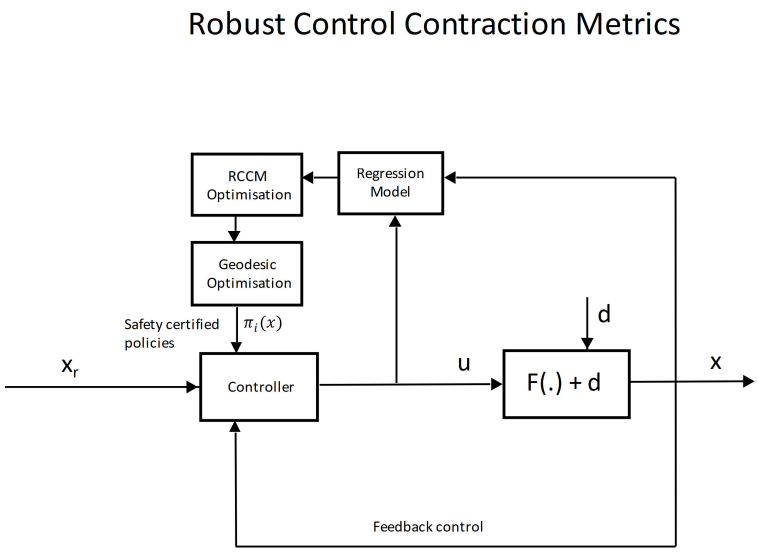


Figure 2.15 – Learning Based - RCCM

The advantage of the CCM technique is that the online controller synthesis MPC problem is simpler than full NMPC problems due to the lack of dynamics constraints and reduced dimensionionality if partial contraction is adequate to verify stability. The only downsides to CCM techniques are the use of SoS techniques that can be limited in scaleability and omit areas of the state space that could be explored safely. The topology of the manifold for rotational systems may also mean that control synthesis is limited to

regions within a cut locus boundary without a tailored approach.

For simple problems the Metric is flat (independant of the state space) and the geodesic is a straight line [76] creating a linear feedback solution. For complex problems however the geodesic can be calcuated by both direct optimisation and indirect methods such as Euler-Lagrange two point boundary equations, phase flow, fast marching or graph cuts [78, 67].

Indirect methods are generally intractable though and strictly offline algorithms or require tailored approaches for most nonlinear systems. Direct optimisation methods include shooting methods, quasi-Newton descent and collocation techniques.

In [67] it was shown that the pseudospectral technique using Chebyshev polynomials and Clenshaw-Curtis quadrature is optimal compared to Legendre polynomials and standard Gaussian quadrature but online predetermination of polynomial degree and number of nodes is required to ensure sufficient accuracy of the solution. Recently Finsler manifolds have been combined with the CCM framework to generate controller paths not requiring minimising geodesics [31, 45]. The Finsler framework offers a larger class of metrics to be used and may be an alternative for complex systems where minimising geodesics are too expensive to be compute [31]. Tsukamoto and Chung [113] use a Neural Contraction Metric (NCM) approach for robust estimation and control that uses deep LSTM-RNN network for global approximation of an optimal contraction metric.

2.11.1 Hamilton-Jacobi-Isaacs (HJI) Reachability Techniques

The HJI Reachability technique in the context of control systems refers to a partial differential equation which explicitly defines the controlled systems behaviour in all states. From this equation trajectories of the system can be calculated to map sets within the state space that define reachable states. Reachability analysis seeks to prove that system trajectories from an initial set subject to all possible inputs and disturbances remain within safe bounds. Figure 2.16 shows the library based feedback control framework using offline differential game theory for the safe operating envelope. $\dot{x}_e$ in the diagram represents the

tracking error dynamics used in the framework by [43]. The tracking error dynamics is used to measure the uncertainty in the model due to disturbances affecting the system by which a safe policy is chosen. An overview of Reachability techniques is provided in [18]. Table 2.12 is a list of recent research using reachability based techniques for safety certification. The offline-computation times shown are not meant for direct comparison just an indication of the range of values found in research. For nonlinear systems reachability techniques rely on the solutions of the Hamilton-Jacobi-Isaacs (HJI) equations to compute the safe set of states and actions. The safe, reachable set $\mathcal{S}$ is affected by the system input values $\mathcal{U}$ and a disturbance set $\mathcal{D}$. The goal is to limit the control inputs to remain within the safe set over a given time horizon [7]. The reachability equations for dynamic systems stem from optimal control theory and dynamic programming, where the dynamics function is augmented with a disturbance, $d(s)$, (2.26).

$$\dot{x} = F(x) + B(x)u + B_d(d(x)) \tag{2.26}$$

The optimal control is found from the Hamiltonian equation using differential game theory, where one player, d, is trying to maximise the function and the other, u, is trying to minimise the function.

$$H(x, \tfrac{\partial V(x,t)}{\partial x}) = \min_u \max_d \left(\tfrac{\partial V(x,t)}{\partial x} . f(x,u,d) + c(x,u,d) \right) \tag{2.27}$$

Reachability using HJI computations is typically done offline over a grid of the state space and is subject to the *curse of dimensionality*. The discretisation of the grid also needs to meet the accuracy of the dynamics problem at hand [33].

Gillula et al. [48], adopt machine learning techniques to increase computational efficiency. The technique adopts optimal control actions only when approaching the border of the unsafe set. At all other times, the freedom to explore actions meant a better dynamics model could be learned. In [43], the authors extend their work to create a unified treatment of model learning and guarantee validation and propose a general safety framework

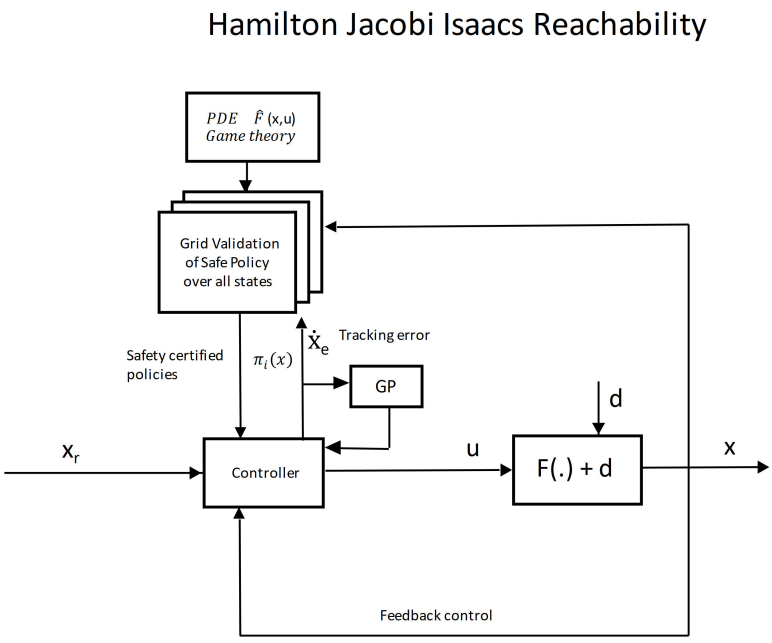


Figure 2.16 – Hamilton-Jacobi-Isaacs Reachability with Gaussian Process Learning Based Safety Function

for learning-based control using reachability analysis to reason online about the validity of computed guarantees. The authors found that it was necessary within the framework to not only ensure safety when a threshold is detected but to also ensure safety when the model assumptions become unreliable. This they term local model reliability. [123] use a NN-Reach tubes technique for a 2D system to create safe sets in 0.383s. Further progress has been made in decomposing higher dimensional systems into subsystems using the Jordan form highlighted in [17, 54]. By solving projections of the dynamics in lower dimensional space and reconstructing higher dimensional envelopes bounded by the projections, a faster computation of reachable sets is achieved, reducing the complexity in the decomposed state dimension vector. Sensitivity to initialisation of the HJI solutions is explored in [54] and a novel technique, FaSTrack (Fast and Safe Tracking) speeds up learning by adopting a coarse model approximation for early planning and tracking model uncertainty to verify that the system is within safe bounds [53]. FaSTrack exploits the convergence properties of model approximations to the true dynamics of the system by updating the planner with the best available knowledge at the earliest possible time.

Occupation Measures Using Reachability

Recent studies into occupation measures as an alterntive to binary classification of safety have looked at a relaxation of unsafe trajectories, [74, 34]. [74] show that by measuring the time a trajectory enters an unsafe region a less conservative safe set can be adopted if a maximum exposure time can be adopted as a safety protocol. They also show that the convex problem is faster to solve than alternation methods using CLF functions. A 4D cart pole system is solved in 15.8s whereas an 8D uBot is solved in 27m in Mosek. Although uncertainty and disturbances are not yet considered for this technique and scaling to higher dimensions is limited due to the computational complexity.

Table 2.12 Recent Studies Using Reachability Techniques

Author	Method	Sys Dim	Offline Comp. Time	Software
[9], (Althoff et al. 2008)	Linearised Zonotopes	30D	704s/746s	-
[53], (Herbert et al. 2018)	HJI-Grid	4D	1542s	ToolboxLS/helperOC
[33], (Chen et al. 2018)	HJI-Grid	6D/10D	30s/2.8hrs	ToolboxLS/helperOC
[43], (Fisac et al. 2019)	HJI-Grid & GP	2D	-	ToolboxLS/helperOC
[123], (Xiang et al. 2018)	NN-Reach Tubes	2D	0.062s-0.383s	NN
[41], (Fedorov & Candelieri 2018)	Fwd Euler App & GP	2D	10s - 200s	Matlab GPML toolbox
[74], (Majumdar et al 2016)	Occupation Measures	4D	15.8s	Mosek
[74], (Majumdar et al 2016)	Occupation Measures	8D	27m	Mosek

2.11.2 Uncertainty Aware Bayesian Optimisation

Uncertainty aware (UA) based techniques promote intelligent exploration via BO and active learning, Table 2.13. For a detailed description of BO, Brochu et al. provide an excellent overview [28]. An overview of active learning can be found in [100]. Fig. 2.17 shows the structure of a control systems that employs a probabilistic guarantee for control policies learned from a regression model. The benefit of active learning is that model uncertainty can be used to speed up training by prioritising exploration within

regions of high uncertainty or high value and can be used to distinguish safe bounds within state constraints. Safe Active Learning (SAL) and Safe Bayesian Optimisation (SBO) are extensions of these techniques to safety critical domains [99]. The techniques share the aim of exploration, determined by an acquisition function which trades off uncertainty for optimisation of an objective function, whilst also keeping the system within safe bounds. Uncertainty provides additional information about a function approximation in the form of a variance parameter about a mean distribution as a function of the state, this is typically achieved via Gaussian process regression.

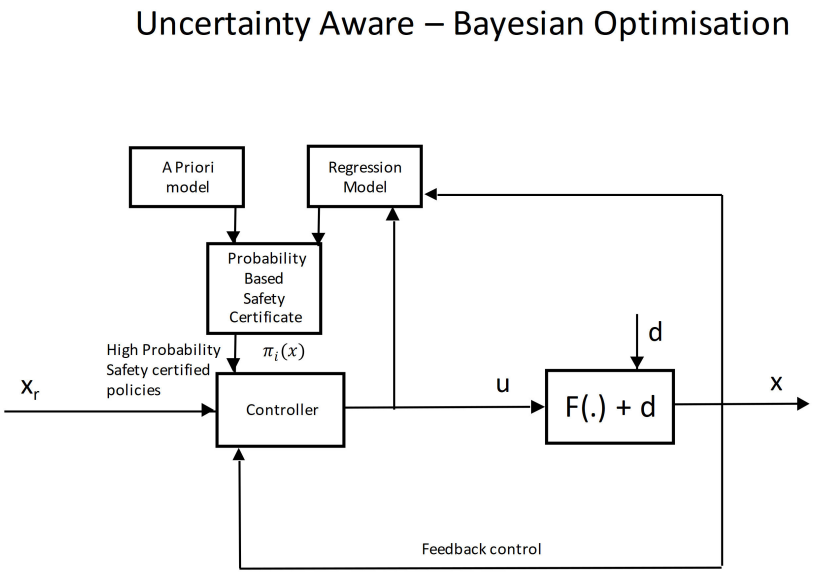


Figure 2.17 – Uncertainty Aware Bayesian Optimisation Safe Learning Framework

In SAL the goal is to optimise a nonlinear objective function $J(x)$, via exploration of the state space, subject to system safety constraints. For example, in [43] the objective was to learn the optimum gains for motor controllers during a quadcopter ascent and descent constrained by a safe distance to the ground subject to disturbances. BO relies on a priori estimates of the model probability distribution [28]. An acquisition function trades off between exploration and exploitation.

At each time step, the function estimate, the uncertainty bounds and the current best-sampled function value, termed the *incumbent*, $f(x^+)$, are updated so that the process

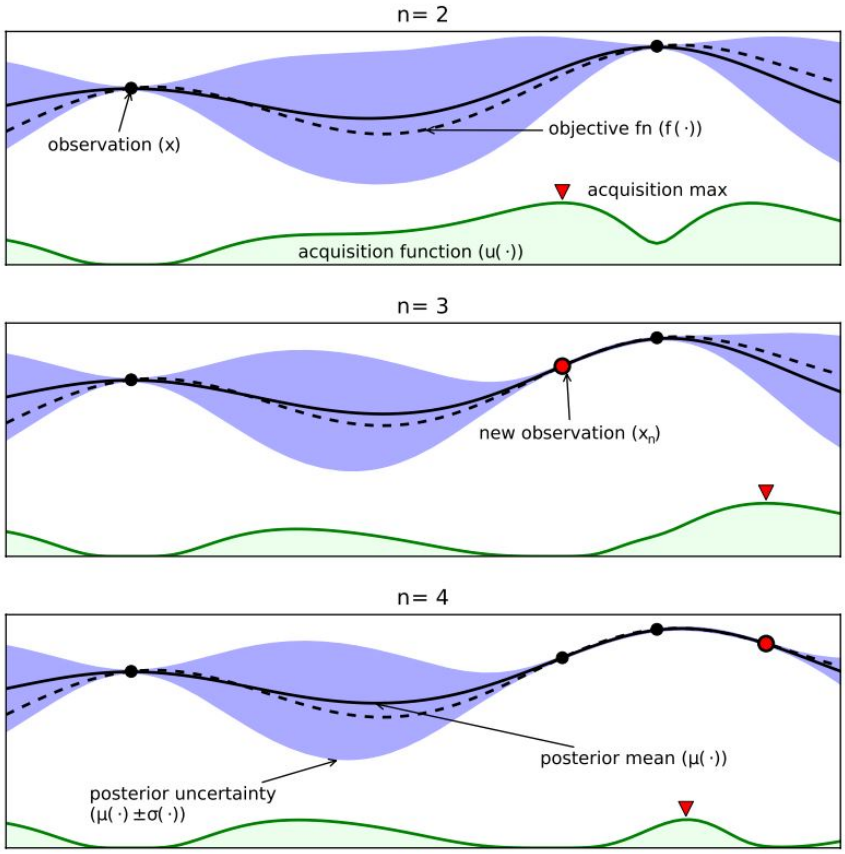


Figure 2.18 – Bayesian optimisation over three iterations. The plots show the mean and confidence intervals estimated with a probabilistic model of the objective function. The acquisition function is high where the model predicts a high objective (exploitation) and where the prediction uncertainty is high (exploration), [27].

of exploration continues via the acquisition function. The acquisition function is ideally high where the highest potential function values will be found.

There are three main improvement based acquisition functions:

1. Probability of Improvement (PI)

2. Expected Improvement (EI)

3. Lower or Upper Confidence Bound (LCB),(UCB).

The SBO approach uses LCB from a Gaussian Process function approximation to decide where to sample new points in the exploration of the system domain.

$$LCB(x) = \mu_{f*}(x) - \kappa\sigma_{f*}(x) \tag{2.28}$$

Here κ is a weighting factor to decide between exploration and exploitation [99]. SBO simultaneously targets a region of minimum loss function for the controller updates.

A program called SafeOpt employs SBO, developed by Berkenkamp and Sui [21, 107].

In Fisac et al. [43], a probability limit is used to switch the control policy to a safe controller when the unsafe reachable set can be reached statistically beyond the safe limits. Once the safety function and safety policy have been computed, they are stored as look-up tables. Uncertainties arise from both the system model and from the uncertainty in environmental disturbances. Kahn et al [59], use an uncertainty-aware collision prediction model. To overcome the risk of exploring collision states, the speed of the robot is reduced when there is high uncertainty. In practice, a balance between safety and optimisation is difficult to find. Recent attempts to reduce computational loads for GPs include combinations of GPs with a nominal controller to only learn nonlinearities [55, 97]. The aim is to reduce sampling times, to be in the millisecond range. Hewing et al, use sparse GPs by selecting just 10 inducing inputs per time step heuristically along the approximate state input trajectory to maintain sampling times to 20ms. Further efficiencies could be made by selectively updating the most important parameters while using less critical parameters from the previous time step to update the dynamics model. Wabersich [120] uses a linear quadratic control Lyapunov function from sampled data points and convex optimisation within a safety framework. The sampled data is further enhanced by adopting the use of a GP model to provide a priori uncertainty. Experiments show safe set computation in 0.2s. It is not thought however that BO [17] based methods are practical above 30-40 dimensions due to computational loads.

2.12 Function Approximation Techniques

In this section we give a brief overview of a non-exhaustive list of function approximation techniques found in recent studies on learning for control systems applications. Online

Table 2.13 Recent Studies Using Uncertainty Aware Based Techniques

Author	Method	Sys Dim	Stability Cert.
[21], (Berkenkamp et al. 2016)	BO with GPs	2D	Multi-Safety Func.
[23], (Berkenkamp et al. 2017)	Safe MB RL + BO	1D	LQR Lyap. Func.
[63], (Kirschner et al. 2019)	1D-Subspaces LINEBO	40D	-
[59], (Kahn et al. 2017)	Safe RL Col. Avoid.	3D	speed cost
[97], (Rosolia et al. 2019)	Safe RL for Racing Car	3D	K-NN convex set
[99], (Schillinger et al. 2017)	PI-Control SAL	1D	discriminating func.
[108], (Sui et al. 2018)	Stagewise-Safe BO G	2D	1-step decrease cond.
[120], (Wabersich et al. 2018)	Safe RL Pred. Safety Filter	2D	two convex optims.
[118], (Wabersich et al. 2019)	Safe Prob. MPC	6D	nom. MPC safety cert.

learning of control systems requires at least one approximation of a mapping function for the system. Either the full first order dynamics under control or a learned control policy as a function of the state of the system, [117, 95]. We focus on the case where the full dynamics function is approximated in this review, where a controlled system is defined by inputs, u, states, x, and the resultant change in state, $\dot{x}$ by a function derived empirically through observation or expertly through prior knowledge of the physics of the system. The true function, $F(.)$, is therefore approximated as accurately as possible by, $\hat{F}(.)$, using regression techniques of the observed data. For dynamic systems, the function can be approximated by a continuous or discrete function for changes of state given input commands over n observations as a dataset, $\Omega\{x_i, u_i, \dot{x}_i \mid i = 1...n\}$. A function approximated by a model is termed *model-based*, but state-of-the-art techniques also include *model-free* methods when a function is approximated using training data from the system with high dimensional universal function approximators. These models include Gaussian Processes, Neural Networks, Approximate Dynamic Programming and stochastic search

techniques. A list of key techniques adopted for function approximation is shown in Table 2.14, including some of the advantages and disadvantags of each approach. Combinations of approaches have also been used, termed *hybrid* techniques. Hybrid techniques benefit from the efficiency of model-based learning as well as the flexibility of model-free approximations. We review the state-of-the-art techniques in the following section.

2.12.1 Recursive Least Squares

Parametric techniques can be used when the dynamics of a system are known to be well approximated by a set of parametrised features or monomial functions that require regression techniques to fit the functions to the data. This is typically achieved via least squares regression or traditional methods employing gradient regularisation such as Ridge Regression and Least Absolute Shrinkage and Selection Operator (LASSO) techniques. The parameterised models seek to reduce the error between empirical system data and the model approximation, 2.29. In reference to the affine function approximation described in 2.1 for the dynamics of a system the following regression approaches are adopted. In the case of the LASSO technique the regression minimisation uses a regularisation term λ_θ that minimises the L_1 norm of the function parameter vector θ and the p in 2.29 is replaced by 1. Minimising the L_1 norm promotes sparsity of the learned parameters whereas traditional ridge regression uses the L_2 norm which minimises all parameters in the function approximation evenly, [88].

$$\underset{\theta}{\operatorname{argmin}} \sum_{i=1}^{n} \|\hat{F}(x_i, u_i) - \dot{x}_i\|_2^2 + \lambda_\theta \|\theta\|_p^2 \tag{2.29}$$

State space models can be sparsely parameterised and have been developed recently for nonlinear system identification using libraries of basis features [60, 103]. Kaiser et al. [60] propose SINDY (System Identification of Nonlinear Dynamic Systems) and compare its performance to neural networks and linear techniques using extended dynamic mode decomposition. They show how SINDY with control for MPC of a number of nonlinear

Table 2.14 Function Approximation Techniques

Technique	**Recent Studies**	**Advantages**	**Disadvantages**
Kernel Methods	Singh 2019, [103] Berkenkamp 2016, [21]	Tractable linear equations "The Kernel Trick" Function regularisation using distances	Features must be selected Linearity may be conservative
Gaussian Processes	Singh 2019, [103] Berkenkamp 2016, [21] Hewing 2019, [55]	Most accurate unbiased linear approximation Provides uncertainty bounds Robust smoothing properties	Scales poorly with number of dimensions Requires knowledge of hyperparameters
Neural Networks	Berkenkamp 2017, [23] Richards 2018, [96]	Scales well for high dimensional systems Universal function approximation	Non-deterministic Slower to train than linear methods Requires large training set
Sum of Squares	Manchester 2017, [76] Lenka 2019, [66]	Proves positiviy of function Simple quadratic form	No. of coefficients is exponential in poly degree Conservative ellipsoidal approximation

systems outperforms neural networks and dynamic mode decomposition in speed and accuracy. Fundamental to the success of the algorithm is enforcing sparsity using the L_1 norm regularisation and a library of features $\Phi(x, u)$ that adequately describe the system dynamics. A diagram of the RLS SINDYc algorithm is shown in Fig. 2.19, where θ is the 2D weighting matrix or parameters and $\Phi(x, u)$ is the library of nonlinear basis features.

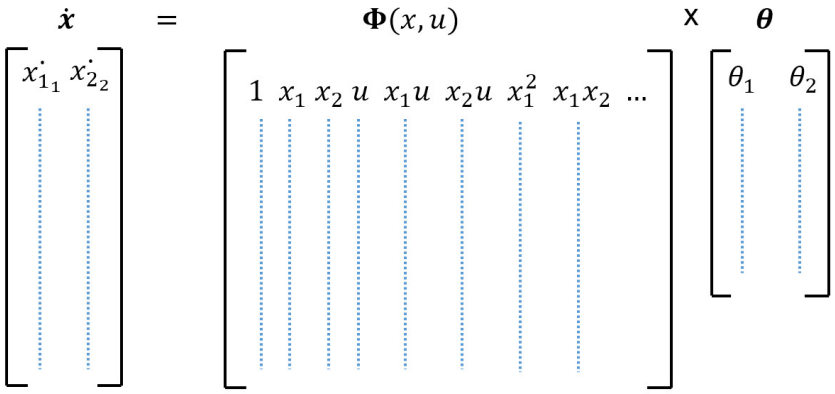


Figure 2.19 – Sparse Identification of Nonlinear Dynamics with Control SINDYc, Kaiser et al. [60]

$$\underset{\theta_i}{\operatorname{argmin}} \|\dot{x}_i - \theta_i \Phi^T(x, u)\|_2^2 + \lambda \|\theta_k\|_1 \tag{2.30}$$

2.12.2 Neural Networks

Neural networks are becoming increasingly common in machine learning applications where high-dimensional input spaces can be iteratively optimised over a network of feature combinations [96, 23]. The number of abstractions in the form of feature layers and the nonlinear optimisation of features via activation functions provide a flexible framework for universal function approximation. The choice of network and method of optimisation must be carefully chosen to avoid overfitting of data so that the function can generalise well to all environments. Deep Neural Networks tend not to generalise well for predictions on unseen data for new environments and can give unpredictable results for outliers within the test data [125]. Heyer et al. 2020, [57], use a single-hidden-layer, fully-connected neural network as parametric structures for an actor-critic, and target-

critic reinforcement learning algorithm. The benefits are high dimensionality with low computational cost and guaranteed differentiable features. However, [127] noted that neural networks may add two sources of errors in applications as flight controllers. One is due to lacking adequate computing power when performing a least-squares approximation of a desired cost-to-go function. The other is the need for offline learning in a simulated environment that may not fully represent the real world. [98] also noted that possible sources of error can arise from reporting practices in the case of training data and from poor model choice which can introduce bias. Nevertheless, their expressiveness and efficiency are desirable attributes if guarantees can be made over their implementation in safety critical systems of which research is ongoing to improve and verify via certification techniques [58, 125]. A single hidden layer neural network is shown in Fig.2.20 without the bias terms included. The NN has a three dimensional input $\mathcal{X}$ and two dimensional output $\mathcal{Y}$ with four hidden nodes. The general expression for a single layer network, 2.31, is the sum of weighted nonlinear activations $\sum_{i=0}^{m} \omega_i^{(2)} \phi_i$ of a weighted input vector $(\omega_i^{(1)})^T x$ plus a constant bias b. The activation function determines when the node is switched on and to what degree, magnitude or sign depending upon the activation function chosen, ie. $\phi(\cdot) := Sigmoid(\cdot), Tanh(\cdot)$, and $ReLu(\cdot)$ activation functions have the respective output ranges $[0, 1]$, $[-1, 1]$, and $[0, x]$. Each layer is a newly weighted dimension of the previous layer and can be rewritten as 2.32. The activation functions create nonlinearities in the neural network but the network without activations would be linear combinations of the input features. There are parallels of neural network topology with polynomials. The maximum number of hidden nodes needed to express a k dimensional polynomial of degree d is k^d although typically fewer nodes are required as functions are typically sparse in nature. Fig. 2.20 shows a neural network with two inputs x and y and the downstream combinations of inputs at each layer for a 2 hidden layer network. In general the hidden layers determine the polynomial degree of monomial terms although a single layer with an infinite number of nodes can approximate any degree polynomial it is less efficient in the use of network connections. A linear function of basis features

can be used instead of a neural network if a library of features is known to be a good representation of the function, as shown in 2.33.

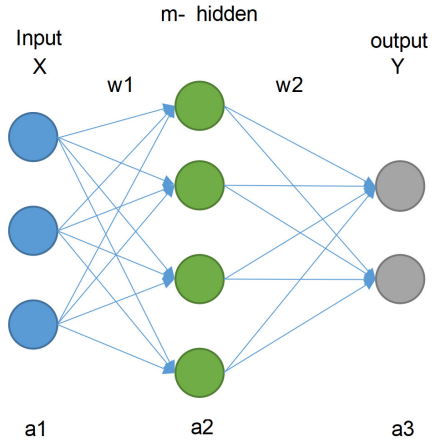


Figure 2.20 – A 3D input, 2D output, neural network with one hidden layer. With activation functions $a1, a2, a3$ and layer weights $w1, w2$. The bias terms are ommited for clarity.

$$y(x) = \sum_{i=0}^{m} \omega_i^{(2)} \phi((\omega_i^{(1)})^T x + b) \tag{2.31}$$

$$y(x) = \sum_{i=1}^{m} \alpha_i \phi(\omega_i^T x) \tag{2.32}$$

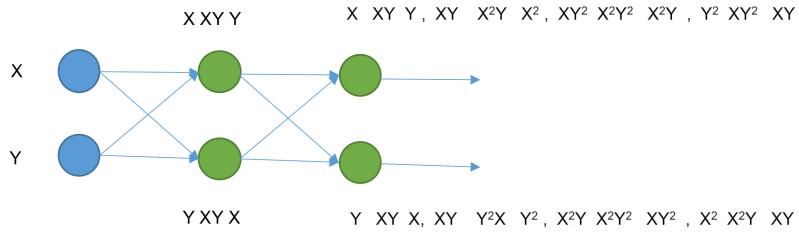


Figure 2.21 – Illustrative representation of a 2D neural network's inputs expanded as monomial terms after two fully connected layers into the output function approximation.

$$y(x) = \sum_{i=1}^{m} \alpha_i \phi_i(x) \tag{2.33}$$

$$e.g. \quad \phi(x) = [x \; y \; xy \;\; x^2 \; y^2 \; x^2y \;\; xy^2 \; x^2y^2]$$

$$\alpha = [\alpha_1, ..., \alpha_m]$$

2.12.3 Kernel Methods

The extension of linear combinations of features as functions is the linear combination of kernels as functions, 2.34. A kernel $\kappa(x, x')$ takes two input values from the independent variable state space and creates a real valued output. When one input value is fixed on an example sampled from observations and is used to construct a function approximation it is termed a support vector. The properties of Kernels being a function of two inputs of the same space mean that they have a reproducing property and symmetry obeying Mercer's Theorem for positive defintite covariant matrices. An overview of Reproducing Kernel Hilbert Spaces (RKHS) is presented in [13] and a comparison of techniques is shown in [88]. As an illustrative example in one dimension, the function $k(x, x') = \sqrt{(x - x')^2}$ is fixed on one co-ordinate of the input space to define a positive function over an infinte range of the second co-ordinate. A weighted sum as a vector of these functions can be used to define a function approximation of the independent variable x, 2.34. Figure 2.22 shows four such kernel functions fixed on four support vectors. In the example the support vectors are just one dimension and two function approximations, f_1, f_2, are shown as weighted sums of the basis kernels. In this way more complicated nonlinear functions can be created to approximate data.

RKHS have been employed in recent studies [79, 23, 103]. The use of RKHS is one technique that makes computation much more efficient for searching function spaces and minimisation tasks. However [104] notes that full matrix-valued expansions grow rapidly with the number of constraint points and state dimensionality which slows computation.

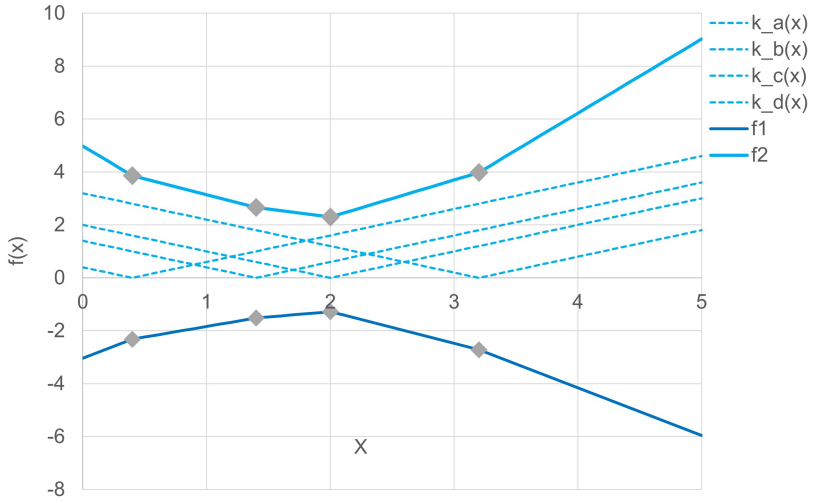


Figure 2.22 – Illustrative example of a 1D Kernel with 4 support vectors defining two function approximations

$$f(x) = \sum_{i=1}^{m} \alpha_i \kappa(x_i, x) \tag{2.34}$$

The kernel function is typically a stationary or shift invariant function or inner product, a real valued function of two input values or vectors, $k : \mathcal{X} \times \mathcal{X}' \rightarrow \mathbb{R}$. The functions can be defined both as linear combinations of kernels or as inner products in a feature space, 2.35.

$$f(x) = \sum_{i=1}^{n} \alpha_i \kappa(x_i, x) \equiv < w, \phi(x) > \tag{2.35}$$

where w is a scalar weighting vector. The inner product of two functions, f and g, given by:

$$< f, g >= \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \beta_j \kappa(x, x_i) \cdot \kappa(x, x_j) \tag{2.36}$$

can be written as the norm in a Reproducing Kernel Hilbert Space, $\mathcal{H}_\kappa$ via a mapping $\phi(x) : \mathcal{X} \rightarrow \mathcal{H}_\kappa$.

$$\kappa(x, x') = \phi(x)^T \phi(x') \tag{2.37}$$

$\kappa(x, x')$ is typically easier to compute than $\phi(x)^T\phi(x')$ and a function can be constructed using the Gram matrix, 2.38, as a linear model for a determined maximum number of data points.

$$\mathbb{K} = \begin{pmatrix} \kappa(x_1, x_1) & \cdots & \kappa(x_1, x_N) \\ \vdots & \ddots & \vdots \\ \kappa(x_N, x_1) & \cdots & \kappa(x_N, x_N) \end{pmatrix} \tag{2.38}$$

In dynamics learning, we are interested in vector-valued basis and so the feature mappings used in Kernels become higher dimensional function classes. See Sindhwani et al [101] for a more complete description of vector valued RKHS. A stochastic search algorithm can reduce the computation time and has been used with matrix-valued kernels to find approximation functions in RKHSs (Minh 2016, Brault et al 2016), [103]. Marinho et al., [79], demonstrate efficient motion planning using inherent smooth trajectories by regularising the norm in an RKHS that minimises the velocity, acceleration, jerk and snap. The benefit of the algorithm is that it can approximate function trajectories smoothly using sparse data.

2.12.4 Gaussian Processes

Gaussian Processes (GPs) are kernel methods that exploit the use of the Gaussian Kernel to provide a powerful nonparametric framework to learn approximations of functions from sample data. A Gaussian Process is a random field of values over a vector space $\mathcal{X}$, which is described by a mean function $\mu : \mathbb{R}^n \rightarrow \mathbb{R}$ and a positive semi-definite covariance kernel function $\kappa : \mathbb{R}^n \times \mathbb{R}^n \rightarrow \mathbb{R}$.

$$f(x) \sim \mathcal{GP}(\mu(x), \kappa(x, x')) \tag{2.39}$$

The interpolation technique is also known as Kriging and is known to be the best linear unbiased prediction of the intermediate function values [88, 47].

The Gaussian kernel, also termed squared exponential or Radial Basis Function (RBF)

is given by:

$$\kappa(x, x') = \sigma_f^2 \exp(-\frac{1}{2l^2}(x - x')^2) \tag{2.40}$$

where l determines the horizontal length scale over which the function varies, and σ_f^2 controls the vertical variation or variance. The kernel defines a measure of similarity between data points and the Gram matrix is a covariance matrix for the Gaussian Process function. Noise free data has diagonal entries all equal to 1. Noisey measurements have diagonal entries with an additional variance term, $(\mathbb{K} + \sigma^2 I)$ where σ is the standard deviation and I the identity matrix.

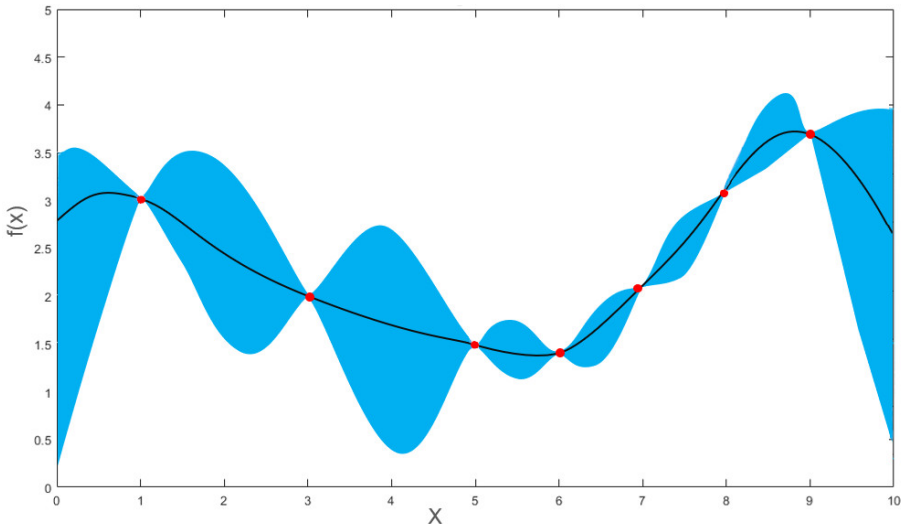


Figure 2.23 – Example of a Gaussian process function approximation with 7 sample points. The blue envelope represents the uncertainty bound of the distribution and the black line the mean function approximation.

A GP is given a prior probability distribution over function values and for a finite set of points, the process defines a joint Gaussian 2.41.

$$p(f|X) = \mathcal{N}(f|\mu, \mathbb{K}) \tag{2.41}$$

where $p(f|X)$ is the probability distribution of f conditioned on X and has the properties of a normal distribution $\mathcal{N}$ with mean μ and covariance $\mathbb{K}$.

Upon collecting a sample set of data $\Omega = \{(x_{*_i}, f_{*_i})\}$ for $i = 1$ to N samples, where

$f_{*_i} = f_*(x_{*_i})$, are the observations of the sample points evaluated at x_{*_i}. A joint Gaussian is formed to determine interpolated function estimates based on the sampled distributions.

$$\begin{pmatrix} f \\ f_* \end{pmatrix} \sim \mathcal{N}\left(\begin{pmatrix} \mu \\ \mu_* \end{pmatrix}, \begin{pmatrix} \mathbb{K} & \mathbb{K}_* \\ \mathbb{K}_*^T \mathbb{K}_{**} \end{pmatrix}\right) \tag{2.42}$$

Where $\mathbb{K} = \kappa(x, x)$ is $N \times N$, $\mathbb{K}_* = \kappa(x, x_*)$ is $N \times N_*$, and $\mathbb{K}_{**} = \kappa(x_*, x_*)$ is $N_* \times N_*$. The posterior Gaussian Process distribution is defined by 2.43. To make predictions using noise free observations, interpolated points have a mean $\mu(x)$ determined by 2.44 and a posterior covariance kernel Σ_* determined via 2.45.

$$p(f_*|x_*, x, f) = \mathcal{N}(f_*|\mu_*, \Sigma_*) \tag{2.43}$$

$$\mu_* = \mu(x) + \mathbb{K}_* \mathbb{K}_{**}^{-1}(f - \mu) \tag{2.44}$$

$$\Sigma_* = \mathbb{K}_{**} - \mathbb{K}_*^T \mathbb{K}^{-1} \mathbb{K}_* \tag{2.45}$$

Gaussian processes only become problematic if the state space is high dimensional due to the processing required to generate a mean and variance of the function via matrix computations. [109] state that the computation is exponential in the number of dimensions, meaning that practically only 10s of dimensions can be used and there is no way to reduce the exponential computation by reducing dimensional dependencies after declaring them. Although their use has become popular for model-free and hybrid use [21, 55, 103], a number of techniques have been used to reduce the computational load by introducing sparsity of basis features [103] representer points, [55] and state intervals, [21].

2.13 Benchmarking Evaluation of Approaches

To gain some insight into the performance, scaleability and challenges of each learning approach we carry out benchmarking tests on nonlinear systems of up to 10 dimensions to compare safety certificate techniques, function approximating algorithms and control synthesis approaches. We test sparse 4th order polynomials designed to be stable to ensure full time history generation of sample trajectories. We compare safety certificate learning times using a Lyapunov convex optimisation in Yalmip and the SDP solver Mosek compared to the MATLAB ARE algorithm and the CCM technique using SoS polynomials. For both the LQR controller and the Lyapunov optimisation the system is linearised at $x = 0$. Controller synthesis is compared for the CCM technique, LQR technique and NMPC using multiple shooting in CasADi. We also compare function approximation techniques in the form of MATLAB's native Sparse Gaussian Process Regression gprfit and a single layer neural network compared to the nonlinear recursive least squares algorithm SINDYc, [60]. All computations were carried out on a Laptop with a 2.6 GHz Intel Core i7 CPU and 16GB of RAM.

The 2D nonlinear system dynamics tested is given in 2.46. The systems are used to generate trajectory data via an ode45 solver over a 20 second time interval at 1000Hz, (2.46). The trajectory data is then sampled at varying time intervals for learning the function approximations.

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} -1.47x_1 - 0.574x_2 - 0.2x_1x_2 \\ -0.760x_2 - x_2^4 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u \tag{2.46}$$

2.13.1 Function Approximations

This section details the function approximations used for benchmarking. All three techniques have a Gaussian noise applied to the nominal trajectory data with a Sigma of 0.005. However the RLS approach was particulary sensitive to the addition of noise and to compare processing times noise was removed to allow the algorithm to converge. In generat-

ing trajectories with learned coefficients, if the coefficients make the system unstable this was a particular problem in generating prediction trajectories.

SINDYc Recursive Least Squares

The SINDYc algorithm [29], uses the L_1 norm recursive least squares regression of the function approximation from a library of state features evolving in the time vector. The library must be pre-determined before the regression. In the case of the polynomial regression, the basis features are chosen as monomials of up to degree 4 in the state dimensions as well as the control input. The baseline code relies on a forcing function to generate forced trajectories. However for this simple study an unforced trajectory was sufficient to make accurate predictions. Parameters must be chosen ahead of time for the sampling frequency, 100Hz, polynomial degree and number of system dimensions.

Sparse Gaussian Process Regression

The native Matlab Sparse Gaussian Process Regression function 'fitrgp' is used to process the training data in batches. The only settings varied for the study were the number of observation points of 20, 50 and 20000 points, by sampling the training data over varying intervals; and the batch sizes used for the regression as an active set of 10 and 50 points. The Sparse Greedy Matrix Approximation is used for the Active Set method and Fully Independant Conditional (FIC) where individual function dimensions are regressed separately as independant functions. A Sigma and lengthscale of 1 is used for the squared exponential function kernel.

Single Hidden Layer Neural Networks

For the single layer neural network the Matlab 'network' function is used with Levenberg-Marquardt backpropagation as the training algorithm. The number of hidden layers is varied to test the sensitivity of the network accuracy using, 3,5,6,8 and 16 nodes as the polynomial degree in the training data is a maximum degree 4 polynomial and a 2D

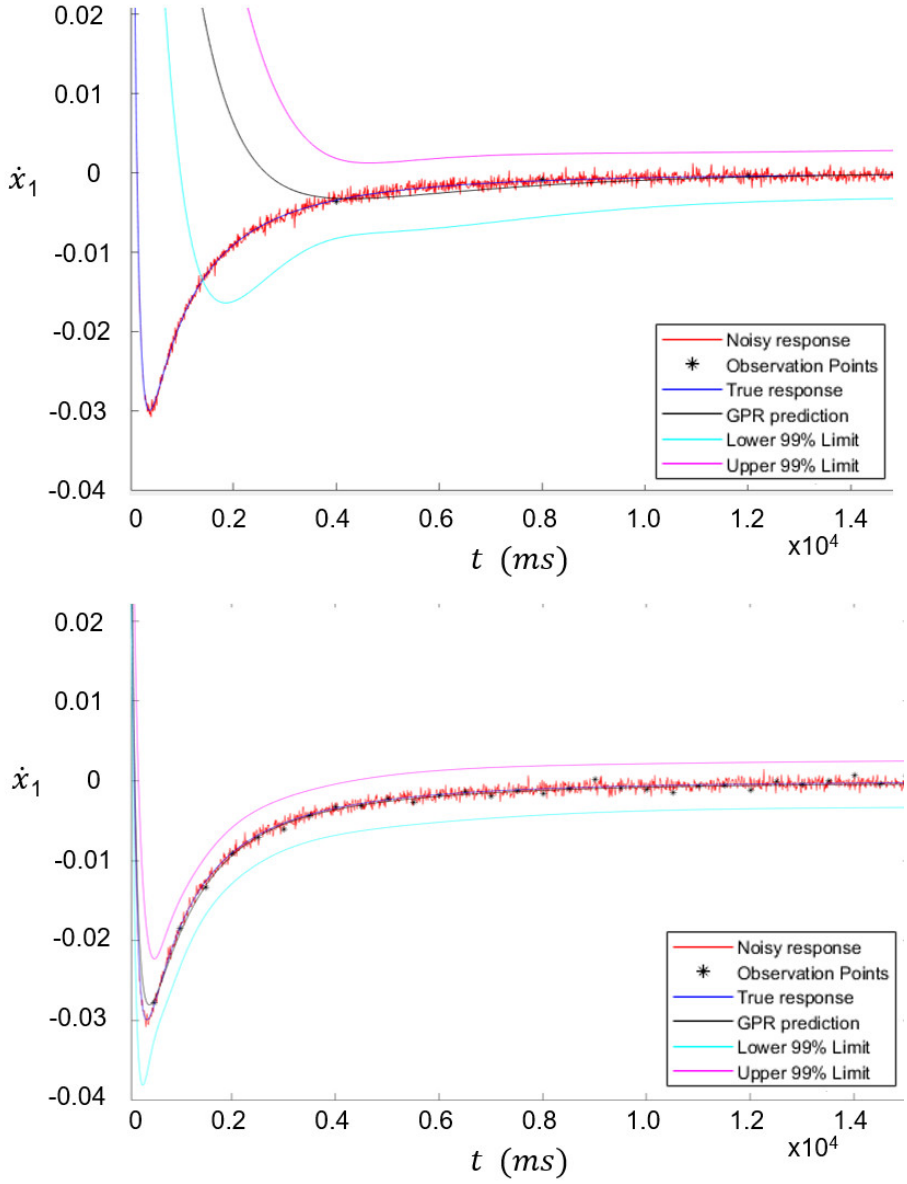


Figure 2.24 – Sparse Gaussian Process Regression with 50 Observations Top and 400 Observations Bottom

system therefore requires a maximum of 2^4 hidden nodes.

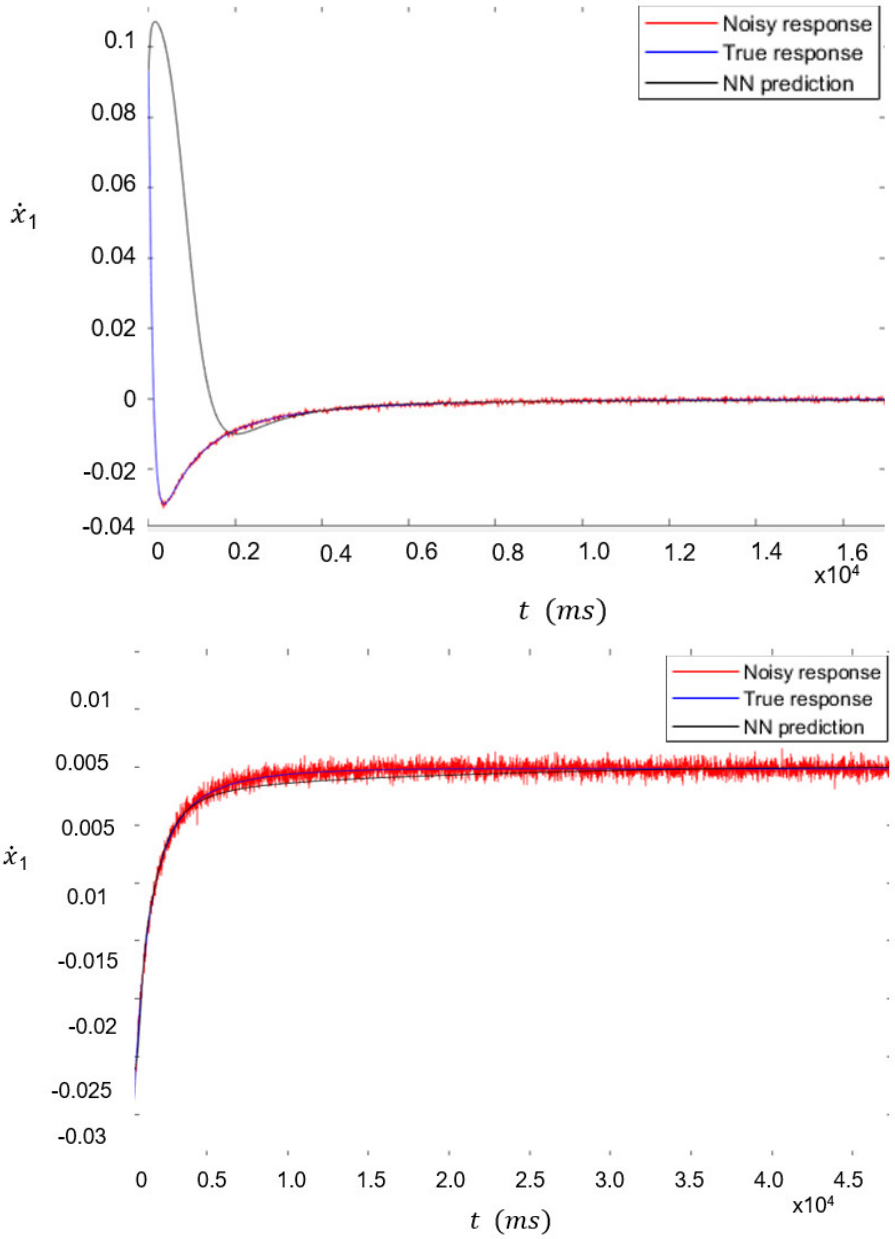


Figure 2.25 – Neural Network with 2 Hidden Nodes Trained on 100 data points Top and 1000 data points Bottom

2.13.2 Safety Certificates

This section compares safety certificate learning times using a Lyapunov convex optimisation in Yalmip and the SDP solver Mosek compared to the MATLAB ARE algorithm and the CCM technique using SoS polynomials. For both the LQR controller and the

Lyapunov optimisation the system is linearised at $x = 0$.

Linearised Lyapunov Convex Optimisation

For the optimisation using linearised feedback control the system equation is linearised around the origin. Linearisation is not possible for some systems, as described in [67], which restricts the technique to certain classes of functions. The linearised function is a nonconvex problem, 2.5, where a congruence transformation can be used as described in Section 2.4.1 to form a convex problem.

Yalmip is used with the Mosek SDP solver to find the optimal P matrix as opposed to using the standard ARE formulation.

2.13.3 Controller Synthesis

This section describes the controller synthesis benchmark test for the NMPC technique, LQR technique and CCM Chebyshev Pseudospectral technique.

NMPC in Casadi

With reference to the study in [67], we chose 100 segments as the tradeoff between speed and accuracy with a 10s time horizon for the multiple shooting algorithm in CasAdi toolbox [11] via nonlinear programming, interior point method, IPOPT solver [82]. For the 2D system the steady state error is in the order of 0.01 whereas at higher dimensions the errors increase by orders of magnitude, 2.15. Even at these error scales the solution times to generate a controller were between 6 and 14s.

CCM Geodesic Controller Synthesis

We employ the same Chebychev Pseudospectral technique used in [67], to optimise the controller from any given displacement in the state space. The technique adopts Chebshev polynomials as basis functions as opposed to Lagrange polynomials for speed and

Table 2.15 NMPC Synthesis Times and Errors

Dim	X_0	Synth Time (s)	SS Error
2	1,1	11.71	0.097
2	9,9	10.38	0.098
4	1,1,1,1	9.5	0.727
4	9,9,9,9	8.87	0.721
6	1,1,..1	8.67	0.168
6	9,9,..9	11.66	1.386
8	1,1,..1	10.8	0.371
8	9,9,..9	13.67	10.2

accuracy improvements. The caveat to the technique is that the polynomial degree and number of nodes used must be optimised ahead of time, either via offline optimisation or from expert knowledge. In this study the a value is fixed at 4, where the number of nodes, $N = deg + a$. The values for the quasi-Newton descent algorithm are given in Table 2.16.

Table 2.16 Chebyshev Pseudospectral Newton Descent Hyperparameters

Parameter	value	Parameter	value
a	4	deg	4
N	4	alph0	1
c	0.1	tau	0.1
sparse eps	$1e^{-20}$	rel tol	$1e^{-7}$
repetition	100	-	-

LQR Control Synthesis

LQR control synthesis was carried out using the standard Matlab ARE routines to generate a P matrix from a linearised function matrix equation $\dot{x} = Ax + Bu$ and Q and R matrices from the LQR cost function where both Q and R are set to unity.

2.13.4 Results of Benchmarking Tests

In this section we compare the results of the benchmark tests for safety certificate learning times, controller syntheis times and function approximation times for each technique.

Safety Certificate Times

Fig. 2.26 shows the computation times for systems of up to 8 dimensions, for all nonlinear controller safety certificates compared to a neural network grid search approximation. The DSoS, SDSoS and SoS solutions using the SPOTless algorithm are not directly comparable to the Yalmip implementation of the Lyapunov and CCM SoS techniques due to not being able to use an optimizer and timeit function to enable embedded timings without overhead processing times. Comparing the SPOTless SDSoS, DSoS and SoS algorithms the SDSoS scales slightly better than the SoS implementation but the DSoS algorithm is slightly slower at 6 and 8 dimensions. The authors in [73] find however that the merits of using DSOS and SDSOS techniques are most beneficial above 30 dimensions. Fig. 2.27 just shows the CCM, LQR and Lyapunov Convex Optimisation techniques. The CCM solution using SoS is exponential but provides a nonlinear solution globally whereas the LQR solution is the fastest linear approach using the ARE compared to the Lyapunov technique in Yalmip/Mosek via convex optimisation. Although the Lyapunov Convex Optimisation is only slightly slower than LQR it does allow for additional optimisation constraints.

Controller Synthesis Times

Fig.2.28 shows the results for the synthesis times for the LQR technique, NMPC technique and the CCM geodesic calculation using the Chebyshev Pseudospectral technique. Clearly the LQR approach is the faster approach as it exploits the use of simple linear algebraic techniques and matrix multiplications with a fixed response time of just less than 10^{-5} seconds. The CCM technique is the middle ground using Pseudospectral collocation with a response time of between 10^{-1} and 10^{-2} for dimensions up to 6D. The 8D

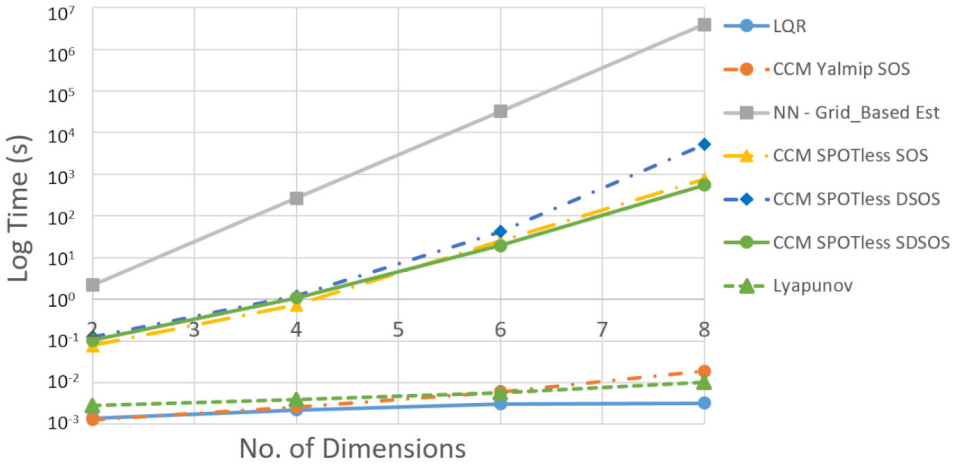


Figure 2.26 – Safety Certificate Learning Times Comparison

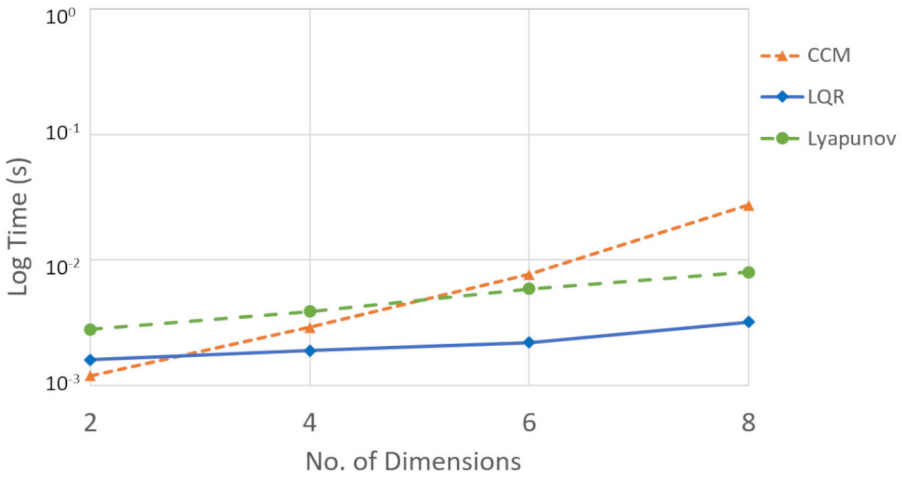


Figure 2.27 – Safety Cerificate Learning Times - Focusing on the LQR, Lyapunov and CCM Curves

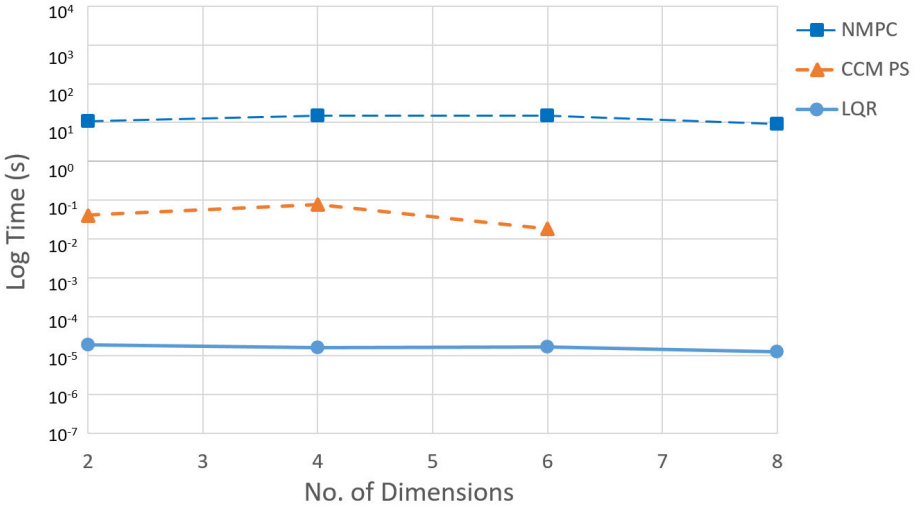


Figure 2.28 – Controller Synthesis Times

result for the CCM technique is omitted due to the high number of polynomial coefficents required to produce a Metric which would have required further automation of the coding using to generate the Metric. The NMPC technique using 100 segments was consistently just slower than 10^1 seconds for all dimensions tested. It is worth noting that the CCM response times have the most variation in the times for each dimension tested which may be less desirable in terms of reliability. This is also related to the fact that the polynomial degree chosen to find a geodesic is dependant upon the shape of the manifold and the computation times involve more complexity than simple LQR calculations.

Function Approximation Times

Fig. 2.29 shows the solution times for a single layer neural network with 3,5,6,8 and 16 hidden nodes compared to a sparse GPR fit using batch sampling of 10 and 50 samples and regression points numbering 20, 50 and 20,000 (abbreviated 2e1r,5e1r and 2e4r respectively). These are compared to a RLS solution using the SINDYc algorithm. It can be seen that the SGPR training times are fairly smoothly increasing as an exponential in the dimension whereas the NN times vary more indeterminately for a given number of hidden nodes with the dimension. This may be because NNs in general learn via more

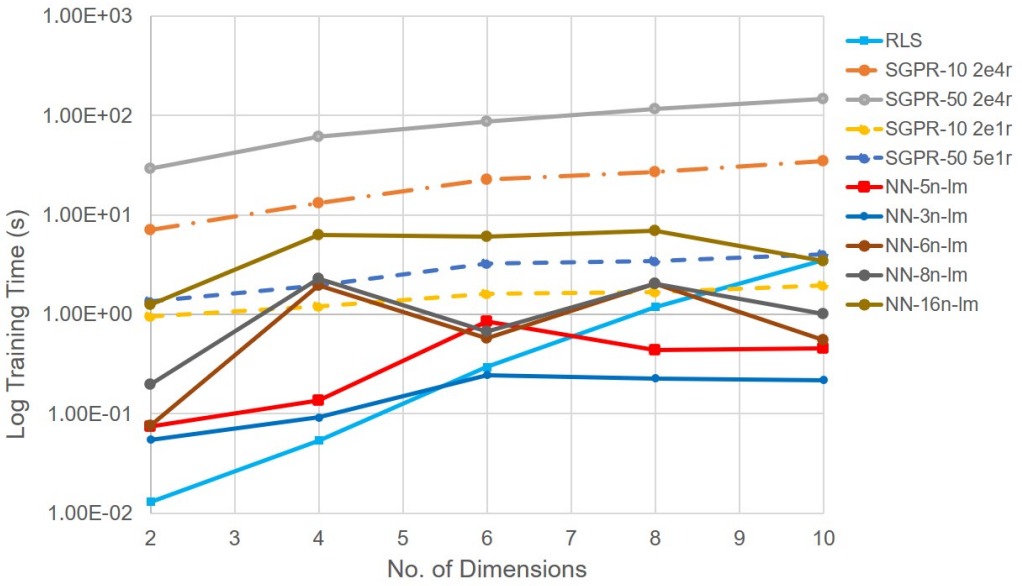


Figure 2.29 – Function Approximation Learning Times

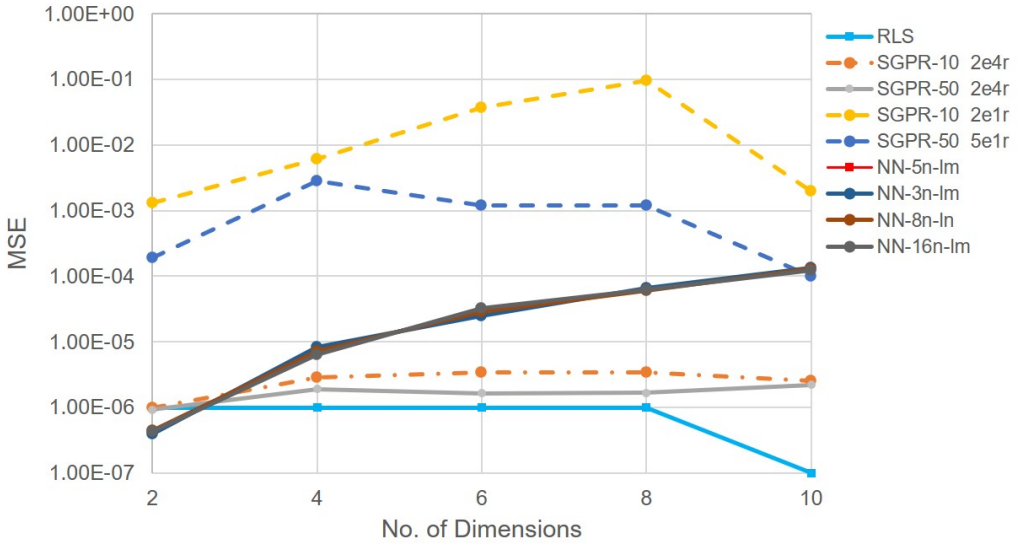


Figure 2.30 – Function Approximation Mean Squared Errors

complex non smooth manifolds that may vary more indeterminately depending upon the combinations of basis features present in the nodal structure paired with the basis present in the dimensionality of the system. It must be noted that due to the nonlinear systems being handcrafted with a unique sparse basis set this trend may average out over a more comprehensive random basis selection study. In contrast to the relatively flat NN and SGPR curves the RLS has a steeper exponential learning time in the dimensionality of the systems which offsets its advantage of being the faster algorithm at 2 dimensions. Comapring the NNs and the SPGRs we see that the 20,and 50 inference point regressions are able to learn the functions as fast as the 10 dimensional RLS algorithm and approximately the NNs above 6 hidden nodes. Fig. 2.30 compares the MSEs of each function approximation. The NNs are all tightly overlaid showing that they are accurate independant of the hidden node structures. This may highlight the fact that the systems are not complex enough to show the full expressiveness of the polynomial degree. The MSE is also slowly rising for the NN whereas for the RLS and SGPRs the errors are constant for the 20000 inference point approximations whereas the 20 and 50 inference point solutions are at least 3 orders of magnitude less accurate and less consistent. This shows how the accuracy of the function approximation is sensitive to the number of inference points and likely the smoothness of the function and number of turning points. Clearly with increasing numbers of inference points the accuracy increases although the optimum number may require iterative approaches to analyse convergence properties. It must be pointed out that although the RLS algorithm has the best accuracy the basis library was chosen to have the polynomial basis present in the system and so in real world dynamics this may not be the case.

2.14 Comparison of Techniques

In this section we summarise some of the benefits and limitations for each of the techniques discussed.

Table 2.17 highlights some of the key advantages and disadvantages of the main approaches. A summary of salient techniques is given in Table 2.18 with approximations of the safety certification and response times for systems of approximately between 3 to 6 dimensions. Table 2.19 highlights a qualitative comparison of the key metrics discussed for the main approaches presented.

Table 2.17 Overview of Advantages and Disadvantages

Technique	**Advantages**	**Disadvantages**
HJI + GPs	Explicit global safety certificate Robust to worst case disturbances Least restrictive high probability guarantees	Strictly offline safety certificate Requires prior probability assumptions Limited in dimensionality $<\sim 6D$
LB-NMPC	Implicit Safety certificate with dynamic constraints Robust to worst case disturbances Provides accurate real time planning	Online computation is typically too expensive Accuracy limited by time horizon and time step Not suitable for stiff systems Requires an accurate model Requires tuning for optimal control
LB-RCCM	Explicit local or global SC, Decomposable, modular safety certificate Can handle constraints in real time Faster online control synthesis vs NMPC	Conservatism bound to FA Complex mathematical theory SoS coefficients exponential in polynomial degree Most problems are bi-convex
UA-BO	Incremental online safety certificate High probability guarantees on safety certificate Exploration driven by uncertainty	Limited in dimensionality $<\sim 6D$ Relies on prior probability estimates Assumes smooth function approximation
LQR-Trees	Stable Path planning safety certificate Robust to worst case disturbances Online multi-step trajectory optimisation	Strictly offline library based technique Relies on linearised local trajectory dynamics More for obstacle avoidance and path planning
Robust Linear CPA	Explicit global safety certificate Robust to worst case disturbances Fast online controller synthesis	Relies on local linear approximations Once setup cannot adapt dynamics online easily
LB-CLFs	Explicit global safety certificate Robust to worst case disturbances Flexible function approximation	Limited by function classes for direct solution Function classes are typically conservative SoS techniques are exponential in polynomial degree
LB-MPC + GPs	Implicit local safety certificate Robust to uncertainty Provides uncertainty bound on safety	Requires a nominal linear model GP functions limited in dimensionality Requires estimates for hyperparameters of GP

HJI techniques are really offline techniques for nonlinear problems especially above 4 or 5 dimensions where processing times are in the 10^4 seconds. They provide one of the most accurate and tractable solutions through the use of barrier functions and provide

Table 2.18 Summary of Learning and Response Times for Salient Techniques

Attributes/Criteria	**HJI**	**CPA**	**CCM**	**NMPC**	**MPC+SGPs**	**SBO**
Learning Time (s)	10^4	10^4	10^1	-	10^1	10^1
Response Time (s)	10^{-2}	10^{-3}	10^{-2}	10^{-1}	10^{-2}	10^{-1}

* Times are approximate for systems of between 3-6 dimensions on a standard desktop PC.

Table 2.19 Summary of Techniques

Attributes/Criteria	**HJI**	**CPA**	**CCM**	**NMPC**	**MPC+SGPs**	**SBO**
Scaleability	•	••••	•••	••	••	••
Reliability	••••	•••••	•••	••	••	••
Conservatism	•••••	••	••••	•••••	••	••••
Learning Time	•	•	•••	••••	••••	•••
Response Time	••••	••••	••••	•	•••	•••
Explainability	••••	••••	•••	••••	••	••

a useful benchmark for the research of other online techniques. If the model is accurate HJI techniques would provide a reliable library of safe states as long as the resolution of the grid is sufficient with only uncertainty to handle in real time. The expressive PDE approach would potentially be non-conservative but there is no guarantee that a PDE accurate for the system can be formed as a tractable Hamiltonian equation. Response times are only limited by the speed of the library lookup and reachability techniques with game theory are tractable analytical techniques providing transparency and explainability.

CPA techniques adopting QP solvers have recently been optimised for embedded applications [36] with sub-millisecond solver times. For nonlinear systems however, multivariable semi-infinite optimisation tasks are a significant challenge. The CPA techniques are generally scaleable as they are offline libraries but limited by memory constraints. Linear techniques are the most reliable in terms of producing a solution for a policy the problem arises if the model is not accurate enough to represent the nonlinearities in the

system for this reason we would rank reliability high but conservatism low. The learning time is also ranked lowest due to the offline library generation. Response times are again only limited by lookup times and explainability is high because of the well established linear theory.

NMPC offers attractive performance and safety verification properties by validating trajectories online but at the expense of slower response times unsuitable for highly dynamic, high dimensional or stiff systems. They also require tuning typically of the hyperparameters to determine the most accurate controllers online. For this reason we rank NMPC as less reliable than other techniques due to the tradeoff of model accuracy versus real time computational constraints. In practice optimising hyperparameters in real time is not be possible and response times in the order of 10^1s not practical for real-time systems. If the computational challenges are met then NMPC would offer a non-conservative solution assuming that the model was accurate. Learning times and explainability are only constrained by the function approximation used and so we also rank NMPC highly for these metrics.

By adopting a linear differential analysis of control systems dynamics in the form of CCMs, a broader class of systems can be characterised in reasonable online learning time frames via a pointwise LMI over a grid or via bi-convex optimisation and line searching, [103]. Learning time frames in the order of $10^0 s$ to $10^2 s$ with controller synthesis response times in the $10^{-3} s$ to $10^{-}1 s$ time range. In terms of scaleability CCM techniques are a middle ground between linear techniques and nonlinear MPC solutions. A few drawbacks to CCM techniques are that SoS techniques create an exponential increase in coefficients for increasing dimensionality and polynomial degree and in general research has shown that some dynamical systems do not admit polynomial Lyapunov functions of any degree [32]. CCM control synthesis is also a non-trivial optimisation and requires a higher level of abstraction and mathematical understanding of the problem formulation using topological vector spaces.

Uncertainty aware based techniques provide probability guarantees for exploration but

limit the scaleability of algorithms using GP techniques and also rely on an understanding of the model priors. SAL is an active area of research for the trade off of exploration and explotation of the policy function. Uncertainty will always increase the computation of any technique by introducing additional parameters and hyperparameters and so expert judgement must be used to maximise performance.

Hybrid approaches using a nominal model and a sparse GP learning function have been the most successsful in recent research for both online optimisation and uncertainty handling. Sparse Gaussian process techniques are proving to be competitive when combined with fast linear MPC or SQP implementations of nominal model behaviour with response times and online learning reported in sub 20ms times. The limitation for this approach is the expressiveness of a sparse GP limited to 10 inducing points may not be expressive enough for more challenging function approximations so we would rank conservatism lower than other techniques as well as scaleability to meet the real time response times. Explainability is a middle ground because hybrid approaches are tractable models combined with model free approximations and even though GPs are well understood they are still dependent upon the choice of lengthscale, uncertainty and inducing points to accurately represent the residual dynamics.

In the BO approach, GPs are the limiting process in terms of scaleability. For reliability SAL is still an active area of research but fundamentally a nonlinear programming approach which is less reliable than convex optimisation where a global optimum is guaranteed. Learning time is limited by the dimensionality of the system and the discretisation of the search space to validate a safety certificate and explainability is ranked higher than a fully hybrid approach. In theory GPs are nonconservative but the assumption and hyperparameters limit the expressiveness to certain classes of smooth functions.

In safety critical online learning, it is not only necessary to produce solutions in fast time frames but also to have guarantees on the quality of the solutions. For this reason model-based control theory may be better suited to the task than sampling based, model-free or neural network approaches that do not lend themselves to analytical proofs or

reliable convergence properties arising from numerous localised minima. That said the speed of NNs and sampling based techniques makes them attractive for research and practical implementations.

Hybrid techniques using model-based and model-free learning are a promising future research direction. Parallelisation and decomposition of systems offer engineers opportunities to overcome the challenges of state space explosion.

2.15 Gaps and Challenges

There is currently much research into the efficient calcualtion of safety certificates for higher dimensional systems. Specifically the use of distributed optimisation algorithms like the Alternating Direction Method of Multipliers (ADMM) [103], sparse representation of function classes as used in the random Fourier basis technique, [95] or the sparse SoS techniques, [73], however the fundamental challenge of searching an optimum feasible solution in high dimensions remains. This poses a problem for real time programming due to the size of the solution spaces generated from state space explosion. There may be smarter ways to address these challenges by localising searches and exploiting domain knowledge to prune the hypothesis space, [112]. Further the isolation of system dependecies and decomposition of large scale problems to one of a more manageable size has been investigated by many researchers, [53].

A fundamental challenge in safe learning is the treatment of uncertainty in the certification process. Currently assumptions must be made based on prior estimates of uncertainty in order to predict future behaviour, [22]. This forms the basis of robust control techniques where a threshold is projected around a certified policy but can only be updated based on collected data, [43]. The other challenge related to robust control is that of too conservative a policy if the threshold for safety is the worst case disturbance for example it means that the system is always operating at the worst performance attributes to maintain the strict guarantees, [119]. Research has addressed these limitations however

the treatment of uncertainty will remain some what of an art and heuristic by nature of the problem.

Neural network approaches to learning remain an attractive general function approximation tool for high dimensional regression problems, [96] and as the number of dependencies increase NN architectures are attractive for performance reasons but the non-deterministic nature of the input and output mappings means that some doubt will always remain over their role in safety critical applications. Linear frameworks that adopt non-linear basis may be the best middle ground for generalism of function classes and several recent studies use these approaches in the form of RKHS [23], GPs [55], RLS techniques, [29], but they still require a choice of discrete basis either in the selection of inducing points or in the choice of a library of features. The challenge of choosing these basis remains, to effectively capture the function class being sampled and to remain computationally efficient in the generation of the model.

We have proposed that the reliability of solutions be the most important characteristic in generating polices online from data but the reality of nonlinear solutions is that they are often non-convex problems which poses a fundamental challenge in guarenteeing that a solution can be found accurately for every iteration of the algorithm. Research has investigated the non-convex problems in the form of Bilinear techniques, sampling techniques or gradient based techniques but for safety critical applications a more robust solution may be required for long term deployment. The trade off of reliability with the performance of function classes is an interesting area of investigation.

2.16 Future Work

Based on our survey of approaches we have identified two of the most promising techniques for extension to online learning problems. Namely the CCM technique and hybrid MPC techniques with residual dynamics learning. These two approaches have gained the most interest and success in real world testing and warrant further investigation. We also

plan to address the following key challenges:

1. In order to build architectures for higher dimensional systems the limitations in scaleability owing to the choice of optimisation technique and function classes will be investigated. The growing interest into Sparse Gaussian Process techniques and Random Fourier Basis is of interest for purely data driven learning and greater generalism whilst also providing tractable and explainable training processes. Convex optimisation is also a growing area of interest for high reliability and speed of processing high dimensional problems where new techniques to handle multivariable problems offer interesting avenues for research.
2. Model reduction, parallelisation or other decomposition techniques are also of interest for the development of efficient learning algorithms that can address the issues of high dimensionality in the generation of safety certificates. By breaking down problems into efficient decision chains to optimise online processing, efficient model exploration will be investigated.
3. Function approximations will be further investigated for sensitivity to system noise and the model-gap between nominal and off-nominal system behaviour considered with respect to the choice of function classes.
4. The reliability of any approaches researched must be compared against engineering standards for certification. The certification of techniques may need to be considered prior to designing a system to reduce the number of infeasible architectures considered.
5. The challenge of developing techniques that are easily implemented in a modular architecture for integration with established engineering practices is also of interest. It is thought that adoption of new techniques will be an incremental process and require layers of oversight and tractable testing procedures where modularity will play an important role.
6. Further the development of algorithm performance metrics and system performance

metrics to compare techniques quantitatively will allow more in depth analysis of different approaches.

2.17 Conclusion

In this review we have compared state-of-the-art safe online learning algorithms for online nonlinear dynamic systems. We highlight some of the challenges in terms of performance and reliability and the most promising developments and future directions have been discussed. This paper is intended to provide a useful introduction to the subject, an overview of the current avenues of research and the potential challenges posed by each. The research is aimed to help progress autonomous real-time learning for nonlinear dynamic control systems.

References

[1] M. Abdolhosseini, Y. M. Zhang, and C. A. Rabbath. An efficient model predictive control scheme for an unmanned quadrotor helicopter. *J. Intell. Robot. Syst. Theory Appl.*, 70(1-4):27–38, 2013.

[2] A. A. Ahmadi, A. Chaudhry, V. Sindhwani, and S. Tu. Safely learning dynamical systems from short trajectories. pages 498–509, 2021.

[3] A. A. Ahmadi and A. Majumdar. Some applications of polynomial optimization in operations research and real-time decision making. *Optim. Lett.*, 10(4):709–729, 2016.

[4] A. A. Ahmadi and A. Majumdar. DSOS and SDSOS optimization: More tractable alternatives to sum of squares and semidefinite optimization. *arXiv*, 3(2):193–230, 2017.

[5] A. A. Ahmadi and A. Majumdar. DSOS and SDSOS Optimization: More Tractable Alternatives to Sum of Squares and Semidefinite Optimization. *SIAM*, 3(2):193–230, 2019.

[6] Air Force Times. F35 high angle of attack, 2019. [Online; accessed April 16, 2021].

[7] A. K. Akametalu, S. Kaynama, J. F. Fisac, M. N. Zeilinger, J. H. Gillula, and C. J. Tomlin. Reachability-based safe learning with gaussian processes. *Proceedings of the IEEE Conference on Decision and Control*, 2015-Febru(February):1424–1431, 2014.

[8] M. D. Aliyu. Iterative computational approach to the solution of the Hamilton-Jacobi-Bellman-Isaacs equation in nonlinear optimal control. *Control Theory Technol.*, 16(1):38–48, 2018.

[9] M. Althoff, O. Stursberg, and M. Buss. Reachability analysis of nonlinear systems with uncertain parameters using conservative linearization. *Proceedings of the IEEE Conference on Decision and Control*, pages 4042–4048, 2008.

[10] A. D. Ames, S. Coogan, M. Egerstedt, G. Notomista, K. Sreenath, and P. Tabuada. Control barrier functions: Theory and applications. *2019 18th European Control Conference, ECC 2019*, pages 3420–3431, 2019.

[11] J. A. Andersson, J. Gillis, G. Horn, J. B. Rawlings, and M. Diehl. CasADi: a software framework for nonlinear optimization and optimal control. *Math. Program. Comput.*, 11(1), 2019.

[12] M. Aps. Mosek modeling cookbook, 2020.

[13] N. Aronszajn. Theory of reproducing kernels. *Transactions of the American Mathematical Society*, 68(3):337–404, 2009.

[14] A. B. Arrieta, N. Díaz-Rodríguez, J. Del Ser, A. Bennetot, S. Tabik, A. Barbado, S. García, S. Gil-López, D. Molina, R. Benjamins, et al. Explainable artificial intelligence (xai): Concepts, taxonomies, opportunities and challenges toward responsible ai. *Information fusion*, 58:82–115, 2020.

[15] E. M. Aylward, P. A. Parrilo, and J.-J. E. Slotine. Stability and robustness analysis of nonlinear systems via contraction metrics and sos. *Automatica*, 44(8):2163–2170, 2008.

[16] A. Ayyad, M. Chehadeh, M. I. Awad, and Y. Zweiri. Real-time system identification using deep learning for linear processes with application to unmanned aerial vehicles. *IEEE Access*, 8:122539–122553, 2020.

[17] S. Bansal, R. Calandra, T. Xiao, S. Levine, and C. J. Tomiin. Goal-driven dynamics learning via Bayesian optimization. *2017 IEEE 56th Annu. Conf. Decis. Control. CDC 2017*, 2018-Janua(1545126):5168–5173, 2018.

[18] S. Bansal, M. Chen, S. Herbert, and C. J. Tomlin. Hamilton-jacobi reachability: A brief overview and recent advances. *2017 IEEE 56th Annual Conference on Decision and Control, CDC 2017*, 2018-January:2242–2253, 2018.

[19] A. Bemporad, M. Morari, V. Dua, and E. N. Pistikopoulos. The explicit linear quadratic regulator for constrained systems. *Automatica*, 38(1):3–20, 2002.

[20] F. Berkenkamp, A. Krause, and A. P. Schoellig. Bayesian optimization with safety constraints: safe and automatic parameter tuning in robotics. *arXiv preprint arXiv:1602.04450*, 2016.

[21] F. Berkenkamp, R. Moriconi, A. P. Schoellig, and A. Krause. Safe learning of regions of attraction for uncertain, nonlinear systems with gaussian processes. *2016 IEEE 55th Conference on Decision and Control (CDC)*, pages 4661–4666, 2016.

[22] F. Berkenkamp and A. P. Schoellig. Safe and robust learning control with gaussian processes. *2015 European Control Conference, ECC 2015*, pages 2496–2501, 2015.

[23] F. Berkenkamp, M. Turchetta, A. P. Schoellig, and A. Krause. Safe model-based reinforcement learning with stability guarantees. *Advances in Neural Information Processing Systems*, 2017-Decem(Nips):909–919, 2017.

[24] R. Bobiti and M. Lazar. Automated-sampling-based stability verification and doa estimation for nonlinear systems. *IEEE Transactions on Automatic Control*, 63(11):3659–3674, 2018.

[25] S. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan. *Linear matrix inequalities in system and control theory*, volume 15. SIAM, 1994.

[26] S. Boyd and L. Vandenberghe. *Convex optimization*. Cambridge University Press, 2004.

[27] M. Branicky and M. Curtiss. Nonlinear and hybrid control via RRTs. *Proc. Intl. Symp. Math. Theory Networks . . .*, pages 1–10, 2002.

[28] E. Brochu, V. M. Cora, and N. de Freitas. A tutorial on bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning. *arXiv*, 2010.

[29] S. L. Brunton, J. Nathan Kutz, K. Manohar, A. Y. Aravkin, K. Morgansen, J. Klemisch, N. Goebel, J. Buttrick, J. Poskin, A. Blom-Schieber, T. Hogan, and D. McDonald. Data-driven aerospace engineering: Reframing the industry with machine learning. *arXiv*, pages 1–35, 2020.

[30] S. Bubeck et al. Convex optimization: Algorithms and complexity. *Foundations and Trends® in Machine Learning*, 8(3-4):231–357, 2015.

[31] T. L. Chaffey and I. R. Manchester. Control Contraction Metrics on Finsler Manifolds. *Proc. Am. Control Conf.*, 2018-June:3626–3633, 2018.

[32] Y.-C. Chang, N. Roohi, and S. Gao. Neural lyapunov control. *Advances in neural information processing systems*, 32, 2019.

[33] M. Chen, S. L. Herbert, M. S. Vashishtha, S. Bansal, and C. J. Tomlin. Decomposition of Reachable Sets and Tubes for a Class of Nonlinear Systems. *IEEE Trans. Automat. Contr.*, 63(11):3675–3688, 2018.

[34] X. Chen, S. Chen, and V. M. Preciado. Safety Verification of Nonlinear Polynomial System via Occupation Measures. (Cdc):1159–1164, 2019.

[35] R. Cheng, G. Orosz, R. M. Murray, and J. W. Burdick. End-to-end safe reinforcement learning through barrier functions for safety-critical continuous control tasks. *Proceedings of the AAAI Conference on Artificial Intelligence*, 33(01):3387–3395, Jul. 2019.

[36] G. Cimini, D. Bernardini, S. Levijoki, and A. Bemporad. Embedded model predictive control with certified real-time optimization for synchronous motors. *IEEE Trans. Control Syst. Technol.*, pages 1–8, 2020.

[37] W. B. Cotton, R. Hilb, S. Koczo, and D. J. Wing. A vision and roadmap for increasing user autonomy in flight operations in the national airspace. In *16th AIAA Aviation Technology, Integration, and Operations Conference*, page 4212, 2016.

[38] K. Dalamagkidis, K. P. Valavanis, and L. A. Piegl. Nonlinear model predictive control with neural network optimization for autonomous autorotation of small unmanned helicopters. *IEEE Trans. Control Syst. Technol.*, 19(4):818–831, 2011.

[39] J. Darbon and S. Osher. Algorithms for overcoming the curse of dimensionality for certain hamilton–jacobi equations arising in control theory and elsewhere. *Research in Mathematical Sciences*, 3(1), 2016.

[40] Electrive. Lillium vtol air taxi, 2020. [Online; accessed April 26, 2021].

[41] S. Fedorov and A. Candelieri. Reachability-based safe learning for optimal control problem. pages 1–13, 2018.

[42] C. Finn, S. Levine, and P. Abbeel. Guided cost learning: Deep inverse optimal control via policy optimization. *33rd International Conference on Machine Learning, ICML 2016*, 1:95–107, 2016.

[43] J. Fisac, A. Akametalu, M. Zeilinger, S. Kaynama, J. Gillula, and C. Tomlin. A general safety framework for learning-based control in uncertain robotic systems. *IEEE Transactions on Automatic Control*, 64(7):2737–2752, 2019.

[44] M. Fisher, V. Mascardi, K. Y. Rozier, B.-H. Schlingloff, M. Winikoff, and N. Yorke-Smith. Towards a framework for certification of reliable autonomous systems. *Autonomous Agents and Multi-Agent Systems*, 35(1):1–65, 2021.

[45] F. Forni and R. Sepulchre. A differential lyapunov framework for contraction analysis. *IEEE Trans. Automat. Contr.*, 59(3):614–628, 2014.

[46] N. Fulton and A. Platzer. Safe reinforcement learning via formal methods: Toward safe control through proof and learning. *32nd AAAI Conference on Artificial Intelligence, AAAI 2018*, pages 6485–6492, 2018.

[47] M. Ghavamzadeh, S. Mannor, J. Pineau, A. Tamar, et al. Bayesian reinforcement learning: A survey. *Foundations and Trends® in Machine Learning*, 8(5-6):359–483, 2015.

[48] J. H. Gillula, G. M. Hoffmann, H. Huang, M. P. Vitus, and C. J. Tomlin. Applications of hybrid reachability analysis to robotic aerial vehicles. *The International Journal of Robotics Research*, 30(3):335–354, 2011.

[49] R. Grandia, A. Taylor, A. Singletary, M. Hutter, and A. Ames. Nonlinear model predictive control of robotic systems with control lyapunov functions. *Robotics: Science and Systems XVI*, Jul 2020.

[50] S. Gros, R. Quirynen, and M. Diehl. Aircraft control based on fast non-linear mpc & multiple-shooting. pages 1142–1147, 2012.

[51] W. Gu, K. P. Valavanis, M. J. Rutherford, and A. Rizzo. A Survey of Artificial Neural Networks with Model-based Control Techniques for Flight Control of Unmanned Aerial Vehicles. In *2019 Int. Conf. Unmanned Aircr. Syst.*, pages 362–371. IEEE, 2019.

[52] J. J. Harris and J. R. Stanford. F-35 flight control law design, development and verification. *2018 Aviat. Technol. Integr. Oper. Conf.*, pages 1–18, 2018.

[53] S. L. Herbert, M. Chen, S. Han, S. Bansal, J. F. Fisac, and C. J. Tomlin. FaSTrack: A modular framework for fast and guaranteed safe motion planning. In *2017 IEEE*

56th Annu. Conf. Decis. Control. CDC 2017, volume 2018-Janua, pages 1517–1522, 2018.

[54] S. L. Herbert, S. Ghosh, S. Bansal, and C. J. Tomlin. Reachability-based safety guarantees using efficient initializations. *arXiv*, 2019.

[55] L. Hewing, J. Kabzan, and M. N. Zeilinger. Cautious model predictive control using gaussian process regression. *IEEE Transactions on Control Systems Technology*, 28(6):2736–2743, 2020.

[56] L. Hewing, K. P. Wabersich, M. Menner, and M. N. Zeilinger. Learning-based model predictive control: Toward safe learning in control. *Annu. Rev. Control. Robot. Auton. Syst.*, 3(1):269–296, 2020.

[57] S. Heyer, D. Kroezen, and E.-J. Van Kampen. Online adaptive incremental reinforcement learning flight control for a cs-25 class aircraft. In *AIAA Scitech 2020 Forum*, page 1844, 2020.

[58] X. Huang, D. Kroening, W. Ruan, J. Sharp, Y. Sun, E. Thamo, M. Wu, and X. Yi. A survey of safety and trustworthiness of deep neural networks: Verification, testing, adversarial attack and defence, and interpretability. *Computer Science Review*, 37:100270, 2020.

[59] G. Kahn, A. Villaflor, V. Pong, P. Abbeel, and S. Levine. Uncertainty-aware reinforcement learning for collision avoidance. *arXiv e-prints*, 2017.

[60] E. Kaiser, J. N. Kutz, and S. L. Brunton. Sparse identification of nonlinear dynamics for model predictive control in the low-data limit. *Proc. R. Soc. A Math. Phys. Eng. Sci.*, 474(2219), 2018.

[61] R. Kamalapurkar, P. Walters, J. Rosenfeld, and W. Dixon. *Reinforcement Learning for Optimal Feedback Control*. Springer, 2018.

[62] H. K. Khalil and J. W. Grizzle. *Nonlinear systems*, volume 3. Prentice hall Upper Saddle River, NJ, 2002.

[63] J. Kirschner, M. Mutny, N. Hiller, R. Ischebeck, and A. Krause. Adaptive and safe bayesian optimization in high dimensions via one-dimensional subspaces. In *Proc. 36th Int. Conf. Mach. Learn.*, 2019.

[64] B. Kiumarsi, K. G. Vamvoudakis, H. Modares, and F. L. Lewis. Optimal and autonomous control using reinforcement learning: A survey. *IEEE Transactions on Neural Networks and Learning Systems*, 29(6):2042–2062, 2018.

[65] W. F. . B. U. Koch. *Flight Controller Synthesis Via Deep Reinforcement Learning*. PhD thesis, Boston University, 2019.

[66] J. Lenka, Arkdev, and M. K. Sarkar. Nonlinear stabilization: Control contraction metric approach via sum of square programming technique. *2nd International Conference on Energy, Power and Environment: Towards Smart Technology, ICEPE 2018*, pages 1–6, 2019.

[67] K. Leung and I. R. Manchester. Nonlinear stabilization via control contraction metrics: A pseudospectral approach for computing geodesics. *Proc. Am. Control Conf.*, pages 1284–1289, 2017.

[68] J. Löfberg. YALMIP: A toolbox for modeling and optimization in MATLAB. *Proc. IEEE Int. Symp. Comput. Control Syst. Des.*, pages 284–289, 2004.

[69] J. Löfberg. Pre- and post-processing sum-of-squares programs in practice. *IEEE Transactions on Automatic Control*, 54(5):1007–1011, 2009.

[70] W. Lohmiller and J.-J. E. Slotine. On contraction analysis for non-linear systems. *Automatica*, 34(6):683–696, 1998.

[71] B. T. Lopez and J.-J. E. Slotine. Contraction metrics in adaptive nonlinear control. *arXiv preprint arXiv:1912.13138*, 2019.

[72] A. Majumdar, A. A. Ahmadi, and R. Tedrake. Control design along trajectories with sums of squares programming. *Proc. - IEEE Int. Conf. Robot. Autom.*, pages 4054–4061, 2013.

[73] A. Majumdar and R. Tedrake. Funnel libraries for real-time robust feedback motion planning. *Int. J. Rob. Res.*, 36(8):947–982, 2017.

[74] A. Majumdar, R. Vasudevan, M. M. Tobenkin, and R. Tedrake. Convex optimization of nonlinear feedback controllers via occupation measures. *The International Journal of Robotics Research*, 33(9):1209–1230, 2014.

[75] I. R. Manchester and J.-J. E. Slotine. Control contraction metrics, robust control and observer duality. *arXiv*, 2014.

[76] I. R. Manchester and J.-J. E. Slotine. Control contraction metrics: Convex and intrinsic criteria for nonlinear feedback design. *IEEE Transactions on Automatic Control*, 62(6):3046–3053, 2017.

[77] I. R. Manchester and J. J. E. Slotine. Robust control contraction metrics: A convex approach to nonlinear state-feedback control. *IEEE Control Systems Letters*, 2(3):333–338, 2018.

[78] I. R. Manchester, J. Z. Tang, and J.-J. E. Slotine. Unifying robot trajectory tracking with control contraction metrics. In *Robotics Research*, pages 403–418. Springer, 2018.

[79] Z. Marinho, B. Boots, A. Dragan, A. Byravan, G. J. Gordon, and S. Srinivasa. Functional gradient motion planning in reproducing kernel hilbert spaces. In *Proceedings of Robotics: Science and Systems*, AnnArbor, Michigan, June 2016.

[80] D. T. McRuer. Pilot-induced oscillations and human dynamic behavior. Technical report, NASA, 1995.

[81] A. Megretski. Spot (systems polynomial optimization tools) manual, 2010.

[82] M. W. Mehrez, G. K. Mann, and R. G. Gosine. Comparison of stabilizing NMPC designs for wheeled mobile robots: An experimental study. *MERCon 2015 - Moratuwa Eng. Res. Conf.*, (September):130–135, 2015.

[83] M. W. Mehrez, K. Worthmann, J. P. Cenerini, M. Osman, W. W. Melek, and S. Jeon. Model Predictive Control without terminal constraints or costs for holonomic mobile robots. *Rob. Auton. Syst.*, 127(February), 2020.

[84] F. Meng, D. Wang, P. Yang, G. Xie, and C. Romero-Meléndez. Application of sum of squares method in nonlinear h control for satellite attitude maneuvers. *Complexity*, 2019, 2019.

[85] I. Mitchell. A toolbox of level set methods, 2004.

[86] I. M. Mitchell and J. A. Templeton. A toolbox of Hamilton- Jacobi solvers for analysis of nondeterministic continuous and hybrid systems. *Lect. Notes Comput. Sci.*, 3414:480–494, 2005.

[87] H. Mo and G. Farid. Nonlinear and adaptive intelligent control techniques for quadrotor uav – a survey. *Asian Journal of Control*, 21(2):989–1008, 2019.

[88] K. P. Murphy. *Machine Learning A Probabilistic Perspective*. MIT Press, London, England, 2012.

[89] E. Najafi, R. Babuška, and G. A. Lopes. A fast sampling method for estimating the domain of attraction. *Nonlinear Dynamics*, 86(2):823–834, 2016.

[90] NASA/JPL-Caltech. Nasa's insight lander, 2018. [Online; accessed April 16, 2021].

[91] A. Papachristodoulou, J. Anderson, G. Valmorbida, S. Prajna, P. Seiler, and P. Parrilo. Sostools version 3.00 sum of squares optimization toolbox for matlab. *arXiv*, 2013.

[92] P. Patrinos and A. Bemporad. An accelerated dual gradient-projection algorithm for embedded linear model predictive control. *IEEE Trans. Automat. Contr.*, 59(1):18–33, 2014.

[93] A. S. Polydoros and L. Nalpantidis. Survey of model-based reinforcement learning: Applications on robotics. *Journal of Intelligent and Robotic Systems: Theory and Applications*, 86(2):153–173, 2017.

[94] A. Ray, J. Achiam, and D. Amodei. Benchmarking safe exploration in deep reinforcement learning. *SemanticScholar*, 2019.

[95] B. Recht. A tour of reinforcement learning: The view from continuous control. *Annual Review of Control, Robotics, and Autonomous Systems*, 2(1):253–279, 2018.

[96] S. M. Richards, F. Berkenkamp, and A. Krause. The lyapunov neural network: Adaptive stability certification for safe learning of dynamical systems. In *Conference on Robot Learning*, pages 466–476. PMLR, 2018.

[97] U. Rosolia and F. Borrelli. Learning how to autonomously race a car: A predictive control approach. *IEEE Transactions on Control Systems Technology*, 28(6):2713–2719, 2020.

[98] S. Saria and A. Subbaswamy. Tutorial: Safe and reliable machine learning. 2019.

[99] M. Schillinger, B. Hartmann, P. Skalecki, M. Meister, D. Nguyen-Tuong, and O. Nelles. Safe active learning and safe bayesian optimization for tuning a pi-controller. *IFAC-PapersOnLine*, 50(1):5967–5972, 2017.

[100] B. Settles. Active Learning Literature Survey. Technical Report 26 January, University of Wisconsin-Madison, 2010.

[101] V. Sindhwani, S. Tu, and M. Khansari. Learning Contracting Vector Fields For Stable Imitation Learning. *arXiv e-prints*, 2018.

[102] S. Singh, A. Majumdar, J. J. Slotine, and M. Pavone. Robust online motion planning via contraction theory and convex optimization. *Proc. - IEEE Int. Conf. Robot. Autom.*, pages 5883–5890, 2017.

[103] S. Singh, S. M. Richards, V. Sindhwani, J.-J. E. Slotine, and M. Pavone. Learning stabilizable nonlinear dynamics with contraction-based regularization. *The International Journal of Robotics Research*, 2020.

[104] S. Singh, V. Sindhwani, J.-J. E. Slotine, and M. Pavone. Learning Stabilizable Dynamical Systems via Control Contraction Metrics. *arXiv e-prints*, page arXiv:1808.00113, July 2018.

[105] S. W. Smith, H. Yin, and M. Arcak. Continuous abstraction of nonlinear systems using sum-of-squares programming. pages 8093–8098, 2019.

[106] U. M. Specs. Mil-f-8785c: Flying qualities of piloted airplanes. Technical Report November 1980, 1980.

[107] Y. Sui, A. Gotovos, J. W. Burdick, and A. Krause. Safe exploration for optimization with gaussian processes. *32nd International Conference of Machine Learning, ICML 2015*, 2:997–1005, 2015.

[108] Y. Sui, vincent Zhuang, J. Burdick, and Y. Yue. Stagewise safe Bayesian optimization with Gaussian processes. *Proceedings of the 35th International Conference on Machine Learning*, 80:4781–4789, 10–15 Jul 2018.

[109] R. S. Sutton and A. G. Barto. *Reinforcement learning: An introduction*. MIT press, 2018.

[110] R. Tedrake. Underactuated robotics: Algorithms for walking, running, swimming, flying, and manipulation (course notes for mit 6.832).

[111] R. Tedrake. *Underactuated Robotics*. 2022.

[112] R. Tedrake, I. R. Manchester, M. Tobenkin, and J. W. Roberts. LQR-trees: Feedback motion planning via sums-of-squares verification. *Int. J. Rob. Res.*, 29(8):1038–1052, 2010.

[113] H. Tsukamoto and S. Chung. Neural contraction metrics for robust estimation and control: A convex optimization approach. *IEEE Control Systems Letters*, 5(1):211–216, 2021.

[114] H. Tsukamoto and S. J. Chung. Robust controller design for stochastic nonlinear systems via convex optimization. *IEEE Transactions on Automatic Control*, pages 1–1, 2020.

[115] H. van Hasselt, Y. Doron, F. Strub, M. Hessel, N. Sonnerat, and J. Modayil. Deep reinforcement learning and the deadly triad. *arXiv e-prints*, 2018.

[116] L. Vandenberghe and S. Boyd. Semidefinite programming. *SIAM review*, 38(1):49–95, 1996.

[117] D. Vrabie, O. Pastravanu, M. Abu-Khalaf, and F. L. Lewis. Adaptive optimal control for continuous-time linear systems based on policy iteration. *Automatica*, 45(2):477–484, 2009.

[118] K. P. Wabersich, L. Hewing, A. Carron, and M. N. Zeilinger. Probabilistic model predictive safety certification for learning-based control. pages 1–11, 2019.

[119] K. P. Wabersich and M. N. Zeilinger. Safe exploration of nonlinear dynamical systems: A predictive safety filter for reinforcement learning. *arXiv*, 2018.

[120] K. P. Wabersich and M. N. Zeilinger. Scalable synthesis of safety certificates from data with application to learning-based control. *2018 Eur. Control Conf. ECC 2018*, pages 1691–1697, 2018.

[121] D. Wang, H. He, and D. Liu. Adaptive critic nonlinear robust control: A survey. *IEEE Transactions on Cybernetics*, 47(10):3429–3451, 2017.

[122] Y. Wang and S. Boyd. Fast Model Predictive Control Using Online Optimization. *IEEE Trans. Control Syst. Technol.*, 18(2):267–278, 2010.

[123] W. Xiang and T. T. Johnson. Reachability analysis and safety verification for neural network control systems. *arXiv*, pages 1–21, 2018.

[124] W. Xiang, P. Musau, A. A. Wild, D. M. Lopez, N. Hamilton, X. Yang, J. Rosenfeld, and T. T. Johnson. Verification for machine learning, autonomy, and neural networks survey. *arXiv*, pages 1–51, 2018.

[125] X. Zhao, A. Banks, J. Sharp, V. Robu, D. Flynn, M. Fisher, and X. Huang. A safety framework for critical systems utilising deep neural networks. pages 244–259, 2020.

[126] Y. Zheng, G. Fantuzzi, and A. Papachristodoulou. Sparse sum-of-squares (SOS) optimization : A bridge between DSOS/SDSOS and SOS optimization for sparse polynomials. In *2019 Am. Control Conf.*, pages 5513–5518, 2019.

[127] Y. Zhou, E.-J. Van Kampen, and Q. Chu. Incremental model based heuristic dynamic programming for nonlinear adaptive flight control. *Proceedings of the international micro air vehicles conference and competition (IMAV)*, 2016.

[128] G. Zhu, J. Du, J. Li, and Y. Kao. Robust adaptive nn tracking control for mimo uncertain nonlinear systems with completely unknown control gains under input saturations. *Neurocomputing*, 365:125–136, 2019.

Chapter 3

Safe Control Learning Pipeline via Static Control Contraction Metric

3.1 Introduction

There is an increased interest in learning techniques for aircraft control systems that can adapt to changes in the environment or changes in an aircrafts' dynamics in real time [13, 47, 21, 25]. For safety critical systems this approach introduces the risks associated with the uncertainty in the system response by generating control actions which may be unsafe [3, 11, 9]. Safety constraints are typically stability requirements or navigational constraints though in this work we focus on stability. Modern control techniques have approached the problem of control adaptation by testing an airframe in mutliple configurations in simulation and flight tests [24] and then, by inspection, designing controllers that are either robust or that adapt to changes at specified operating points [54, 7]. Robust techniques however, introduce conservative control policies that are not ideal for maneuvrable aircraft and adaptive techniques like gain scheduling are expensive to develop and inflexible to airframe modifcations or mission damage. Adaptive control techniques such as backstepping [19, 26, 40, 49], sliding mode control (SMC) [57, 45], nonlinear dynamic inversion (NDI) [4, 2] require accurate dynamics models that may not have formal guarantees for stability and changes in the dynamics are limited to be within approximations close to the original design of the aircraft. Recently incremental safe learning approaches have been introduced that offer guarantees on stability whilst learning more about the

flight envelope to update a control policy online. This iterative learning process via a reinforcement of learned policies via rewards ensures the optimal performance is achieved whilst remaining within controlled bounds, [10, 22]. Stability certifcation for nonlinear systems has traditionally been in the form of control Lyapunov functions (CLFs) where the system equations require a certain structure as an energy like function. For highly nonlinear function classes the Lyapunov candidate functions are not easy to find however or may be intractable, [35]. Therefore the use of CLFs limits the complexity and the scope of the types of learning problems that can be automated without offline inspection. Nonlinear model predictive control techniques have typically been limited by computational constraints for online control synthesis for high dimensional systems or long time horizons and also require accurate models over an undetermined time horizon to safely predict future behaviour [55, 39, 23, 20]. The use of convex optimisation, if a problem can be posed as one, offers greater computational efficiency in deriving solutions in real time whilst also remaining tractable [36, 53, 12, 30]. This paper explores the use of a control contraction metric (CCM) for a longitudinal dynamics learning problem for an F-16 aircraft. The aim is to compare the learning technique to traditional approaches like linear quadratic regulator (LQR) control and gain scheduling. LQR control is the most widely adopted optimal control technique over an infinite horizon to minimise a weighted sum of squares for the state and control inputs of a system. For linear systems this optimisation can be solved as a linear matrix inequality (LMI) or by solving the algebraic Ricatti equation (ARE). The solutions however are only valid for local approximations of nonlinear systems where the dynamics function must first be linearised at equilibrium points. In this paper we compare the numerical techniques requried for both approaches as well as the increase in utility, robustness and uncertainty handling capabilities of the CCM technique.

3.1.1 Related Works

This paper is inspired by the recent works in the field of safe model-based reinforcment learning. The key theme of the work includes learning control policies from trajectory data with stability certificates [47, 10, 11], incremental safe learning from data [18, 8, 21], and the recent works in contraction theory [35, 36, 47, 56, 53, 51]. In learning control policies from data the work by [47] uses a control theoretic regularisation technique to both learn a convex solution for the dynamics approximation while also ensuring that the learned function is one that is stabilisable using alternation between two optimisation problems. The results showed that learning from a small number of expert trajectories the control theoretic regularised model outperformed standard ridge regression techniques that have no inherent stability certification guarantees. In [10] the control system is resticted to a safe set while learning probability guarantees on stability via a Lyapunov function using Gaussian process regression on the training data. In this manner the safe set is incrementally increased in size as more confidence is gained in the approximation of the true dynamics of the system. In [11] instead of trying to certify a closed form solution of the dynamics function, certificates for stability are derived from sampled trajectories by checking the Lie derivative of a function V is negative along the trajectories. The approach is shown to efficiently certify systems empirically as an alternative for safe adaptive control. In [18] control sythesis for a system both excites and regulates the dynamics to be within safe bounds for an unknown linear system by using tools from system level synthesis (SLS). SLS is a framework that utilizes the closed loop transfer function to enforce input and output constraints on the system. In [21] a safety framework is adopted using Hamilton-Jacobi Isaacs reachability techniques to guarantee safe bounds on the dynamics of the system with wind disturbances. The framework provides probabilistic guarantees using Gaussian process regression on the uncertain dynamics that triggers changes in the control policy to remain within a safe envelope. The recent works in control contraction theory extend Lyapunov theory to the differential dynamics of a system by exploiting linear eigenvalue analysis of the Jacobian [31]. Since

the theory's first introduction [32] it has since been developed as a control contraction metric CCM, [35], for ensuring stabilisable control. The technique offers a computational middle ground between linear methods like LQR control and the more computationally expensive nonlinear model predictive control (NMPC) techniques that are impractical for complex systems [27]. The convex properties of CCM linear matrix inequalities (LMIs) have also been exploited for efficient learning with the use of neural neworks in [53] and using sum of squares (SOS) techniques [46, 37] which make them attracive for real time learning. The global function properties have also been shown to solve the linear parameter varying stability uncertainty problem due to the coupling of unknown interpolation dynamics with that of the unknown true system dynamics by way a virtual control contraction metrics [56]. Previous work related to adaptive control approaches such as sliding mode control includes [57, 45, 19, 26], and linear parameter varying techniques such as [1, 58, 54, 15]. For prior work related to the simulation and testing of control policies for the F-16 the reader may find useful [41, 49, 44, 28].

3.1.2 Contributions

The main contributions are:

1. We propose an online learning pipeline for a flight controller that can certify stability of non-minimum phase systems using the CCM technique.

2. We compare a static metric optimisation to the optimisation of an LQR controller with proportional gains. A review of the two techniques and the connections to the Algebraic Ricatti formula is discussed to highlight the parallels in the two approaches. Further we show how two similar controllers can be found by optimisations in the two approaches and variations thereof based on a choice of cost function.

3. We propose a new objective function following on from the work in [46] to tailor the condition number of a metric to the dynamics problem to design a controller

relevant to aircraft design specifications.

4. The chapter also offers an insight into the use of metrics via an illustrative example of a static formulation of a metric.

The rest of the paper is structured as follows. Section 5.2 describes longitudinal modelling and simulation of the F-16 based on NASA wind tunnel data [42], section III details the 2D short period mode approximation for our learning problem, section IV a review of LQR theory, section V control contraction metric theory and LTI theory, section VI discusses the results of our study and section VII a conclusion and future work.

3.2 Overview

The aim of this paper is to explore an online learning pipeline for a nonlinear system based on the CCM technique with a focus on explainability, learning speed and safety. To effectively design a learning pipeline that fulfills all of the above requirements typically means a tradeoff must be made between the conservatism of the resulting controller or a black box model free learning solution that is difficult to both certify and explain in terms of the system dynamics. The learning pipeline in this paper consists of trajectory data collection and regression of the full state dynamics, optimisation of a control contraction metric (CCM) and online control synthesis of the learned feedback controller. The CCM guarantees exponential convergence of all pairwise trajectories due to any disturbance, where the maximum disturbance is bound by control saturation and handling constraints. This paper does not develop a full nonlinear control policy as a function of the state space but first looks at a static time invariant metric for a short period pitching mode of the aircraft to explore the use of convex learning methods. The stages of the learning pipeline have been labelled for easier navigation as functions with codes which are listed in Table 3.1.

Table 3.1 Stages of the Learning Pipeline

Stage	**Procedure**
F00 -	Batch Online Data Collection
F01 -	Quadratic Programme (QP) Online Regression
F02 -	CCM Optimisation Problem
F03 -	Online Controller Synthesis

1. **F00** The first part of the learning process is the collection of trajectory data without feedback control and an excitation via a sinusoidal stick input. Data is collected over a 40s time horizon. In practice the aircraft FCS remains a closed loop system for safety but for this initial study the data collection and control synthesis are separated as open and closed loop simulations. The collection of open vs closed loop data is thought to be part of a separate study in system identification and so is not considered for this analysis.

2. **F01** The second stage of learning is a regression on the collected flight data as a quadratic program with linear constraints of regressors and coefficients. For the quadratic programme we use YALMIP [29] a linear matrix and symbolic parser of semi-definite programming (SDP) variables and Mosek [5] as the convex optimisation solver. The L^1 norm[1] is used as a minimising objective as well as the function approximation error $(\hat{F}(x,u) - \dot{x})$ and the function coefficient norm. Minimising the L^1 norm promotes sparsity of learned regressors and minimising the coefficient norm promotes optimal dynamics. The function can also be broken down into separate affine blocks based on a regularisation weighting to favour expert knowledge about the systems dynamics if learning a nonlinear model. The learning problem in this paper uses a linear model of nonlinear functions or monomials termed, features in machine learning, as a convex optimisation. Sparsity of features as adopted in the

[1]The Euclidean norm is defined as $\|x\| := \sqrt{x_1^2 + \cdots + x_n^2}$ also called the L^2 norm whereas the L^1 norm is defined as $\|x\|_1 := \sum_{i=1}^{n} \|x_i\|$ is the sum of the absolute values of the columns of the input vector x.

work by [13] means that the control decisions can be interrogated and understood more easily which is benficial for safety critical systems.

3. **F02** The learned dynamics function and the system constraints are then used to search for an optimal CCM for the system. Where a positive definite matrix W and a contraction rate λ is found from a linear matrix inequality that both certify stability for the system and minimise the overshoot of the controlled response and maximise the contraction rate to improve response time as a higher-level objective. In this study we search for a static metric for specified operating points, similar to gain scheduling for LQR control.

4. **F03** The final part of the pipeline is the generation of a controller via the learned policy. For state varying metrics this would be a geodesic found via optimisation over smooth paths joining the state tracking differential but for a static metric this is more simplified and only requires a straight line formulation which is a linear matrix transform of the tracking differential. For parameter varying techniques the metric varies with the operating points. In state varying metrics the optimisation as an LMI also adopts a state varying gain term $\rho(x)$ but is not necessary in the linear case.

3.2.1 The CCM Controller Learning Framework and Full State Feedback Control

The CCM learning framework is shown in Fig. 3.1 and consists of a two state feedback controller with a learning component that samples trajectory data and control inputs online; a CCM metric optimisation component that accepts the function approximation $\hat{F}$ and learned parameters θ_f, θ_b; and a control synthesis function that accepts the learned metric M and control effectiveness parameters B to determine the feedback gains for the system. The CCM controller is comparable to a full state feedback controller in pitch rate and angle of attack as shown in Fig. 3.2, [16]. The full state feedback controller can be

designed using pole placement or LQR gains and the controller acts as a full state regulator in pitch. The simplification of the controller to two states means that control in pitch is set by the pilots commands via the stick and steady state errors can be reduced for the alpha channel by varying the input gain block K_r.

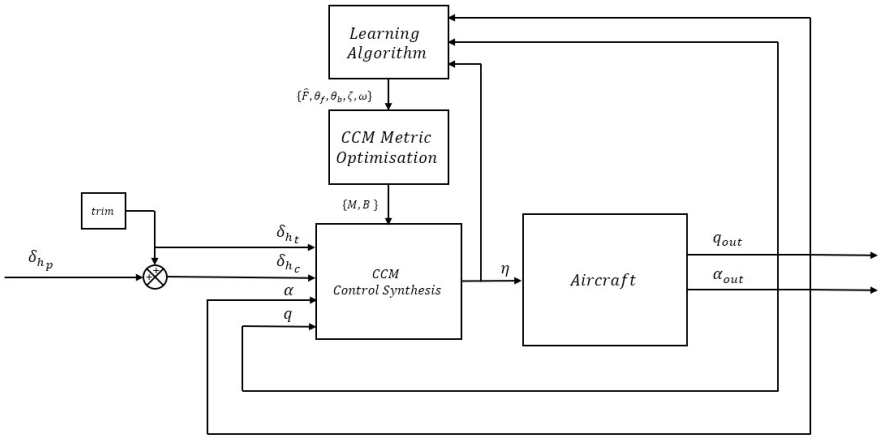


Figure 3.1 – CCM Learning Framework

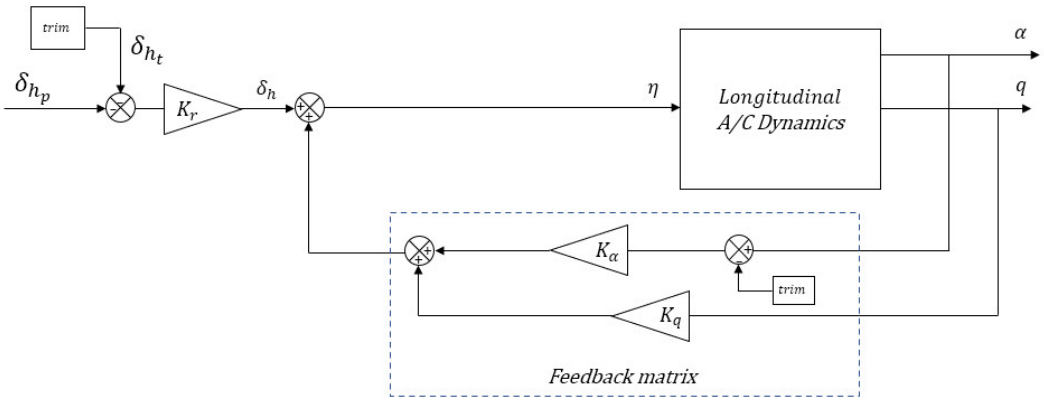


Figure 3.2 – Benchmark 2-Loop Proportional Feedback Controller

3.3 The Learning Problem

3.3.1 Control Affine Approximation

The learning algorithm presented in this work relies on a general control-affine equation of the form in Eq. 3.1:

$$\dot{x}(t) = f\left(x(t)\right) + B\left(x(t)\right)u(t) + d(t) \tag{3.1}$$

where the state vector is $x \in \mathcal{X} \subseteq \mathbb{R}^n$ and the control input $u \in \mathcal{U} \subseteq \mathbb{R}^m$, with disturbance $d \in \mathcal{D} \subseteq \mathbb{R}^{n_d}$. The functions are mappings, $f : \mathcal{X} \to \mathbb{R}^n$, and $B : \mathcal{X} \to \mathbb{R}^{n\times m}$ in general but for this study there is only one input the elevator and so the B matrix is a single column. The model function approximation is represented as $\hat{F}(x, u, d) = \hat{f}\left(x\right) + \hat{B}\left(x\right)u + \hat{d}(t)$. A learning algorithm updates the feedback control policy, $\pi_i(x, t)$ via optimisation of a cost function under dynamic constraints as data is collected online in batches over a time horizon T as $\{(x_i, x_i^*, u_i, \dot{x}_i)\}_{i=0}^{T}$ where x^* is the set point or tracking reference signal. The system is initialised with an a priori control policy π_0 and the approximation of the system dynamics is improved over time as the system explores the state space. The disturbance $d(t)$ is assumed to be bound by a compact set $\hat{\mathcal{D}} \subseteq \mathcal{D}$. The true function, $F(.)$, is approximated by, $\hat{F}(.)$, using a quadratic programme optimisation, Eq. 3.2, of the observed data $\{(\dot{x}_i, x_i, u_i)\}$. The disturbance term is omitted in the regression to begin with. Where μ_{θ_p} is the regularisation term of the model parameters, minimised over the L^1 norm in the optimisation, and subscript p is the dimension α or q. For this study we learn the dynamics independently of the introduction of disturbances to simplify the analysis.

$$\min_{\theta} \sum_{i=1}^{n} \|\hat{f}(x_i) + \hat{B}(x_i)u_i - \dot{x}_i\|^2 + \mu_{\theta_p}\|\theta_p\|_1 \tag{3.2}$$

3.3.2 Longitudinal 2D Short Period Approximation

The proposed structure for the learning algorithm simplifies the 3 DOF longitudinal dynamics to two state equations in α and q with a linear parameterisation. In this section we show that this approximation and simplification under certain constraints is valid for the short period mode dynamics.

Longitudinal Nonlinear Dynamics in α and q

The full nonlinear equations of motion for the aircraft longitudinal motion where lateral and directional dynamics are negelcted are given by eq. 3.3, [42].

$$\begin{aligned}
\dot{u} &= -qw - g\sin\theta + \frac{\bar{q}S}{m}C_{X_t}(\alpha, q, \delta_h) + \frac{F_T}{m} \\
\dot{w} &= qu + g\cos\theta + \frac{\bar{q}S}{m}C_{Z_t}(\alpha, q, \delta_h) \\
\dot{\theta} &= q \\
\dot{V_T} &= \frac{(u\dot{u} + w\dot{w})}{V_T} \\
\dot{\alpha} &= \frac{u\dot{w} - w\dot{u}}{u^2 + w^2} \\
\dot{q} &= \frac{C_{m_t}(\alpha, q, \delta_h)\bar{q}S\bar{c}}{I_{yy}}
\end{aligned} \tag{3.3}$$

where $u, w, \theta, V_T, \alpha$ and q are the aircraft's body x-axis velocity, body z-axis velocity, pitch angle, total velocity, angle-of-attack and pitch-rate respectively. F_T is the single engine thrust force. m is the mass of the aircraft and I_{yy} the moment of inertia about the y-axis. $\bar{q}$ is the dynamic pressure which is equal to $\frac{1}{2}\rho V_T^2$. S is the wing area and $\bar{c}$ is the mean aerodynamic chord.

To derive the expression for $\dot{\alpha}$, we can substitute the expressions for $\dot{u}$ and $\dot{w}$ from the body acceleration equations:

$$\dot{\alpha} = \frac{u(qu + g\cos(\theta) + \frac{\bar{q}S}{m}C_{Z_t}(\alpha, q, \delta_h)) - w(-qw - g\sin(\theta) + \frac{\bar{q}S}{m}C_{X_t}(\alpha, q, \delta_h) + \frac{F_T}{m})}{u^2 + w^2} \tag{3.4}$$

Assuming small angles of attack for α, implies $\sin(\alpha) \approx \alpha$ and $\cos(\alpha) \approx 1$, and considering steady flight where $\theta \approx \alpha$ and $V_T = \sqrt{u^2 + w^2} \approx u$ (since $w << u$ in small angle approximations), the equation simplifies to:

$$\dot{\alpha} = q + \frac{g}{V}\cos(\alpha) + \frac{\bar{q}S}{mV}(C_{X_t}(\alpha, q, \delta_h)\sin(\alpha) - C_{Z_t}(\alpha, q, \delta_h)\cos(\alpha)) \tag{3.5}$$

Where the $\dot{q}$ dynamics remain the same:

$$\dot{q} \quad = \quad \frac{C_{m_t}(\alpha, q, \delta_h)\bar{q}S\bar{c}}{I_{yy}} \tag{3.6}$$

These equations provide a direct relationship between the rate of change of angle of attack and the pitch rate, gravitational forces, and aerodynamic coefficients, under the assumptions of small angle of attack and steady, level flight where C_{X_t}, C_{Z_t}, and C_{m_t} are aerodynamic coefficient functions of the system. These coefficients are parameterized as follows, [42]:

$$C_{X_t}(\alpha, q, \delta_h) = C_{X_0}(\delta_h) + C_{X_1}(\delta_h)\alpha + \frac{\bar{c}q}{2V}C_{X_q}, \tag{3.7}$$

$$C_{Z_t}(\alpha, q, \delta_h) = C_{Z_0}(\delta_h) + C_{Z_1}(\delta_h)\alpha + \frac{\bar{c}q}{2V}C_{Z_q}, \tag{3.8}$$

$$C_{m_t}(\alpha, q, \delta_h) = C_{m_0}(\delta_h) + C_{Z_t}(x_{np} - x_{cg}) + \frac{\bar{c}q}{2V}C_{m_q}, \tag{3.9}$$

with C_{X_q}, C_{Z_q}, and C_{m_q} approximately constant for the short period mode dynamics.

3.3.3 Parameterisation and Expert Analysis

The aerodynamic coefficient equations are then parameterised as a set of linear functions in θ for nonlinear features of the input variables (α, q, δ_h). Inspection of the lookup tables over a range of α between 0 and 20° produces an expert sparse model of the function approximation that informs the features required in the regression function. It can be seen that the input control term δ_h in this system is nonlinear with monomial terms of up to 4^{th} order, Eq. 3.10. Due to the assumptions in the class of functions used to find a CCM however a simplification must first be made on the regression function to only include linear terms of δ_h. An assumption which can be investigated for accuracy later on.

$$
\begin{aligned}
C_{X_0}(\delta_h) &= \theta_1^T \phi_1(\delta_h) = \theta_{10} + \theta_{11}\delta_h + \theta_{12}\delta_h^2 \\
C_{Z_0}(\delta_h) &= \theta_2^T \phi_2(\delta_h) = \theta_{20} + \theta_{21}\delta_h \\
C_{m_0}(\delta_h) &= \theta_3^T \phi_3(\delta_h) = \theta_{30} + \theta_{31}\delta_h + \theta_{32}\delta_h^2 \\
C_{X_1}(\delta_h) &= \theta_4^T \phi_4(\delta_h) = \theta_{40} + \theta_{41}\delta_h \\
C_{Z_1}(\delta_h) &= \theta_5^T \phi_5(\delta_h) = \theta_{50} + \theta_{51}\delta_h + \theta_{52}\delta_h^2 + \theta_{53}\delta_h^3 + \theta_{54}\delta_h^4 \\
C_{m_1}(\delta_h) &= \theta_6^T \phi_6(\delta_h) = \theta_{60} + \theta_{61}\delta_h + \theta_{62}\delta_h^2
\end{aligned} \tag{3.10}
$$

Small Perturbation Theory and Short Period Mode Decoupling

Typically for aircraft modelling the system dynamics are represented as a linear set of equations that represent the dynamics sensitivity to small perturbations from a trimmed state over a limited range from the state of equilibrium where the linear assumptions still hold. Small perturbation theory simplifies the analysis of an aircraft's dynamics by considering small deviations from the trim condition, allowing for the linearization of aerodynamic coefficients [16, 50].

To derive the 2D state space reduced order model we first look at the typical longitudinal 4th order model, 3.11.

The 4D state space model for an aircraft's longitudinal dynamics is given by:

$$\begin{bmatrix} \dot{u} \\ \dot{w} \\ \dot{q} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} X_u & X_w & X_q & X_\theta \\ Z_u & Z_w & Z_q & Z_\theta \\ M_u & M_w & M_q & M_\theta \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} u \\ w \\ q \\ \theta \end{bmatrix} + \begin{bmatrix} X_{\delta_h} \\ Z_{\delta_h} \\ M_{\delta_h} \\ 0 \end{bmatrix} \delta_h \tag{3.11}$$

The variables u, w, q, and δ_h represent the body x-axis velocity, body z-axis velocity, pitch rate, and pitch angle, respectively. The control input δ_h corresponds to the elevator deflection.

The characteristic polynomial defining the system's dynamics and stability properties factors into two distinct modes: the Phugoid and Short Period modes. It is expressed as:

$$\Delta(s) = (s^2 + 2\zeta_p\omega_p s + \omega_p^2)(s^2 + 2\zeta_s\omega_s s + \omega_s^2) = 0 \tag{3.12}$$

To focus on the short-period dynamics and simplify the analysis, we consider the relationship between the angle of attack α and the body velocities u and w. For small angle approximations, α is related to u and w by $\alpha \approx \frac{w}{u}$, and its derivative by $\dot{\alpha} \approx \frac{\dot{w}u - w\dot{u}}{u^2}$. This approximation allows us to express α and its dynamics in terms of u and w, facilitating the reduction of the 4D model to a 2D model focusing on α and q.

Implementing these changes into the original 4D state space model, we derive the 2D reduced-order model for the short-period dynamics as follows:

$$\begin{bmatrix} \dot{\alpha} \\ \dot{q} \end{bmatrix} = \begin{bmatrix} Z_\alpha & Z_q \\ M_\alpha & M_q \end{bmatrix} \begin{bmatrix} \alpha \\ q \end{bmatrix} + \begin{bmatrix} B_\alpha \\ B_q \end{bmatrix} \delta_h \tag{3.13}$$

In this reduced-order model, Z_α, Z_q, M_α, and M_q are coefficients of the aerodynamic and control effects on the angle of attack and pitch rate dynamics, respectively. This model reduces the state space and dynamics to two dimensions relevant to the short-period mode.

To arrive at the 2D state space model we make two key assumptions. First, we as-

sume that the velocity (V), altitude, and thrust are constant, which is reasonable for the short period mode where changes in these variables are relatively slow compared to the dynamics of α and q. Under this assumption, we can neglect the changes in the axial force equation related to acceleration and focus on the rotational dynamics.

Second, we assume that the aircraft operates within a specific angle of attack range, from -5 to 15 degrees, which typically encapsulates the short period mode. Within this range, we can linearise the aerodynamic coefficients around a trim condition and consider them to be functions of α, q, and control surface deflection δ_h. These simplifications give a reduced-order nonlinear model to describe the aircraft's short-period response and are valid for the specific flight regime and are not appropriate for manoeuvers that involve large changes in velocity, altitude, or thrust, or when considering the effects of high angle of attack where nonlinearities become more significant.

3.3.4 Validation of the Decoupled Short Period Dynamics

The validity of the simplified state-space model, which focuses on the linearized short-period mode dynamics represented by angle of attack (α) and pitch rate (q), is confirmed by approximating these dynamics as a 2D linear function. The effectiveness of this simplification is assessed by comparing the response of the simplified model with that of a nonlinear simulation incorporating the full set of coupled equations of motion for both Phugoid and Short Period modes, as detailed in Section 5.5.

In Section 3.8, we present a linear-to-nonlinear comparison of simulated trajectories, depicted in Fig. 3.10, subject to sinusoidal input disturbances within the flight envelope designated for short period mode analysis. This comparison is conducted over a time horizon of 40 seconds. The results demonstrate that the linearised 2D model provides a close approximation to the behavior captured by the full 4th order dynamics simulation over the selected timeframe so that a control policy derived from the approximation will provide control strategies effective for the aircraft within the specified flight conditions for the baseline case.

3.4 The Learning Pipeline

In this section we describe in more detail the stages of the learning approach adopted in this paper.

3.4.1 F00 - Exploration and Data Processing of Trajectory Data

The first stage of the learning pipeline is the collection of trajectory data under control but as an open loop with a nominal control policy to ensure stability over a limited range of motion or states within the state space. It is envisaged that maneuvers and commands from the pilots stick have a conservative feedback policy, ie a high gain, when the tracking errors deviate by a threshold amount or when the aircraft approaches the limits of the safe operating envelope to remain within a range of angle of attack, pitch rate and pitch acceleration. For the experiment a 40s time horizon T is used to collect trajectory data as training data $\{(\dot{x}, x^*, x, u)\}_{t=0}^{T}$. The data collected online is batch processed as an array into a Library Φ, a column list of regressor functions forming a row history depending upon the sampling rate which for this study is set at $0.01s$.

3.4.2 F01 - Online Regression QP Problem

The online regression problem is posed as a quadratic programme (QP) in Yalmip using the convex solver Mosek. A generic QP problem, 3.14, is the minimisation of a function $f(x)$ with linear $q^T x$ and quadratic $\frac{1}{2}x^T Q x$ terms. The minimisation is typically constrained by linear constraints as equality 3.15 and inequality 3.16, 3.17 conditions on the state variable x.

$$\min \qquad f(x) = q^T x + \frac{1}{2} x^T Q x \tag{3.14}$$

$$s.t. \qquad Ax = a \tag{3.15}$$

$$Bx \leq b \tag{3.16}$$

$$x \geq 0 \tag{3.17}$$

The quadratic equation minimises the error term of the dynamics, a least squares problem, and an additional regularisation term is added to the optimisation to promote sparsity by penalising the L1 norm of the θ parameters defining the function approximation. It is important to note that this is not possible using a standard ordinary least squares approach and requires a convex solver. The optimisation can also be embedded for speed by utilising the Yalmip optimizer functionality which produces solutions in sub-millisecond time frames.

$$\min_{\theta} \sum_{i=1}^{n} (\|e_{x_i}\|_1 + e_{x_i}^T e_{x_i}) + \mu \|\theta\|_1 \tag{3.18}$$

$$s.t. \qquad e_{x_i} = \Phi \theta_x - \dot{x} \tag{3.19}$$

$$\theta_x[i] = 0, \forall i \in I_x = 0 \tag{3.20}$$

$$\tag{3.21}$$

The error term e_{x_i} is the error representing the 2D state in α and q. The parameters θ_x for each equation are constrained to equal zero using an index vector I_x that contains manually selected coefficient terms for the model selection of monomials in the library matrix Φ_x. The regularisation term can be further expanded into an affine set of progressive models where the regularisation term represents the probability weighting of the model. By assigning a higher regularization value to higher-order models or those that are less likely, the preference for sparse, lower-order models within an acceptable accu-

Table 3.2 Index for Library Regressors for the Polynomial Model functions

	$Model$	$\#1$		$\#2$		$\#3$		$\#4$	
Index	Φ_{Poly}	α	q	α	q	α	q	α	q
0	1	1	1	1	1	1	1	1	1
1	α	1	1	1	1	1	1	1	1
2	q	1	1	1	1	1	1	1	1
3	δ_h	1	1	1	1	1	1	1	1
4	α^2	0	0	1	1	1	1	1	1
5	αq	0	0	1	1	1	1	1	1
6	q^2	0	0	1	1	1	1	1	1
7	$\alpha\delta_h$	0	0	1	1	1	1	1	1
8	$q\delta_h$	0	0	1	1	1	1	1	1
9	δ_h^2	0	0	0	0	0	0	0	0
10	α^3	0	0	0	0	1	1	1	1
11	$\alpha^2 q$	0	0	0	0	1	1	1	1
12	αq^2	0	0	0	0	1	1	1	1
13	q^3	0	0	0	0	1	1	1	1
14	$\alpha^2\delta_h$	0	0	0	0	0	0	0	0
15	$\alpha q\delta_h$	0	0	0	0	0	0	0	0
16	$q^2\delta_h$	0	0	0	0	0	0	0	0
17	$\alpha\delta_h^2$	0	0	0	0	0	0	0	0
18	$q\delta_h^2$	0	0	0	0	0	0	0	0
19	α^4	0	0	0	0	0	0	1	1
20	$\alpha^3 q$	0	0	0	0	0	0	1	1
21	$\alpha^2 q^2$	0	0	0	0	0	0	1	1
22	αq^3	0	0	0	0	0	0	1	1
23	q^4	0	0	0	0	0	0	1	1

racy threshold for the dynamics regression problem is encouraged. This promotes the use of simpler models that can capture the system's behavior while minimizing the function complexity.

$$\min_{\theta} \sum_{i=1}^{n} (e_{x_i}^T e_{x_i}) + \mu_1 \|\theta_1\| + \mu_2 \|\theta_2\| + \mu_3 \|\theta_3\| \tag{3.22}$$

Table 3.2 gives the Library regressor vector Φ_{poly} and α and q equation index vectors for constraints imposed on the QP problem to enforce the use of only certain regressors from the library as part of the model approximation.

3.4.3 F02 - Optimisation Problem

The linear approximation for the dynamics determines the Jacobian matrix and B matrix terms for the search for a CCM dual metric W as a semi-definite programme (SDP). We use the software Yalmip, see [29] and Mosek [5] as a solver and we reference the work by [46] in determining an optimisation strategy for manipulating the constraints on the Eigenvalues of the metric. The standard CCM optimisation is similar to a Lyapunov optimisation to find a decay rate and P matrix but in the CCM optimisation, 4.5, a dual metric W is used and a maximisation of the trace adopted. The Eigenvalues are constrained to be greater than zero instead of one because the dual metric seeks values that once inverted become the limit of the positive Eigenvalues in the metric M. In this way the magnitude of the upper limit in metric is the smallest fraction in the dual metric. There is also the addition of the derivative constraint which contains the additional control matrix and exponential constraint terms parameterised by the contraction rate λ.

$$\min_{\{w_{11}w_{22}\}\in\mathbb{R}_{>0}} \quad (-w_{11} - w_{22}) \tag{3.23}$$

$$s.t. \quad W \geq 0 \tag{3.24}$$

$$-(AW + WA^T - BB^T + 2\lambda W) \geq 0 \tag{3.25}$$

$$\text{where} \quad \underline{\beta} I_2 \leq W \leq \bar{\beta} I_2 \tag{3.26}$$

$$W = \begin{bmatrix} w_{11} & w_{12} \\ w_{12} & w_{22} \end{bmatrix} \tag{3.27}$$

$$M = W^{-1} \tag{3.28}$$

$$\underline{\sigma}(M) = \frac{1}{\bar{\beta}}(W), \quad \bar{\sigma}(M) = \frac{1}{\underline{\beta}}(W), \tag{3.29}$$

3.4.4 Optimisation Algorithm A002

The first algorithm to find a CCM and contraction rate is adapted from [33], with modifications to suit the specific requirements of the CCM LMI. A lower and upper bound for the contraction rate λ is defined based on a reasonable estimate for the bounds of the specifications for the systems performance limits. The metric is then optimised with a bisection over λ to find the largest λ feasible while the trace of the metric is minimised.

Algorithm 1: CCM Optimisation A002

Input: A and B matrix coefficients of the learned dynamics equation linear approximation.
Initialize: $\lambda_{min} \leftarrow 0.1 \quad \lambda_{max} \leftarrow 3$
tol $\leftarrow 0.01$
$\lambda_{works} \leftarrow \lambda_{lower}$
while $\lambda_{upper} - \lambda_{lower} > 0.01$ **do**
    $\lambda_{loop} \leftarrow (\lambda_{upper} + \lambda_{lower})/2$
    $W \leftarrow \{w_{11}, w_{12}, w_{22}\} \leftarrow$ **CCM**
    if $(W_{feas} == true)$ **then**
        $\lambda_{upper} = \lambda_{loop}$
    else
        $\lambda_{lower} = \lambda_{loop}$
        $\lambda_{works} = \lambda_{loop}$

3.4.5 Optimisation Algorithm A005

In the next algorithm instead of using a bisection over λ, a range of values is searched over with performance parameters found for each iteration. The metric and contraction rate is optimised based on the performance parameters ζ and ω_n. Three conditions are used in the choice of the metric and contraction rate. The first, 3.30 is a cost function as adopted in [48] where the condition number divided by the square of the contraction rate. This has the effect of minimising the gains of the controller whilst maximising the decay rate for a certified stable system. The second condition, 3.31 is to find the minimum distance to a design goal for the damping ratio and natural frequency and to select the closest feasible

decay rate. The third condition, 3.32 is to maximise λ over the search space to produce the most aggressive controller.

Criteria 1:

$$min\left(\frac{cond(M)}{\lambda^2}\right) \tag{3.30}$$

Criteria 2:

$$min\left(\frac{\lambda}{\zeta\omega_n} - 1\right)^2 \tag{3.31}$$

Criteria 3:

$$max(\lambda) \tag{3.32}$$

Algorithm 2: CCM Optimisation A005

Input: A and B matrix coefficients of the learned dynamics equation linear approximation.
for *i = 1:50* **do**
    $\Delta\lambda = i \times 0.05$
    $\lambda = 0.1 + \Delta\lambda$
    $W \leftarrow \{w_{11}, w_{12}, w_{22}\} \leftarrow$ **CCM**
    if $(W_{feas} == true)$ **then**
        $M = W^{-1}$
        $CondN = \bar{\sigma}/\underline{\sigma}$
        $Cost \leftarrow CondN/\lambda^2$
        $ratio \leftarrow \lambda/(\zeta\omega_n)$
    else
        $M \leftarrow \mathbb{I}_2$
        $CondN \leftarrow 1$
        $Cost \leftarrow 0$
        $ratio \leftarrow 0$
Output: W for $(minCost|ratio|max\lambda)$

3.4.6 F03 - Control Synthesis Policy

Once a metric $M = W^{-1}$ is found an online controller uses the following formula to generate a control feedback signal from the tracking error, where u^* is a nominal open loop input.

$$u(t) = u^*(t) - 0.5\rho B^T M(x^*(t) - x(t)) \tag{3.33}$$

3.5 F-16 Longitudinal Modelling and Simulation

The F-16, Fig. 3.3 is a highly maneuverable aircraft and thus can be characterised by a highly nonlinear dynamic system in general over the full flight envelope. In order to have high maneuvrability the center of pressure is forward of the centre of gravity so that the horizontal stabiliser corrects for pitch up rotation by a positive lift command at the tail. This design also improves endurance because the lift to drag ratio is higher. Because of the inherent instability with the onset of increased angle of attack the aircraft requires a flight control system (FCS) that automatically corrects the alpha offset with a feedback controller to reduce the pilot workload and maintain the aircraft within safe operating and handling limits. The feedback gains therefore determine a tradeoff between an augmented closed loop stability and the maneuvrability of the aircraft. The tradeoff is determined by an acceptable pilot workload, expected pertubations, and the design responsiveness of the airframe. The aircraft also utilises several actuators to maintain controlled flight including leading edge flaps (LEFs) for high-alpha flow control, ailerons, symmetric split rear stabilisers, a rudder, speed-brakes and a thrust vectoring single engine. For this study however we limit the aircraft simulation to the longitudinal axis and only use the rear stabiliser as a pitch controller. The LEFs are effectively removed, the engine dynamics ignored and the thrust is set to equal drag at a cruising altitude so that the short period mode dynamics are simplified. In the real aircraft the aerodynamic force and moment

coefficients are functions of the payload, actuator deflections, structural deformations or damage incurred in flight as well as the Mach number, altitude, and weather conditions. Typically the changes in aerodynamic coefficents are accounted for using a gain schedule depending upon the Mach number and altitude which is developed via wind tunnel testing and extensive flight testing. When the aircraft is in service however changes occuring to the payload or structure of the aircraft need to be accounted for in the control policy by re-optimising the control gains to maintain safe flying and handling qualities. This initial study proposes an online method by which the system dynamics can be re-learned in-flight and a certified control policy iteratively updated by the FCS.

Figure 3.3 – F-16 Aircraft

3.5.1 The F-16 Reference Axis, Forces and Moments

Fig. 3.4 describes the longitudinal axis, forces, moments, and velocity vectors. The body axis components are in black and the wind axis components are shown in blue. The thrust T and elevator angle δ_h are control inputs and are shown in red, although the thrust is assumed constant. The aircraft motion is described by the body axis velocities in the axial direction u and in the normal direction w with the total velocity vector in the wind axis V_T. The weight of the aircraft mg acts at an angle θ towards the center of the Earth. The angle of attack α gives the body angle relative to the wind axis and the glide slope γ is the angle of motion of the aircraft relative to the Earth's horizontal axis. The pitch rate q and pitching moment M act positive nose up. The Aerodynamic forces are the lift L and drag

D in the wind axis and the $\bar{Z}$ body normal force and axial force $\bar{X}$ with the coefficients terms, C_M, C_L, C_D, C_Z, C_X, respectfully.

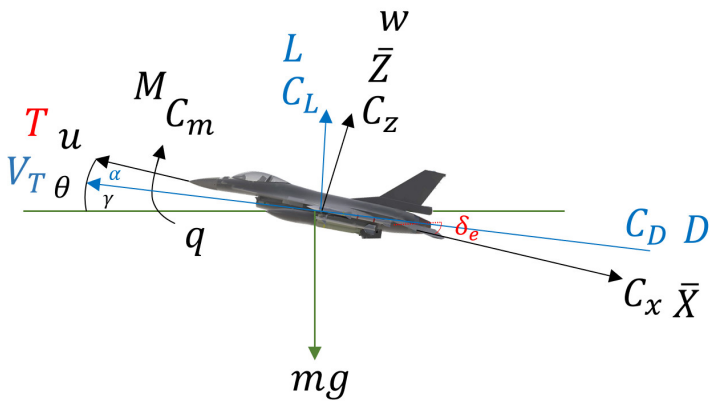


Figure 3.4 – The 3-DOF F-16 Reference Axes, Force and Moment Vectors for the Longitudinal Dynamics

3.5.2 The F-16 3-DOF Simulator

For this study we use a nonlinear 3 degrees-of-freedom (3-DOF) simulation of an F-16 aircraft derived from NASA wind tunnel data from the study by [42]. The model is constructed in reference to the work by, [44, 43] and the equations of motion are derived from, [50, 49, 42]. The simulator 3.5 is constructed with actuator dynamics for the elevator but a constant thrust input for trimmed steady level flight. A stick input includes pre-programmed test inputs. The learning component processes batches of trajectory data through an online convex optimization, which in turn provides parameters to the CCM optimization certificate. The certificate ensures a stable policy is passed to the adaptive controller. The full nonlinear equations of motion for the simulator were previously discussed Eq. (5.1) with aircraft properties for the model shown in Table 5.2.2.

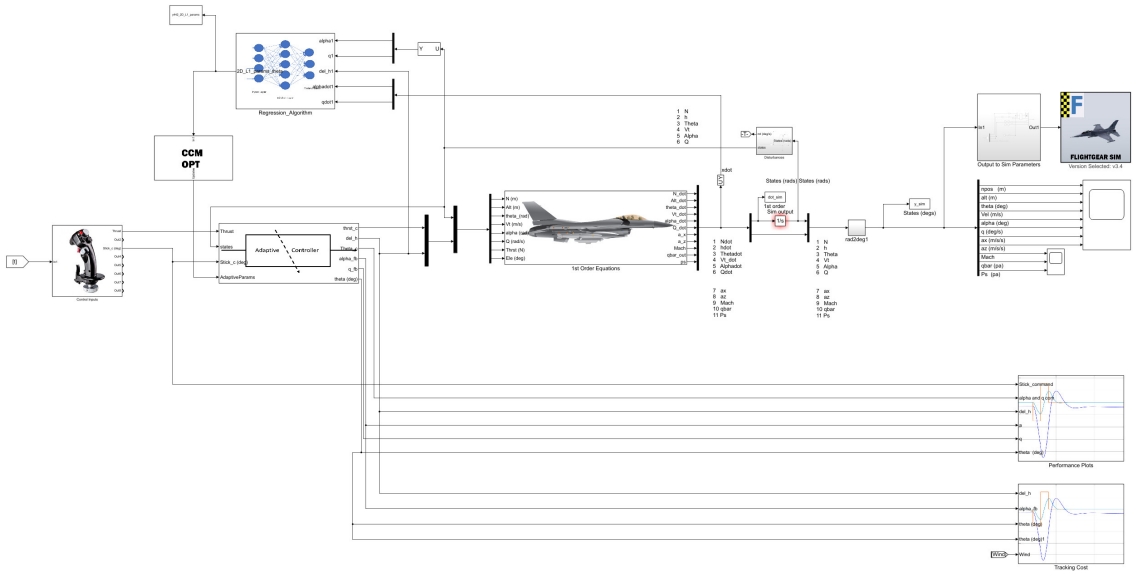


Figure 3.5 – F-16 Longitudinal Dynamics Learning Simulator

Table 3.3 Aircraft Properties

Parameter	Symbol	Value	Units
Mass	m	9295.44	kg
Wing Area	S	27.87	m^2
Mean Aerodynamic Chord	$\bar{c}$	3.45	m
Moment of Inertia	I_{yy}	75673.6	$kg.m^2$
Neutral Point Location	x_p	$0.35\bar{c}$	m
CG location	x_{cg}	$0.3\bar{c}$	m

The aerodynamic force coefficients in the body axis are C_{Z_t}, C_{X_t} and the aerodynamic moment coefficient is C_{m_t}. All coefficients are functions of the states α and q and the elevator deflection angle δ_h. Values for the coefficents are taken from the high-fidelity model tables listed in [42].

$$
\begin{aligned}
C_{X_t}(\alpha, q, \delta_h) &= C_X(\alpha, q, \delta_h) + \frac{\bar{c}q}{2V} C_{X_q} \\
C_{Z_t}(\alpha, q, \delta_h) &= C_Z(\alpha, q, \delta_h) \frac{\bar{c}q}{2V} C_{Z_q} \\
C_{m_t}(\alpha, q, \delta_h) &= C_m(\alpha, q, \delta_h) + C_{Z_t}(x_{cgr} - x_{cg}) + \frac{\bar{c}q}{2V} C_{m_q}
\end{aligned}
\tag{3.34}
$$

The aerodynamic force coefficents C_Z, C_X and pitching moment coefficient C_M are functions of the the states α, q where higher order terms, $\dot{\alpha}, \dot{q}$, are neglected. The simulation also calculates the coefficients as a nonlinear function of the control input from the elevator δ_h. Implemented into the model are lookup tables with breakpoints taken from the NASA wind tunnel study in α and δ_h. Figure 3.6 gives the nonlinear coefficient data for the high-fidelity model for the full angle of attack range from -20 degrees to 90 degrees. It can be seen that from 0 to 20 degrees the aerodynamic derivatives are almost linear and we can expect the approximating model to fit the dynamics equations well unless an aerodynamic modification is made to the lookup tables. The lookup tables interpolate linearly between breakpoints and can be directly manipulated in the system model more easily than the original C-program developed by [44]. Initiation of the simulated model uses the trim settings shown in Table 5.2.3.

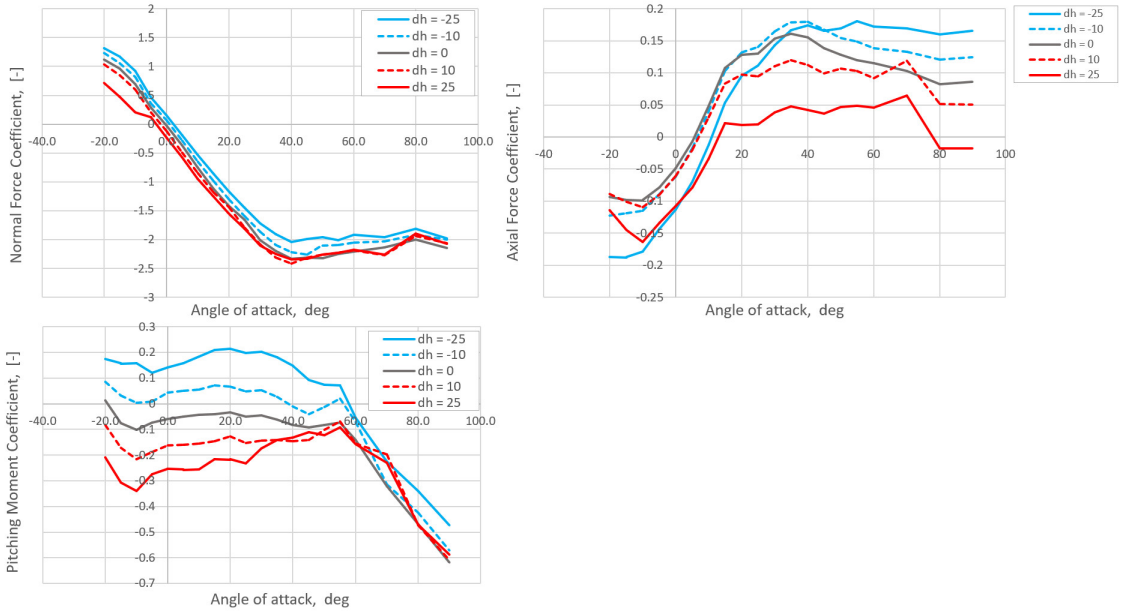


Figure 3.6 – Aerodynamic force coefficient wind tunnel data as a function of angle of attack and elevator deflection angle

Table 3.4 Simulation Trim Settings

Parameter	Symbol	Value	Units
Altitude	h	$4572\ (15,000)$	$m\ (ft)$
Velocity	V_T	$152.4\ (500)$	$m/s\ (ft/s)$
Elevator Trim	$\delta_{h_{tr}}$	-4.55	deg
AoA Trim	α_{tr}	4.47	deg
Pitch Rate Trim	q_{tr}	0	deg/s
Thrust Trim	F_T	9433	N

The control surface limits are given in the Table 3.5 for minimum and maximum elevator deflection and rate limits. Figure 3.7 shows the modified aerodynamic coefficients to represent arbitrary damage to an upper surface of the aircraft wings causing an increase in drag between 0 and 20 degrees of angle of attack. The increase in drag causes a negative pitching moment and a decrease in lift generated by the wings. It is envisaged that many configurations can be tested in future experiments but this first analysis is a proof of concept.

Table 3.5 Control Limits

Control	units	Min	Max	rate limits
Elevator	deg	-25	25	+- 60 deg/s
Thrust	N	4448	85000	-

3.5.3 Controller Design Specification Derived from MIL-STD-C-F-8785C

The F-16 falls under the Class-IV category for military specifications for high maneuverability aircraft. For this study, we consider the aircraft to be in cruising or mission flight phases classified as CAT A - Precision Combat or CAT B - Non-Precision Cruise but stable and gradual flight. The Flying Qualities are then further broken down into levels depending on the capability of the handling and maneuverability. Level 1 being mission-

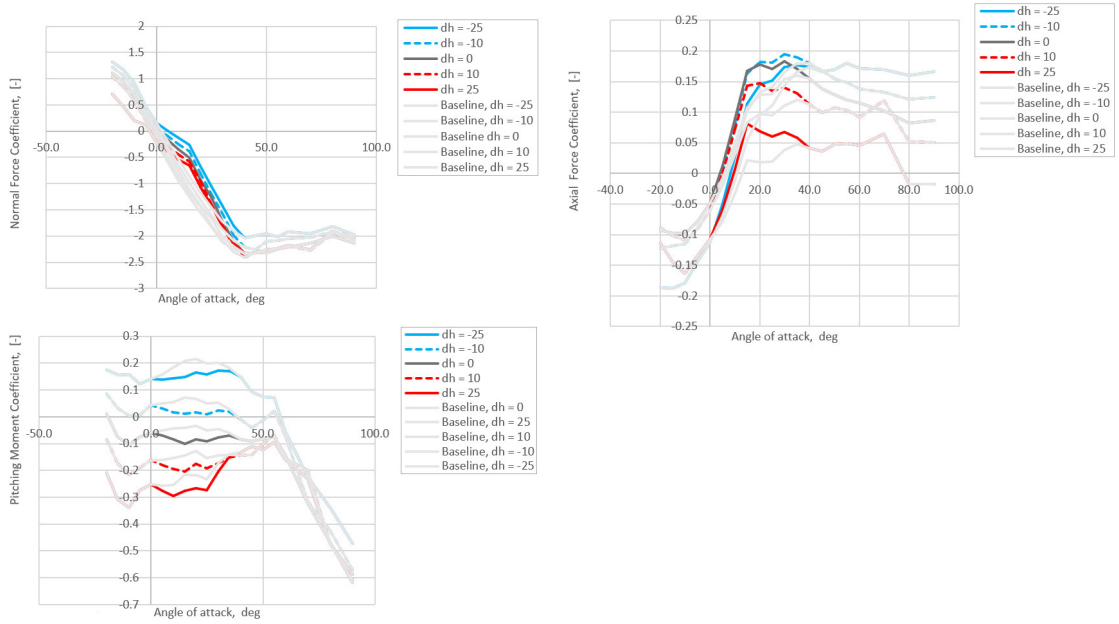


Figure 3.7 – Aerodynamic force coefficient data modified to simulate damage as a function of angle of attack and elevator deflection angle

level precision flying, Level 2 being an increased pilot workload but still able to carry out a mission, and Level 3 which has an excessive workload with some loss in performance but still safe to fly. We use Requirements 3.2.2.1.1 and 3.2.2.1.2 relating to the angle of attack response for controls free and controls fixed flying at a constant speed. For this study we use simple bounding box limits to specify the required damping ratio and natural frequency for Level 1 flying qualities, Table 3.6. Since natural frequency limits are defined also by

Table 3.6 CAT A Level 1 Flying Qualitites for Class IV Aircraft

Parameter	Lower limit	Upper Limit
$\omega_{n_{sp}}^2 (\frac{n}{\alpha})^{-1}$ (rad/s)	0.28	3.6
$\omega_n, \frac{n}{\alpha} = 14\ (rad^{-1})$	2	7
ζ_{sp} (-)	0.35	1.3

3.6 LQR technique

For a linear system given by Eqn. 3.35, where the system is time invariant.

$$\dot{x}(t) = Ax(t) + Bu(t) \qquad x(0) = x_0 \tag{3.35}$$

A cost function is defined as a quadratic sum of a weighted squared term in the state variable and a weighted squared term in the control input over an infinite time horizon. The weighting matrices Q and R respectively give the relative importance of the input and state dimensions with the default values being the identity matrix.

$$\min_{u,x} J(u) = \int_0^{+\infty} (x^T(t)Qx(t) + u^T(t)Ru(t))dt \tag{3.36}$$

The minimum of the cost function is given by the quadratic initial state solution weighted by the optimal P matrix, Eqn. 3.37

$$J_{opt} = {x_0}^T P x_0 \tag{3.37}$$

Where the solution to the P matrix is found from the Algebraic Ricatti Equation, Eqn.3.38

$$A^T P + PA + Q - PBR^{-1}B^T P = 0 \tag{3.38}$$

The optimal control gain is given by Eqn. 3.39

$$K = R^{-1}B^T P \tag{3.39}$$

The optimal control input is a state feedback controller of the following form, Eqn. 3.40.

$$u(t) = -Kx(t) \tag{3.40}$$

The Q and R matrices must be tuned to find the optimal feedback control policy. For

this study we fixed the R value to 1 and adjust the diagonal elements of the Q matrix to find a weighting that most closely matches the design condition for the closed loop reference model.

3.7 The CCM Technique

In contraction theory, the evolution of an initial virtual displacement, δ_{x_0}, of a pair of flow lines or trajectories, Figure 3.8, is certified to be contracting exponentially by analysing the negative definitness of the Jacobian of the system by way of a metric tensor $M(x,t)$. If a metric is found for the system then the system is provably incrementally stable by the contraction of all pairwise trajectories.

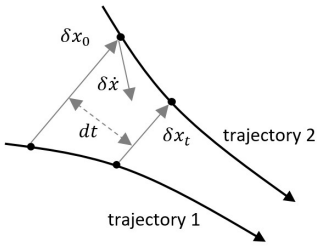


Figure 3.8 – Contraction theory diagram, Adapted from [32].

The theory has been developed into a Control Contraction Metric (CCM) for control affine systems in [35] where a differential control input is added as in Eqn. 3.41.

$$\delta \dot{x} = A(x,u,t)\delta x + B(x,t)\delta u \tag{3.41}$$

The A matrix is given as: $A := \frac{\partial f}{\partial x} + \sum_{i=1}^{m} \frac{\partial b_i}{\partial x} u_i$,

A Metric $M(x,t)$ is a CCM if there is a controller $u(x,x^*,u^*)$ for all δ_x for a given $\lambda \geq 0$ if the following inequality holds:

$$\frac{d}{dt}(\delta_x^T M(x)\delta_x) \leq -2\lambda \delta_x^T M(x)\delta_x \tag{3.42}$$

This proves that an initial displacement in x will converge exponentially to have zero tracking error as time tends to infinity:

$$\|x(t) - x^*(t)\| \leq \|x(0) - x^*(0)\| e^{-\lambda t} \tag{3.43}$$

The inequality can be expressed as a matrix inequality:

$$\dot{M} + M\widehat{(A + BK)} + 2\lambda M \prec 0 \tag{3.44}$$

Where $K = \frac{\partial u}{\partial x}$ and $\dot{M} = \sum_{i=1}^{n} \frac{\partial M}{\partial x^i} \dot{x}^i$,

A key result in contraction theory is the fundamental theorem that if the following conditions exist:

$$B_\perp^T \left(-\partial_f W(x) + \widehat{\frac{\partial f(x)}{\partial x} W(x)} + 2\lambda W(x) \right) B_\perp \prec 0 \tag{3.45}$$

$$B_\perp^T \left(\partial_{b_j} W(x) - \widehat{\frac{\partial b_j(x)}{\partial x} W(x)} \right) B_\perp = 0, j = 1, ..., m, \tag{3.46}$$

where $B_\perp^T$ is an annihilator matrix of $B(x)$ satisfying $B^T B = 0$, and $W(x) = M(x)^{-1}$ is the dual metric, then there exists a differential tracking controller $k(x, x^*, u^*)$ that stabilises the closed loop system with a contraction rate λ under the metric $M(x)$. The solution to the CCM can also be found from the alternative form using the dual metric and a scalar valued gain function $\rho(x)$ as a Linear Matrix Inequality (LMI).

$$-\partial_f W(x) + \widehat{\frac{\partial f(x)}{\partial x} W(x)} - \rho(x) B(x) B(x)^T + 2\lambda W(x) \prec 0 \tag{3.47}$$

Or in the more concise notation as 3.48.

$$-\dot{W} + W A^T + A W - \rho B B^T < -2\lambda W \tag{3.48}$$

The inequality is a convex optimisation in $W(x,t)$ and $\rho(x,t)$.

The LMI is an infinite dimensional problem and can be made finite-dimensional by a suitable approximating function, enforcing constraints by gridding over a region of the state space or as a sum-of-squares optimisation problem [38]. A description of contraction metrics using the sum-of-squares technique is descirbed in Aylward et al. [6].

The resulting $W(x)$ and $\rho(x)$ form the stabilising differential control equation:

$$u(t) - u^*(t) = -K(x(t) - x^*(t)) \tag{3.49}$$

$$\textit{where} \quad K = -\frac{1}{2}\rho(x)B^T W(x)^{-1} \tag{3.50}$$

3.7.1 Linear Time Invariant Control Contraction Metric

The definition of the CCM can be alternatively defined as a differential control Lyapunov function (dCLF), 3.51 defining incremental stability. Where the quadratic function is weighted by the positive definite symmetric matrix M and the derivative is described in 3.52 and 3.53.

$$V(\delta_x) = \delta_x^T M \delta_x \tag{3.51}$$

$$\dot{V}(\delta_x) = \frac{d}{dt}(\delta x^T M \delta x) = \frac{d(\delta x^T)}{dt} M\delta x + \delta x^T M \frac{d(\delta x)}{dt} \tag{3.52}$$

$$= (A\delta x + B\delta u)^T M \delta x + \delta x^T M (A\delta x + B\delta u) \tag{3.53}$$

Which can be re-arranged as 3.54

$$= \delta x^T (A^T M + MA)\delta x + \delta u^T B^T M \delta x + \delta x^T M B \delta u \tag{3.54}$$

And introducing a differential feedback gain term for $\delta_u = K\delta_x$ we get 3.55.

$$= \delta x^T (A^T M + MA)\delta x + \delta x^T K^T B^T M \delta x + \delta x^T M B K \delta x \tag{3.55}$$

Then re-arranging for left and right factors of δ_x we have 3.56.

$$= \delta x^T (A^T M + MA + K^T B^T M + MBK)\delta x \tag{3.56}$$

This can be expressed as an LMI, where if the LMI is negative definite it proves the system stable 3.57. But the LMI is non-convex because the decision variables in M and K are factored together creating a bi-linear equation.

$$A^T M + K^T B^T M + MA + MBK \prec 0 \tag{3.57}$$

To make the decision variables linearly independant first we multiply both sides by W the dual metric of M, 3.58.

$$WA^T + WK^T B^T + AW + BKW \prec 0 \tag{3.58}$$

Then substituting $KW = L$ the LMI becomes 3.59.

$$WA^T + AW + L^T B^T + BL \prec 0 \tag{3.59}$$

This form of LMI is shown to be equivalent to 3.60 via Finsler's Lemma 3.A. Where ρ is a positive scalar.

$$WA^T + AW - \rho BB^T \prec 0 \tag{3.60}$$

Another equivalent form of the LMI is also given 3.61. Where for any $B_\perp$, the orthogonal complement of B, of maximum rank such that $B_\perp^T B_\perp = 0$.

$$B_\perp \left(WA^T + AW\right) B_\perp \prec 0 \tag{3.61}$$

Equation 3.60 is exponentially stable in λ if the inequality is made strict by being more negative definite than $-2\lambda W$, 5.3.

$$AW + WA^T - \rho BB^T + 2\lambda W \prec 0 \tag{3.62}$$

Since the equation is linear in W the ρ term can also be absorbed into the decision variables of W. It was also shown in [34] that by adding diagonal weighting matrices R and Q into the LMI, 3.60, the Riccati inequality is recovered, 3.63.

$$AW + WA^T - BR^{-1}B^T + WQW = 0 \tag{3.63}$$

And by multiplying either side of the equation by $M = W^{-1}$ the exact form of the Riccati equation is found as in the LQR formulation, Equation 3.64.

$$A^T M + MA + Q - MBR^{-1}B^T M = 0 \tag{3.64}$$

The optimal control input found using the metric in the LQR formula is a state feedback controller of the following form, Eqn. 3.65.

$$u(t) = -R^{-1}B^T M x(t) \tag{3.65}$$

In the CCM formulation where a static metric is found for a given contraction rate λ, the control input is of the form, Eq. 5.4.

$$u(t) = -0.5\rho B^T W^{-1}\delta x(t) \tag{3.66}$$

Which is equivalent to a differential feedback controller with static gain matrix K:

$$u(t) = -K\delta x(t) \tag{3.67}$$

From which the relation between metric and feedback gain matrix becomes:

$$K = \frac{1}{2}B^T W^{-1} = \frac{1}{2}B^T M \tag{3.68}$$

3.7.2 Robustness of a CCM Controller

If the disturbance $d(t)$ is bound by $\|d(t)\| \leq \bar{d}$ and the tracking controller under a metric is bound by $\underline{m}I \leq M(x) \leq \bar{m}I$ then the trajectory tracking error can be shown to have a worst case theoretical bound given by:

$$\|x(t) - x^*(t)\| \leq \frac{R_0}{\sqrt{\underline{m}}}e^{-\lambda t} + \sqrt{\frac{\bar{m}}{\underline{m}}}\frac{\bar{d}}{\lambda}(1 - e^{-\lambda t}) \tag{3.69}$$

where, $R_0 = \int_{x(0)}^{x^*(0)} \sqrt{\delta_x^T M(x)\delta_x}$.

R_0 is the initial displacement, defined by a geodesic in the metric between two trajectories. For a static metric the displacement is a straight line in Euclidean space, $R_0 = \sqrt{(x(0) - x^*(0))^T M (x(0) - x^*(0))}$. A tradeoff of the contraction rate λ and the condition number was briefly studied in Manchester et al, [35]. The tracking error converges to a steady-state value defined by the maximum disturbance term, the square root of the condition number, divided by the contraction rate. The exponential bound on the tracking error effectively creates a robustness tube, [52], Figure 3.9. A smaller condition number or a larger decay rate will lead to a smaller steady-state tracking error, or a more robust controller that is less affected by disturbances. From the definition of the tracking error bound, it can be seen that an appropriate cost function for the tracking error yields the condition number divided by the square of the contraction rate. This cost function represents a quadratic cost over a time horizon of the disturbances, which will be utilized later. Additionally, the equation demonstrates that the initial displacement, normalized by the square root of the minimum eigenvalue of the metric, is exponentially forgotten as a worst-case scenario and provides a conservative estimate of the tracking error.

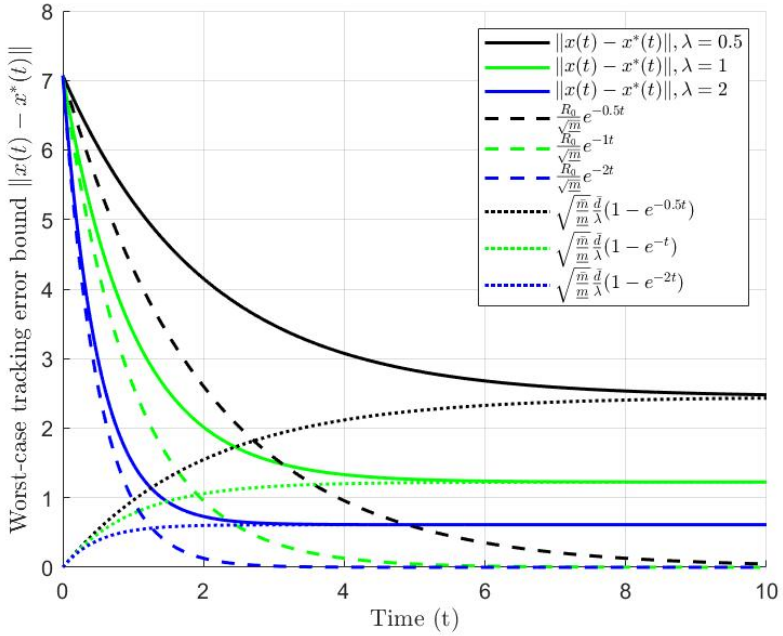


Figure 3.9 – Example of Worst Case Robustness Tube Bounds for: $\left(\bar{d} = 1, \bar{m} = 3, \underline{m} = 2, R_0 = 10\right)$ and $(\lambda = 0.5, 1, 2)$

3.7.3 CCM Illustrative Example

A simple second order system in one degree of freedom can be represented in differential dynamics form as 3.70

$$\begin{bmatrix} \dot{\delta}_{x_1} \\ \dot{\delta}_{x_2} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\omega_n^2 & -2\zeta\omega_n \end{bmatrix} \begin{bmatrix} \delta_{x_1} \\ \delta_{x_2} \end{bmatrix} + \begin{bmatrix} 0 \\ -\omega_n^2 \end{bmatrix} \begin{bmatrix} \delta_u \end{bmatrix} \tag{3.70}$$

Where the closed loop dynamics are given by 3.71:

$$\begin{bmatrix} \dot{\delta}_{x_1} \\ \dot{\delta}_{x_2} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\omega_n^2 - \omega_n^2 K_1 \delta_{x_2} & -2\zeta\omega_n - \omega_n^2 K_2 \delta_{x_2} \end{bmatrix} \begin{bmatrix} \delta_{x_1} \\ \delta_{x_2} \end{bmatrix} + \begin{bmatrix} 0 \\ -\omega_n^2 \end{bmatrix} \begin{bmatrix} \delta_v \end{bmatrix} \tag{3.71}$$

In the closed-loop dynamics, the feedback gain terms K_1 and K_2 are derived from the certified control contraction metric (CCM). The term δ_v represents an exogenous input or disturbance. The natural contraction of the system is determined by the eigenvalues of the Jacobian matrix associated with the drift dynamics. If the drift dynamics do not exhibit natural contraction, a control input must be applied to enforce negative definiteness of the closed-loop eigenvalues, thereby guaranteeing stability. In a recent publication by Manchester et al. [38], it was demonstrated that a two-dimensional metric for a second-order system can be analytically constructed with an infinite number of values for the elements of a static dual metric, provided they satisfy the condition for a scalar metric as shown in Eqn. 3.72.

$$w_{12} < -\lambda w_{11} \tag{3.72}$$

Where the dual metric is defined as a positive definite symmetric matrix with elements:

$$W = \begin{bmatrix} w_{11} & w_{12} \\ w_{12} & w_{22} \end{bmatrix} \tag{3.73}$$

And the LMI condition to prove exponential stability is given as:

$$AW + WA^T - BB^T \prec -2\lambda W \tag{3.74}$$

Since w_{11} is positive w_{12} must be negative. The metric $M = W^{-1}$ must have all positive entries due to the inverse properties of matrices. To illustrate the use of the metric and the contraction rate in the CCM technique we provide some examples for a 2D dual metric $W = [1 \quad -0.3; -0.3 \quad 1]$. The metric gives the primary axis and scaling of contraction as a transformation of the state coordinates. The contraction rate λ then defines the exponential change in the systems flow field vectors in the metric.

$$\left(\begin{bmatrix} 0 & 1 \\ -1 & -1 \end{bmatrix} \begin{bmatrix} 1 & -0.3 \\ -0.3 & 1 \end{bmatrix} + \begin{bmatrix} 1 & -0.3 \\ -0.3 & 1 \end{bmatrix} \begin{bmatrix} 0 & -1 \\ 1 & -1 \end{bmatrix} - \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix} \right) \leq -2\lambda \begin{bmatrix} 1 & -0.3 \\ -0.3 & 1 \end{bmatrix} \tag{3.75}$$

And reduces to:

$$\left(\begin{bmatrix} -0.6 & 0.3 \\ 0.3 & -2.4 \end{bmatrix} \leq -\lambda \begin{bmatrix} 2 & -0.6 \\ -0.6 & 2 \end{bmatrix} \right) \tag{3.76}$$

Where subtracting the right hand side on the left and taking the determinant yields the characteristic equation in λ, $1.35 - 5.64\lambda + 3.64\lambda^2$ with real roots $\lambda_1 = 1.25$ and $\lambda_2 = 0.29$. Since we are interested in the worst case bound, the slowest decay rate for the system, the metric certifies the system as stable and contracting in $\lambda = 0.295$.

Certification of the CCM for a given contraction rate, λ, establishes that the system's contraction, facilitated by closed-loop feedback, exceeds the rate of contraction due to

		Drift Dyn Eigs		Closed Loop Eigs			M Eigs		Feedback Gains	
Ex	λ	eig1	eig2	eig1	eig2	CondN	eig1	eig2	K_1	K_2
1	2.15	$-2.1+2.14i$	$-2.1-2.14i$	-2.15	-2.15	20.67	0.09	1.95	-0.21	-0.10
2	0.55	$-0.5+1.94i$	$-0.5-1.94i$	$-0.55+1.94i$	$-0.55-1.94i$	15.18	0.18	0.83	-0.055	-0.1
3	2.05	1	-2	-2.05	-2.05	44.27	0.18	33.02	-6.36	-3.1

Table 3.7 Examples of Metric Lower Bound Certification and Feedback Parameters

the natural drift dynamics. If a system is already contracting, the contraction rate defines the minimum bound. To emphasize this property, Table 3.7 presents examples of systems where a metric and a contraction rate have been identified to certify a feedback gain matrix where the example system dynamics are given in 3.77.

$$\begin{aligned}
&\text{Ex 1} \quad A_1 = \begin{bmatrix} 0 & 1 \\ -9 & -4.2 \end{bmatrix}, \quad B_1 = \begin{bmatrix} 0 \\ -1 \end{bmatrix}, \quad W_1 = \begin{bmatrix} 0.06 & -0.14 \\ -0.14 & 0.49 \end{bmatrix}, \quad M_1 = \begin{bmatrix} 74.93 & 21.84 \\ 21.84 & 8.40 \end{bmatrix} \\
&\text{Ex 2} \quad A_2 = \begin{bmatrix} 0 & 1 \\ -4 & -1 \end{bmatrix}, \quad B_2 = \begin{bmatrix} 0 \\ -1 \end{bmatrix}, \quad W_2 = \begin{bmatrix} 0.10 & -0.269 \\ -0.27 & 1.19 \end{bmatrix}, \quad M_2 = \begin{bmatrix} 23.2 & 5.2 \\ 5.2 & 2 \end{bmatrix} \\
&\text{Ex 3} \quad A_3 = \begin{bmatrix} 0 & 1 \\ 2 & -1 \end{bmatrix}, \quad B_3 = \begin{bmatrix} 0 \\ -1 \end{bmatrix}, \quad W_3 = \begin{bmatrix} 0.02 & -0.04 \\ -0.04 & 0.22 \end{bmatrix}, \quad M_3 = \begin{bmatrix} 125.3 & 21.8 \\ 21.8 & 8.4 \end{bmatrix}
\end{aligned} \tag{3.77}$$

In the table 3.7 it can be seen that the closed loop certified contraction rate is the same as that of the the closed loop eigenvalues as a complex pole pair due to the system being underdamped for the contraction rate defined. For systems that are already contracting and hence exhibit natural stability, examples 1 and 2, the eigenvalues of the drift dynamics are negative by the real part of the complex pole pair slightly less than the certified minimum contraction under control inputs, certified by feedback in the metric. This is because the CCM explicitly certifies contraction for feedback control so natural contraction becomes the minimum bound.

3.7.4 Linear Time Invariant Analysis

In this analysis, we describe the linearized dynamics of an aircraft in terms of the state-space representation, as well as the closed-loop control system for the aircraft's pitch dynamics. The equations of motion, state-space representation, and transfer functions are derived to characterize the aircraft's short period mode. The stability and handling

qualities of the short period mode can be described by the natural frequency and time constant, which are used as design criteria.

State Space to Transfer Function Formulas

The linearized dynamics of the aircraft where the outputs are assumed equal in magnitude to the states, ie. the C matrix is identity and D matrix zero, are given by:

$$\dot{x} = Ax + Bu$$

Where $u = v - Kx$ describes a nominal control input v and feedback term $-Kx$ with feedback matrix K. The system can be written as:

$$\dot{x} = Ax + B(v - Kx)$$

And then converting into the Laplace frequency domain the following formulas can be used to convert the system into a transfer function:

$$sX(s) = \Big[A - BK\Big] X(s) + BV(s)$$
$$X(s) = \Big[Is - A + BK\Big]^{-1} BV(s)$$

Algebraic form of the state space to transfer function

The state-space representation of the aircraft's pitch dynamics is given as:

$$\begin{bmatrix} \dot{\alpha} \\ \dot{q} \end{bmatrix} = \begin{bmatrix} Z_\alpha & Z_q \\ M_\alpha & M_q \end{bmatrix} \begin{bmatrix} \alpha \\ q \end{bmatrix} + \begin{bmatrix} B_\alpha \\ B_q \end{bmatrix} u \tag{3.78}$$

Where $Z_\alpha, Z_q, M_\alpha, M_q$ are the aerodynamic force and moment coefficients. The control input u is derived from the feedback law:

$$u = \delta_h - \begin{bmatrix} K_\alpha & K_q \end{bmatrix} \begin{bmatrix} \alpha \\ q \end{bmatrix} \tag{3.79}$$

The closed-loop dynamics can then be written as:

$$\begin{bmatrix} \dot{\alpha} \\ \dot{q} \end{bmatrix} = \begin{bmatrix} Z_\alpha - B_\alpha K_\alpha & Z_q - B_\alpha K_q \\ M_\alpha - B_q K_\alpha & M_q - B_q K_q \end{bmatrix} \begin{bmatrix} \alpha \\ q \end{bmatrix} + \begin{bmatrix} B_\alpha \\ B_q \end{bmatrix} \delta_h \tag{3.80}$$

The corresponding Laplace transform can be expressed as:

$$s \begin{bmatrix} \alpha(s) \\ q(s) \end{bmatrix} = \begin{bmatrix} Z_\alpha - B_\alpha K_\alpha & Z_q - B_\alpha K_q \\ M_\alpha - B_q K_\alpha & M_q - B_q K_q \end{bmatrix} \begin{bmatrix} \alpha(s) \\ q(s) \end{bmatrix} + \begin{bmatrix} B_\alpha \\ B_q \end{bmatrix} \delta_h(s) \tag{3.81}$$

The transfer function relating the states to the control input is given by:

$$\begin{bmatrix} \alpha(s) \\ q(s) \end{bmatrix} = \begin{bmatrix} s - Z_\alpha + B_\alpha K_\alpha & -Z_q + B_\alpha K_q \\ -M_\alpha + B_q K_\alpha & s - M_q + B_q K_q \end{bmatrix} \begin{bmatrix} \alpha(s) \\ q(s) \end{bmatrix}^{-1} \begin{bmatrix} B_\alpha \\ B_q \end{bmatrix} \delta_h(s) \tag{3.82}$$

To obtain the individual transfer functions for the pitch angle and pitch rate, we can derive the following expressions:

$$\frac{1}{s^2 + s(B_\alpha K_\alpha + B_q K_q - Z_\alpha - M_q) + K_\alpha(Z_q B_q - M_q B_\alpha) + K_q(B_\alpha M_\alpha - Z_\alpha B_q) + Z_\alpha M_q - M_\alpha Z_q} \cdot$$

$$\cdot \begin{bmatrix} s - M_q + B_q K_q & Z_q - B_\alpha K_q \\ M_\alpha - B_q K_\alpha & s - Z_\alpha + B_\alpha K_\alpha \end{bmatrix} \begin{bmatrix} B_\alpha \\ B_q \end{bmatrix} \delta_h$$

For the short period mode, we can represent the transfer functions as:

$$\frac{\alpha(s)}{\delta_h(s)} = \frac{B_\alpha(s - M_q + B_q K_q) + B_q(Z_q - B_\alpha K_q)}{s^2 + s(B_\alpha K_\alpha + B_q K_q - Z_\alpha - M_q) + K_\alpha(Z_q B_q - M_q B_\alpha) + K_q(B_\alpha M_\alpha - Z_\alpha B_q) + Z_\alpha M_q - M_\alpha Z_q}$$

$$\frac{q(s)}{\delta_h(s)} = \frac{B_q(s - Z_\alpha + B_\alpha K_\alpha) + B_\alpha(M_\alpha - B_q K_\alpha)}{s^2 + s(B_\alpha K_\alpha + B_q K_q - Z_\alpha - M_q) + K_\alpha(Z_q B_q - M_q B_\alpha) + K_q(B_\alpha M_\alpha - Z_\alpha B_q) + Z_\alpha M_q - M_\alpha Z_q}$$

The two transfer functions represent a second order characteristic for angle of attack and pitch rate responses to elevator inputs given by:

$$\frac{\alpha(s)}{\delta_h(s)} = \frac{K_\alpha(s + \frac{1}{T_\alpha})}{s^2 + s(2\zeta\omega_n) + \omega_n^2}$$

$$\frac{q(s)}{\delta_h(s)} = \frac{K_q(s + \frac{1}{T_{\theta_2}})}{s^2 + s(2\zeta\omega_n) + \omega_n^2}$$

Where the stability and handling qualities for the short period mode can be described by ω_n and T_{θ_2} and used as design criteria. ω_n is the short period natural frequency and represents the oscillation rate of the aircraft. It is a measure of how quickly the aircraft responds to a disturbance in pitch. A higher ω_n indicates a faster response, while a lower ω_n indicates a slower response. The time constant T_{θ_2} is associated with the damping of the short period mode. It represents the time it takes for the pitch rate to decay to approximately 37% of its initial value in response to a disturbance. A smaller T_{θ_2} indicates faster damping of the short period mode.

The equations for damping ratio and the natural frequency from the characteristic equation becomes:

$$2\zeta\omega_n = B_\alpha K_\alpha + B_q K_q - Z_\alpha - M_q$$
$$\omega_n^2 = K_\alpha(Z_q B_q - M_q B_\alpha) + K_q(B_\alpha M_\alpha - Z_\alpha B_q) + Z_\alpha M_q - M_\alpha Z_q$$

Looking at the design parameters the only control variables are the gain terms where the other parameters are fixed for an operating point. We have therefore:

$$\zeta = \frac{C_1 K_\alpha + C_2 K_q - C_3}{2\omega_n}$$
$$\omega_n^2 = K_\alpha C_4 + K_q C_5 + C_6$$

The solutions of the characteristic equation are the poles for the system transfer function and can be found using the general quadratic solution:

$$p_{1,2} = -\zeta\omega_n \pm \frac{\sqrt{(2\zeta\omega_n)^2 - 4(\omega_n^2)}}{2} \tag{3.83}$$

If $\omega_n^2 > (\zeta\omega_n)^2$, the discriminant inside the square root is negative, and the roots are complex conjugates. In this case, the system exhibits underdamped behavior, characterized by oscillations around the equilibrium point with decreasing amplitude. The system poles are described by 3.84.

$$p1 = -\zeta\omega_n + j\omega_n\sqrt{1-\zeta^2} \tag{3.84}$$
$$p2 = -\zeta\omega_n - j\omega_n\sqrt{1-\zeta^2} \tag{3.85}$$

The optimum underdamped system has a damping ratio of $\frac{1}{\sqrt{(2)}} = 0.707$ or approximately 0.7 where the overshoot is minimised and the response is non-oscillatory. If

$\omega_n^2 \leq (\zeta\omega_n)^2$, the roots are real, and the system is either overdamped or critically damped. The poles for each case are:

Critically damped ($\zeta = 1$):

$$p1 = -\omega_n \tag{3.86}$$

$$p2 = -\omega_n \tag{3.87}$$

Overdamped ($\zeta > 1$):

$$p1 = -\zeta\omega_n + \omega_n\sqrt{\zeta^2 - 1} \tag{3.88}$$

$$p2 = -\zeta\omega_n - \omega_n\sqrt{\zeta^2 - 1} \tag{3.89}$$

In both critically damped and overdamped cases, the system converges to the equilibrium point without oscillations, but the rate of convergence is faster in the critically damped case compared to the overdamped case.

For our design condition of $\zeta = 0.7$ and $\omega_n = 3$ the poles become the pair:

$$p1 = -2.1 + j2.142 \tag{3.90}$$

$$p2 = -2.1 - j2.142 \tag{3.91}$$

For second order systems there are also analytical solutions for the overshoot, rise time, settling time

3.8 Results and Discussion

In this section we give results for the linear model approximations learned from the simulation for the baseline and damaged cases. We then examine the feedback gains employed

by each control optimization method. To compare the performance of each controller for the baseline and damaged aircraft, we conduct an analysis of the longitudinal dynamics in response to an initial disturbance. Since the contraction metric's performance is measured by its ability to return the state trajectory back to a nominal controlled trajectory following a disturbance, an initial impulse disturbance of 5 degrees in pitch and 5 degrees per second in pitch rate is selected as the test input. This choice represents a sensible range for the dynamics of the short period mode.

3.8.1 The Linear Regression

The linear model approximation was trained on a 40-second time horizon using a sinusoidal stick input to simulate a pilot's command. The magnitude of the angle of attack (α) and pitch rate (q) were constrained within the intervals [-5, 15] and [-10, 10], respectively. The time history was sampled at 0.01-second intervals, yielding a total of 4,000 training examples. The learned parameters for θ, which represent the linear approximations in this study, are shown in Table 3.8. The following results show the performance of the linear model approximation when trained over the specified time horizon and input constraints.

Table 3.8 Learned Parameters for Library Regressors for the Polynomial Model Approximation of the Short Period Dynamics for the Baseline and Damaged Airframe

	Baseline Aircraft				Damaged Aircraft		
Θ	Φ_{Poly}	$\dot{\alpha}$	$\dot{q}$		Φ_{Poly}	$\dot{\alpha}$	$\dot{q}$
1	α	-0.7266	-1.3476		α	-0.584	-1.595
2	q	0.9678	-0.7799		q	0.588	0.137
3	δ_h	-0.0895	-6.0169		δ_h	-0.5515	-3.285

In order to assess the performance of the linear dynamics approximation for the given system, we utilised four sinusoidal tracking trajectories as test data, as depicted in Fig. 3.10. These trajectories were excited within a nonlinear simulation of the F-16, and the resulting data was compared against the integration using the Runge-Kutta (4,5) method

for solving ordinary differential equations (ODEs) in Matlab, employing both the learned model and the forcing function utilised during the simulation. Root mean square errors were analyzed for each trajectory and can be found in Table 3.9.

While the trajectory sample size is insufficient for a robust statistical analysis, it can be seen that the short-period mode is well-approximated for the majority of the time horizon. The most significant deviations are observed during instances of substantial change in angle of attack (alpha). This discrepancy may be attributed to the linear model's inability to fully capture higher-order and trigonometric terms in alpha. In the trajectory plots, solid lines represent the actual alpha and pitch rate (q) trajectories from flight data, while dotted lines indicate the model predictions.

It is important to note that noise, a considerable factor influencing model approximation success, has not been incorporated into the training data. Specifically, for the baseline and damaged aircraft scenarios considered in this study, process and measurement noise have not been taken into account, and they are deemed outside the scope of this work. As estimation, averaging, and filtering are not the primary focus of this study, they have been excluded from this initial pipeline development. Furthermore, while the L1-norm-based regression technique is recognised for its noise handling capabilities due to promoting sparse features in regression, its relevance to the study is limited at this stage, as linear approximations are the primary focus.

Table 3.9 Percentage Errors of Learned Model Tracking Tests

Testing	B L*	B NL	T1 L**	T1 NL	T2 L	T2 NL	T3 L	T3 NL	T4 L	T4 NL
α (rmse) /10%	1%	1%	7%	8%	17%	13%	2%	1%	4%	3%
q (rmse) /10%	2%	1%	6%	7%	16%	13%	2%	1%	5%	5%

* B is the baseline trajectory, T# is the test trajectory number.

** L is the linear model approximation and NL the nonlinear approximation.

In general, the errors for the linear model approximation are comparable to those for the nonlinear approximation. The linear model approximation exhibits slightly higher percentage errors for regions of the test trajectories with greater changes in amplitude,

especially T2 and T4. This indicates that the learned linear model provides a reasonable approximation of the system behaviour for the four trajectories although the frequency and amplitudes tested are fairly uniform.

3.8.2 Learned Gains and Equivalent CCM Gains Comparisons

Figures 3.11 and 3.12 show the values for K_α and K_q found via each controller optimisation technique. The results are also listed in Table 3.10. A closed loop damping ratio and natural frequency is also listed to give the theoretical linear performance parameters for the gains adopted. From the two gain plots we can see that the pole placement technique for both the baseline dynamics and the damaged aircraft puts a controller in the same location relative to design specifications for damping ratio and natural frequency described by the limits of the bounding red box. In the damaged case however the pole placement generates a controller with a much higher gain for K_α. The tradeoff therefore of having a design goal and a limit on the controller gains must be understood to optimise the control choice. The LQR technique using the identity weighting for both baseline and damaged aircraft places the gains outside the design limits in the damaged case but a suitably weighted Q matrix found via iteration produces gains at the design condition for both the baseline but quite far outside the design limits for the damaged case. This poses the question of how to tune the Q weighting for the adaptive case to maintain performance requirements. The first CCM algorithm, A002 finds the largest feasible condition number which increases the norm of the K matrix resulting in large K_α gains. While the controller may be stable for the closed loop system, the large gains may become unstable when introducing uncertainty into the dynamics creating a non-robust solution. The CCM A005 algorithm maximising the contraction rate finds a feasible solution for both damaged and baseline dynamics but is also a more aggressive controller compared to the nominal design condition. The minimising cost CCM A005 algorithm is outside the design specifications for both cases being too conservative. Both the CCM solutions that optimise for a minimum distance to a design ratio of the negative pole provide solutions

close to the pole placement technique and within the design specifications.

Table 3.10 Gains Optimised for the Baseline and Damaged Aircraft for Each Technique

Alg. Test #	Description	Aircraft Condition	K_α	K_q	ζ_{cl}	$\omega_{n_{cl}}$
MATLAB	Pole Placement	Baseline	-0.90	-0.43	0.70	3.00
MATLAB	LQR Q 1 1	Baseline	-0.29	-0.92	1.29	2.74
MATLAB	LQR Q 2 002	Baseline	-0.84	-0.41	0.69	2.93
A002	Bisection Lambda	Baseline	-3.05	-1.07	0.83	4.94
A005	Max Lambda	Baseline	-1.07	-0.56	0.77	3.26
A005	Min Cost	Baseline	-0.06	-0.13	0.69	1.67
A005	Min Ratio Dist	Baseline	-0.64	-0.44	0.77	2.74
Grid Search	Optimum Tracking Error	Baseline	-0.5	-0.24	0.62	2.42
MATLAB	Pole Placement	Damaged	-4.14	-0.45	0.70	3.00
MATLAB	LQR Q 1 1	Damaged	-0.23	-0.97	1.24	1.51
MATLAB	LQR Q 2 002	Damaged	-0.80	-0.40	0.66	1.66
A002	Bisection Lambda	Damaged	-22.12	-0.50	0.85	6.43
A005	Max Lambda	Damaged	-5.91	-0.39	0.71	3.50
A005	Low Cost	Damaged	-0.63	-0.37	0.64	1.55
A005	Min Ratio Dist	Damaged	-3.98	-0.47	0.71	2.96
Grid Search	Optimum Tracking Error	Damaged	-0.375	-0.3	0.60	1.37

3.8.3 Optimum Gains via Grid Search and Reference Model

Figure 3.13 show grid searches for $0 \geq K_\alpha \geq -2.5$ and $0 \geq K_q \geq -1.2$ overlaid with the reference model response and the closest response to the reference model in terms of tracking distance. The responses are all initiated with a disturbance of 5 degrees angle of attack or 5 degrees per second for the pitch rate responses respectively. The upper

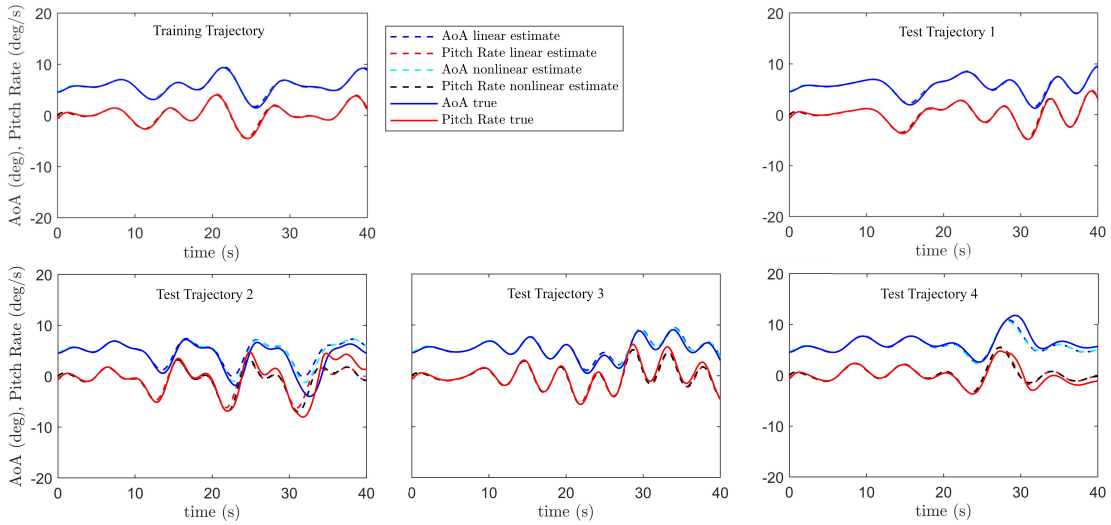


Figure 3.10 – Flight Trajectory Data Compared to Learned Model Estimated Trajectories from Input Excitations

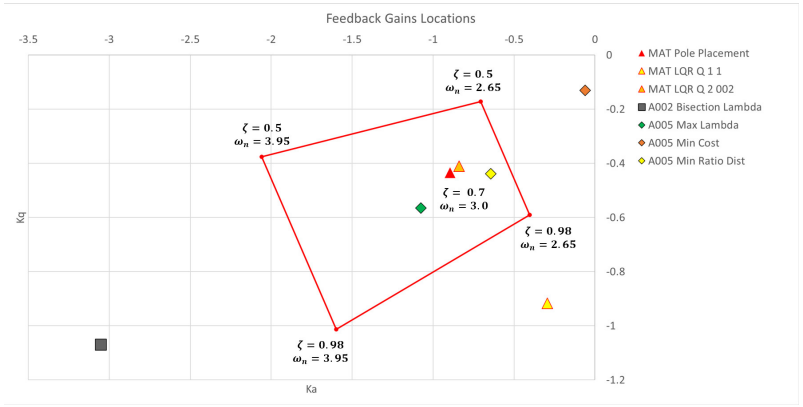


Figure 3.11 – Baseline Optimised Gains Compared to Design Specification Limits Using Pole Placements

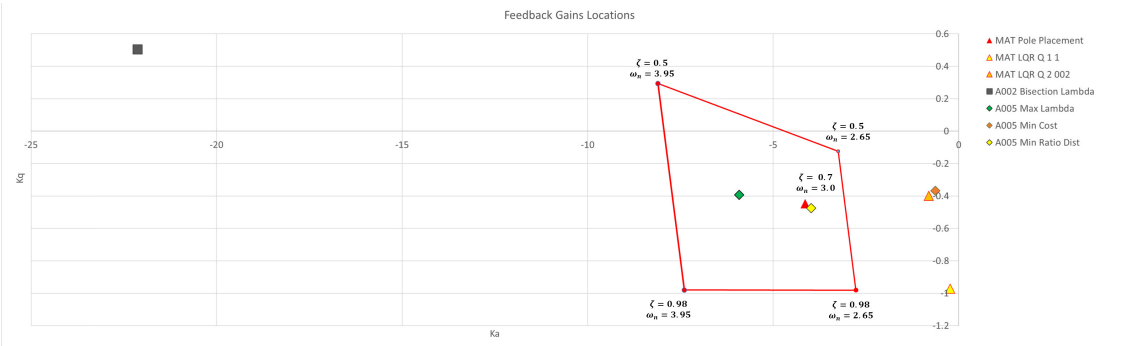


Figure 3.12 – Optimised Gains for the Damaged Aircraft Compared to Design Specification Limits Using Pole Placements

row of plots in Fig. 3.13, are for the baseline aircraft model and the lower row plots are for the damaged aircraft model. The damaged grid search expanded the search space to include the range found for the pole placement solution so that $0 \geq K_\alpha \geq -5$ and $0 \geq K_q \geq -1.2$. The first thing to note is that the damaged aircraft is actually more stable in the low range gains and unstable in the high gain ranges in angle of attack whereas all responses are stable but underdamped for the baseline aircraft. The damaged aircraft is therefore less maneuvrable as a result of the damage. It is important to highlight that using tracking error to optimize gains, even with a reference model, leads to a different optimum solution compared to the pole placement technique. This difference arises because the cost function has a vast solution space, or flat minimum, where steady-state errors consistently exhibit a nominal offset as well as nonlinear response variations.

3.8.4 Controller Response Comparisons for Initial Displacements in AoA and Pitch Rate

Figures 3.14 and 3.15 present the angle of attack and pitch rate responses for each controller. Figure 3.14 illustrates the responses obtained from a nonlinear simulation, which employs controllers derived from a linear approximation of the aircraft's dynamics. It is important to note that these control policies are tested against a nonlinear dynamics model of the aircraft for both baseline and damaged profiles. Conversely, Figure 3.15 shows responses from a linear model simulation, where both the approximation of the dynamics and the derivation of each control policy are derived from linear models for both the baseline and the damaged aircraft dynamics. The comparison between these figures is not merely an evaluation of the baseline control policies' robustness across different controllers but rather an assessment of each controller's adaptability as an adaptive control policy. This adaptability is highlighted when the control policies, formulated based on a linear model approximation of the dynamics, are applied to both the nonlinear system and a simplified linear model simulation. The key takeaway is that the nonlinear simulation reveals instabilities not apparent in the linear model simulation. This difference under-

scores the nonlinear model's capacity to introduce additional uncertainties that might be stable in a linear model but create instability in the nonlinear framework. From the plots we can see that the controllers that fail for the damaged adaptive control responses are controllers that met the design goal or pole placement or were more aggressive in designing the controller gains. So the CCM A002 bisection controller became unstable as well as the CCM A005 maximum lambda, and the CCM A005 minimum distance to the desired poles as well as the standard pole placment technique where a specific damping and natural frequency is designed for. It can also be seen that the controllers that failed in the nonlinear simulation were stable in the linear model approximation which highlights the fact that they were non-robust in some uncertainty in the nonlinear model. This is something that can be explored further in designing a more robust adaptive controller. Looking at the LQR solutions the naive weighted LQR controller with the identity weightings is less aggressive than the optimised LQR solution but both seem to be robust to the uncertainties in the damaged case. There is a tradeoff between the performance desired by the controller and the robustness in the LQR responses because the optimum controller responses are within a middle ground of the two approaches. Where the LQR controllers succeed in remaining stable there is some loss in performance in the linear approximations in terms of response times. It can also be seen that the CCM controller that minimises the cost function is stable in all cases and seems to find a middle ground in robustness and performance but lacks performance in the baseline case. This means that minimising the cost may in some sense be a conservative but robust solution should a more aggressive controller fail to meet the stability requirements. Apart from the reference model all controllers have a small steady state error due to being proportional feedback controllers with an estimated input gain which makes it more difficult to compare to the reference model in terms of tracking errors.

3.8.5 Cummulative tracking errors compared to the Reference Model

The right hand plots in Fig. 3.13 show how the tracking errors are used to define the optimum controller via a grid search over a range of K_α and K_q values. The individual controller tracking errors for the baseline and damaged aircraft are shown in Figs. 3.14 and 3.15 for the linear approximation. It can be seen that the tracking errors optimise for a single set of gains based on the distance to a reference signal but the distance is still significant due to the grid search resolution, large solution space and due to steady state errors. In this sense the tracking error of a reference model only provides limited information about the suitability of a controller but provides a useful guide. In the nonlinear simulation for the damaged aircraft the controllers that followed the reference model most closely were the CCM A005 minimum cost controller and the two LQR solutions whereas in the baseline case all controllers except for the CCM A002 bisection controller were fairly closely bunched together offset by the steady state error accumulation. In the linear model approximations the minimum expected tracking errors were from the controllers optimised to the design conditions, namely the CCM A005 min ratio distance, pole placement, and the CCM A005 maximum lambda controllers. The gap between the linear approximation and the nonlinear uncertainty is where the optimum controller must be found.

3.9 Conclusion

We have developed an initial study for an online learning framework for an F-16 which utilizes convex optimisation as a linear regression of monomial basis features and certifies a control policy via a static control contraction metric. The learning pipeline is implemented as a simulation in Simulink on a 3-DOF nonlinear model and the learning technique is compared to LQR control optimisation of a two dimensional full state feedback control scheme. The control policy is demonstrated via disturbance rejection and tracking of a reference model for the short period mode approximation of the dynamics.

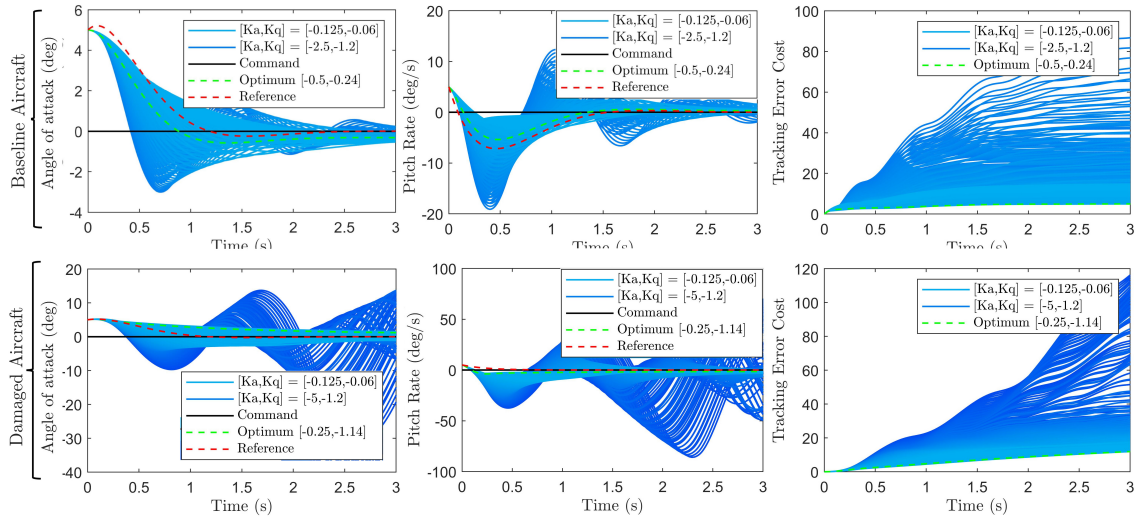


Figure 3.13 – Grid Search Disturbance Rejection Optimisation for Baseline vs Damaged Aircraft from Initial Offset of 5 deg Angle of Attack and 5 deg/s Pitch Rate

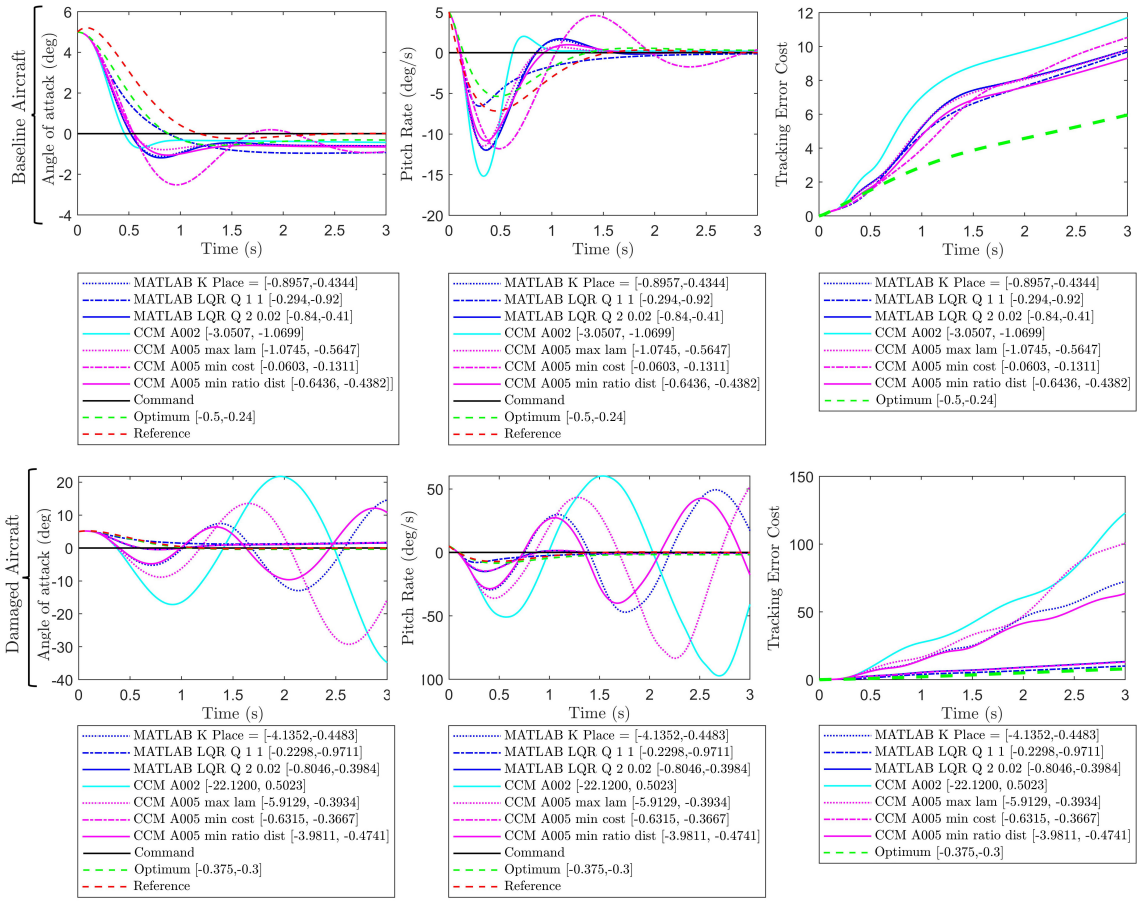


Figure 3.14 – Disturbance Rejection of the Nonlinear Baseline vs Damaged Aircraft from Initial Offset of 5 deg Angle of Attack and 5 deg/s Pitch Rate

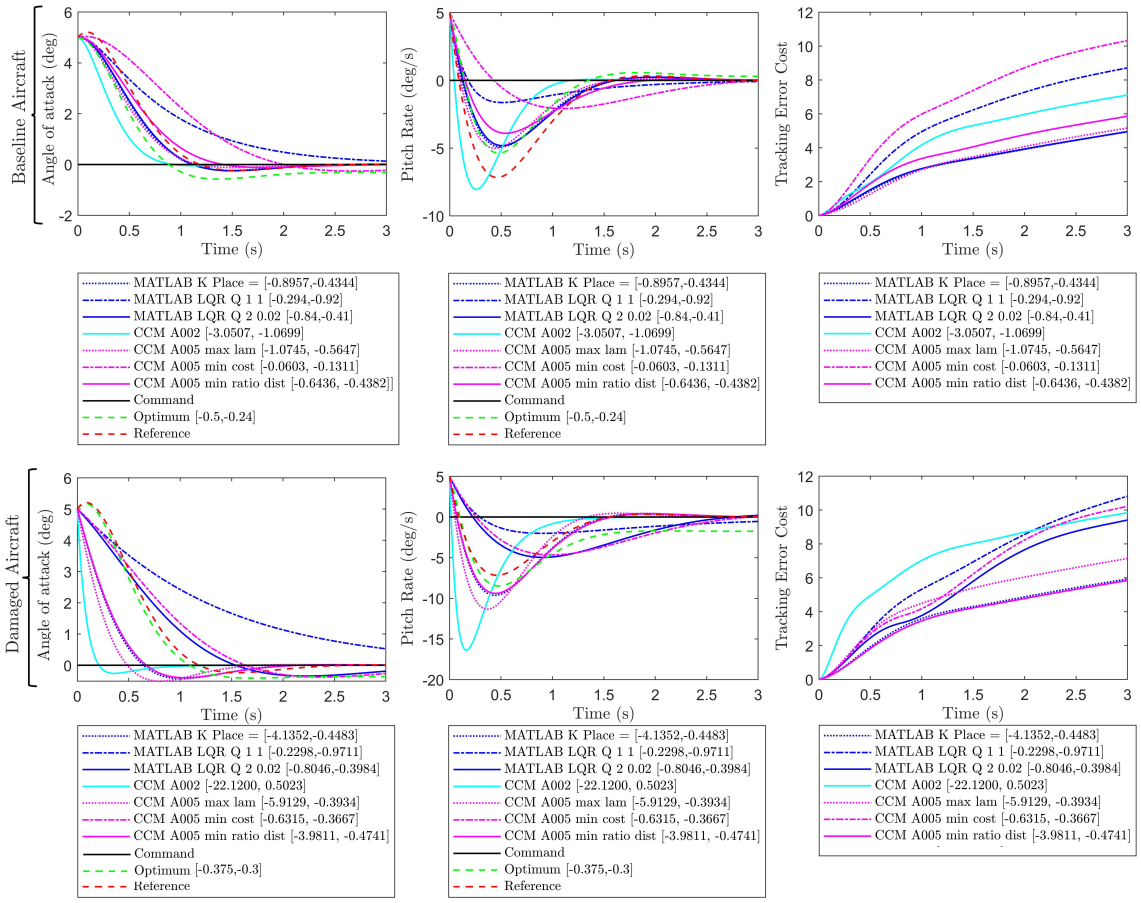


Figure 3.15 – Disturbance Rejection of a Linear Model Approximation for Baseline vs Damaged Aircraft from Initial Offset of 5 deg Angle of Attack and 5 deg/s Pitch Rate

The technique provides an insight into the use and optimisation of control contraction metrics for greater understanding and development of the control policy for the wider envelope and a route into the nonlinear case either as a gain scheduling policy or via nonlinear convex optimisation techniques such as sum-of-squares programming. It is thought that the use of expert knowledge of the system and pre-determined physics models will be the best approach for learning models in flight. Future work will expand upon the types of faults introduced and explore the robustness of the control policy while considering inner and outer loop optimisation.

Appendix

3.A Finsler's Lemma

Lemma 3.A.1 (Finsler's Lemma)**.** *Given $w \in \mathbb{R}^{n_x}$, $\mathcal{L} \in \mathbb{R}^{n_x \times n_x}$, and $\mathcal{B} \in \mathbb{R}^{m_x \times n_x}$, with rank $\mathcal{B} < n_x e\mathcal{B}_\perp$ a basis for the null space of $\mathcal{B}$ (i.e., $\mathcal{B}\mathcal{B}_\perp = 0$). Then the following conditions are equivalent [14]:*

$$
\begin{aligned}
&1) \quad w^T \mathcal{L} w < 0, \forall w \neq 0 \colon \mathcal{B} w = 0 \\
&2) \quad \mathcal{B}_\perp^T \mathcal{L} \mathcal{B}_\perp < 0 \\
&3) \quad \exists \mu \in \mathbb{R} \colon \mathcal{L} - \mu \mathcal{B} \mathcal{B}^T < 0 \\
&4) \quad \exists \mathcal{X} \in \mathbb{R}^{n_x \times m_x} \colon \mathcal{L} + \mathcal{X}\mathcal{B} + \mathcal{B}^T \mathcal{X}^T < 0
\end{aligned}
$$

Proof. Finsler's lemma proof can be found in [17]. □

References

[1] A. A. Abdullah, P. A. Ioannou, and G. Samaras. Control design for f-16 longitudinal motion. *IFAC Proceedings Volumes*, 37(6):529–534, 2004.

[2] R. J. Adams, J. M. Buffington, and S. S. Banda. Design of nonlinear control laws for high-angle-of-attack flight. *Journal of Guidance, Control, and Dynamics*, 17(4):737–746, 1994.

[3] A. A. Ahmadi, A. Chaudhry, V. Sindhwani, and S. Tu. Safely learning dynamical systems from short trajectories. pages 498–509, 2021.

[4] O. Albostan and M. Gokasan. High angle of attack manoeuvring control of f-16 aircraft based on nonlinear dynamic inversion and eigenstructure assignment. In *7th European Conference for Aeronautics and Space Sciences (EUCASS)*, pages 3–6, 2017.

[5] M. Aps. Mosek modeling cookbook, 2020.

[6] E. M. Aylward, P. A. Parrilo, and J.-J. E. Slotine. Stability and robustness analysis of nonlinear systems via contraction metrics and sos. *Automatica*, 44(8):2163–2170, 2008.

[7] G. J. Balas. Flight control law design: An industry perspective. *European Journal of Control*, 9(2-3):207–226, 2003.

[8] F. Berkenkamp, A. Krause, and A. P. Schoellig. Bayesian optimization with safety constraints: safe and automatic parameter tuning in robotics. *arXiv preprint arXiv:1602.04450*, 2016.

[9] F. Berkenkamp and A. P. Schoellig. Safe and robust learning control with gaussian processes. *2015 European Control Conference, ECC 2015*, pages 2496–2501, 2015.

[10] F. Berkenkamp, M. Turchetta, A. P. Schoellig, and A. Krause. Safe model-based reinforcement learning with stability guarantees. *Advances in Neural Information Processing Systems*, 2017-Decem(Nips):909–919, 2017.

[11] N. M. Boffi, S. Tu, N. Matni, J. J. E. Slotine, and V. Sindhwani. Learning stability certificates from data. *arXiv*, pages 1–30, 2020.

[12] S. Boyd and L. Vandenberghe. *Convex optimization*. Cambridge University Press, 2004.

[13] S. L. Brunton, J. Nathan Kutz, K. Manohar, A. Y. Aravkin, K. Morgansen, J. Klemisch, N. Goebel, J. Buttrick, J. Poskin, A. Blom-Schieber, T. Hogan, and D. McDonald. Data-driven aerospace engineering: Reframing the industry with machine learning. *arXiv*, pages 1–35, 2020.

[14] L. F. Buzachero, E. Assuncao, M. C. Teixeira, and E. R. Da Silva. New techniques for optimizing the norm of robust controllers of polytopic uncertain linear systems. *Frontiers in advanced control systems*, pages 75–100, 2012.

[15] Y. Chow, O. Nachum, A. Faust, E. Duenez-Guzman, and M. Ghavamzadeh. Lyapunov-based safe policy optimization for continuous control. *arXiv e-prints*, 2019.

[16] M. V. Cook. *Flight dynamics principles: a linear systems approach to aircraft stability and control*. Butterworth-Heinemann, 2012.

[17] M. C. de Oliveira and R. E. Skelton. Stability tests for constrained linear systems. In *Perspectives in robust control*, pages 241–257. Springer, 2007.

[18] S. Dean, S. Tu, N. Matni, and B. Recht. Safely learning to control the constrained linear quadratic regulator. In *2019 American Control Conference (ACC)*, pages 5582–5588, 2019.

[19] C. Edwards, H. Alwi, and C. Tan. Sliding mode methods for fault detection and fault tolerant control with application to aerospace systems. *International Journal of Applied Mathematics and Computer Science*, 22(1):109–124, 2012.

[20] U. Eren, A. Prach, B. B. Koçer, S. V. Rakovic, E. Kayacan, and B. Açikmese. Model predictive control in aerospace systems: Current state and opportunities. *Journal of Guidance, Control, and Dynamics*, 40(7):1541–1566, 2017.

[21] J. Fisac, A. Akametalu, M. Zeilinger, S. Kaynama, J. Gillula, and C. Tomlin. A general safety framework for learning-based control in uncertain robotic systems. *IEEE Transactions on Automatic Control*, 64(7):2737–2752, 2019.

[22] J. F. Fisac, A. K. Akametalu, M. N. Zeilinger, S. Kaynama, J. Gillula, and C. J. Tomlin. A general safety framework for learning-based control in uncertain robotic systems. *IEEE Transactions on Automatic Control*, 64(7):2737–2752, 2019.

[23] R. Grandia, A. Taylor, A. Singletary, M. Hutter, and A. Ames. Nonlinear model predictive control of robotic systems with control lyapunov functions. *Robotics: Science and Systems XVI*, Jul 2020.

[24] J. J. Harris and J. R. Stanford. F-35 flight control law design, development and verification. *2018 Aviat. Technol. Integr. Oper. Conf.*, pages 1–18, 2018.

[25] S. Heyer, D. Kroezen, and E.-J. Van Kampen. Online adaptive incremental reinforcement learning flight control for a cs-25 class aircraft. In *AIAA Scitech 2020 Forum*, page 1844, 2020.

[26] T. Krüger, P. Schnetter, R. Placzek, and P. Vörsmann. Fault-tolerant nonlinear adaptive flight control using sliding mode online learning. *Neural Networks*, 32:267–274, 2012.

[27] K. Leung and I. R. Manchester. Nonlinear stabilization via control contraction met-

rics: A pseudospectral approach for computing geodesics. *Proc. Am. Control Conf.*, pages 1284–1289, 2017.

[28] F. L. Lewis and D. Vrabie. Design Optimal Adaptive Controllers. *IEEE Control Syst.*, 32(November):76–105, 2012.

[29] J. Löfberg. YALMIP: A toolbox for modeling and optimization in MATLAB. *Proc. IEEE Int. Symp. Comput. Control Syst. Des.*, pages 284–289, 2004.

[30] J. Löfberg. Pre- and post-processing sum-of-squares programs in practice. *IEEE Transactions on Automatic Control*, 54(5):1007–1011, 2009.

[31] W. Lohmiller and J.-J. E. Slotine. On contraction analysis for nonlinear systems analyzing stability differentially leads to a new perspective on dynamic systems. *Automatica*, 34(6)(December 1997):683–696, 1997.

[32] W. Lohmiller and J.-J. E. Slotine. On contraction analysis for non-linear systems. *Automatica*, 34(6):683–696, 1998.

[33] J. Löfberg. Decay rate computation in lti system, 2022.

[34] I. R. Manchester and J.-J. E. Slotine. Control contraction metrics and universal stabilizability. *IFAC Proceedings Volumes*, 47(3):8223–8228, 2014.

[35] I. R. Manchester and J.-J. E. Slotine. Control contraction metrics, robust control and observer duality. *arXiv*, 2014.

[36] I. R. Manchester and J.-J. E. Slotine. Control contraction metrics: Convex and intrinsic criteria for nonlinear feedback design. *IEEE Transactions on Automatic Control*, 62(6):3046–3053, 2017.

[37] I. R. Manchester and J. J. E. Slotine. Robust control contraction metrics: A convex approach to nonlinear state-feedback control. *IEEE Control Systems Letters*, 2(3):333–338, 2018.

[38] I. R. Manchester, J. Z. Tang, and J.-J. E. Slotine. Unifying robot trajectory tracking with control contraction metrics. In *Robotics Research*, pages 403–418. Springer, 2018.

[39] M. W. Mehrez, G. K. Mann, and R. G. Gosine. Comparison of stabilizing NMPC designs for wheeled mobile robots: An experimental study. *MERCon 2015 - Moratuwa Eng. Res. Conf.*, (September):130–135, 2015.

[40] M. A. Mohd Basri, A. R. Husain, and K. a. Danapalasingam. Intelligent adaptive backstepping control for MIMO uncertain non-linear quadrotor helicopter systems. *Transactions of the Institute of Measurement and Control*, 37(3):345–361, 2015.

[41] E. A. Morelli. Global nonlinear parametric modelling with application to f-16 aerodynamics. In *Proceedings of the 1998 American Control Conference. ACC (IEEE Cat. No. 98CH36207)*, volume 2, pages 997–1001. IEEE, 1998.

[42] L. T. Nguyen. *Simulator study of stall/post-stall characteristics of a fighter airplane with relaxed longitudinal static stability*, volume 12854. National Aeronautics and Space Administration, 1979.

[43] D. T. Ocaña, H.-S. Shin, and A. Tsourdos. Development of a nonlinear reconfigurable f-16 model and flight control systems using multilayer adaptive neural networks. *IFAC-PapersOnLine*, 48(9):138–143, 2015.

[44] S. Russell, Richard. Non-linear f-16 simulation using simulink and matlab. *University of Minnesota, Version*, 1, 2003.

[45] S. Seshagiri and E. Promtun. Sliding mode control of f-16 longitudinal dynamics. In *2008 American control conference*, pages 1770–1775. IEEE, 2008.

[46] S. Singh, A. Majumdar, J. J. Slotine, and M. Pavone. Robust online motion planning via contraction theory and convex optimization. *Proc. - IEEE Int. Conf. Robot. Autom.*, pages 5883–5890, 2017.

[47] S. Singh, S. M. Richards, V. Sindhwani, J.-J. E. Slotine, and M. Pavone. Learning stabilizable nonlinear dynamics with contraction-based regularization. *The International Journal of Robotics Research*, 2020.

[48] S. Singh, V. Sindhwani, J.-J. E. Slotine, and M. Pavone. Learning Stabilizable Dynamical Systems via Control Contraction Metrics. *arXiv e-prints*, page arXiv:1808.00113, July 2018.

[49] L. Sonneveldt, E. R. Van Oort, Q. P. Chu, C. C. De Visser, J. A. Mulder, and J. H. Breeman. Lyapunov-based fault tolerant flight control designs for a modern fighter aircraft model. *AIAA Guidance, Navigation, and Control Conference and Exhibit*, (August):1–23, 2009.

[50] B. L. Stevens, F. L. Lewis, and E. N. Johnson. *Aircraft control and simulation: dynamics, controls design, and autonomous systems*. John Wiley & Sons, 2015.

[51] D. Sun, S. Jha, and C. Fan. Learning certified control using contraction metric. *arXiv preprint arXiv:2011.12569*, 2020.

[52] H. Tsukamoto and S. Chung. Neural contraction metrics for robust estimation and control: A convex optimization approach. *IEEE Control Systems Letters*, 5(1):211–216, 2021.

[53] H. Tsukamoto and S. J. Chung. Robust controller design for stochastic nonlinear systems via convex optimization. *IEEE Transactions on Automatic Control*, pages 1–1, 2020.

[54] A. Tunik and O. Nadsadna. Robust digital gain-scheduling control of the uav altitude. In *2017 IEEE 4th International Conference Actual Problems of Unmanned Aerial Vehicles Developments (APUAVD)*, pages 175–179. IEEE, 2017.

[55] K. P. Wabersich and M. N. Zeilinger. Safe exploration of nonlinear dynamical systems: A predictive safety filter for reinforcement learning. *arXiv*, 2018.

[56] R. Wang, R. Tóth, and I. R. Manchester. Virtual control contraction metrics: Convex nonlinear feedback design via behavioral embedding. *arXiv preprint arXiv:2003.08513*, 2020.

[57] S. Wang, G. Zhu, X. Chen, X. Zhang, J. Xu, X. Li, and H. Cao. Neural Networks-based Robust Adaptive Dynamic Surface Sliding Mode Control of Flight Path Angle with Tracking Error Constraints. *2019 IEEE 28th International Symposium on Industrial Electronics (ISIE)*, pages 587–592, 2019.

[58] W. Yang, M. Hammoudi, G. Herrmann, M. Lowenberg, and X. Chen. Two-state dynamic gain scheduling control applied to an f16 aircraft model. *International Journal of Non-Linear Mechanics*, 47(10):1116–1123, 2012.

Chapter 4

Cascaded 2-Loop CCM and Hybrid RCCM Adaptive Controller Design for Longitudinal Flight Control

4.1 Introduction

Adaptive control systems to enhance safety, efficiency, and performance of aircraft are of great interest. Progress in data-driven learning, speed of computation and a desire to reduce the costs of testing and servicing of aircraft have increased the amount of research in the field. In this work we study the use of static Control Contraction Metrics (CCMs) within a Linear Matrix Inequality (LMI) framework. LMIs are known for their uncertainty handling capabilities and a convex formulation of a safety certificate offers a flexible framework for approximating functions. The main advantage of the CCM framework is the two stage certification and synthesis of a controller online for nonlinear systems, which is more efficient than nonlinear model predictive control (NMPC). This study serves as an initial exploration of the use of CCMs for linear approximations and their relationships to conventional methodologies such as pole placement, the Linear Quadratic Regulator (LQR), and the root locus method. CCMs are a mathematical framework first proposed by Manchester and Slotine (2014) [8] and have gained attention recently for nonlinear control problems for various applications, including robotics and UAV control [17],[11]. The key idea behind CCMs is to leverage linear analysis tools to

certify contraction in a generalised co-ordinate space by using set invariance tools like convex optimisation and then to solve a less computationally expensive control synthesis problem as an integration of a geodesic between tracking offsets. In the case of linear time invariant systems the geodesic is just a straight line and can be easily solved using linear matrix multiplications.

4.1.1 Problem Statement

Although a powerful technique, currently the use of CCMs for control systems is focused on nonlinear control theory and there is currently a gap between linear and nonlinear design for control systems. The nonlinear CCM approaches bound the upper and lower eigenvalues of the metric to define the closed loop performance of the resulting system. For nonlinear systems a robustness tube is typically adopted because the dynamics are inherently more complex. For linear systems however robustness can be guaranteed by adopting an optimal control policy via LQR controllers. The problem of adopting the LQR technique is that of finding weighting matrices using trial and error which can be time consuming due to the need to optimise over a number of set points. It is also desirable to find weighting matrices that meet eigenstructure requirements for Linear Time Invariant (LTI) design specifications based on second order dynamics. For aerospace applications there are also LTI based performance specifications such as the military standards for flying qualities. In order to leverage new control synthesis approaches such as CCMs an understanding how these two approaches can work together is desirable for real world application. Current convex optimization techniques adopted for the optimisation of CCMs bound the upper and lower eigenvalues of a metric and lack an explicit design of the placement of poles specifically in reference to second-order reference models and the analysis of LTI theory. Further there is often a trade-off between robustness and performance defined by the use of different control synthesis approaches that are certified via formal methods. The adoption or trade-off of these formal methods to certify a mixed performance is often not possible due to non-convex solution spaces. We explore these

formal mixed methods under certain conditions to provide guarantees on performance and open up the opportunity to parameterise key performance metrics. This work therefore aims to address the gaps in knowledge on the application of the CCM technique and convex optimisation to work towards a framework and technology applicable within the aerospace industry and to propose learning parameters suitable for online adaptive control to improve aircraft performance from trajectory data.

4.1.2 Approach Overview

In this research, we present a framework for aircraft control optimization using CCMs, with a focus on longitudinal control. The work continues the study of a learning pipeline for a longitudinal aircraft controller using convex optimisation and an approximation of the aircraft dynamics from trajectory data. In this work we explore the use of CCMs in higher dimensions and the challenges in creating a robust controller subject to deviations and faults in the nonlinear aircraft dynamics approximated as a linear estimate for the short period mode. The 6 damaged aircraft profiles relate to safe controllable deviations from the baseline dynamics taking the aircraft out the nominal Level 1 flying qualtities to show how the performance of the aircraft can be improved without the need for recovery or control allocation techniques. Our approach aims to provide a methodology that can be tailored to other fixed wing aircraft.

4.1.3 Statement of Contributions

The main contributions are:

1. The integration of flying qualities into the design of a CCM controller.
2. Developed a novel cascaded two-loop control contraction metric to explicitly optimise state contraction properties.
3. A study of a hybrid LMI approach that combines robust and exponential constraints with a mixing parameter.

4. Conditions for stability guarantees of a linear sum hybrid control policy and simulated verfication.

4.1.4 Related Work

There have been many recent works in control contraction metrics for aerospace applications in research, [22, 14, 25]. Mostly for nonlinear control problems adopting convex optimisation either via the sum-of-squares technique, [14] or more recently a convex neural network approach, [25] or via kernel methods, [16] for function approximation. The control synthesis however for nonlinear control problems using the CCM technique is non-trivial, [9] and requires knowledge of differential geometry and the use of geodesic mathematics on manifolds. Similarly the complexity of processing a controller is numerical challenging, typical methods being the use of a Pseudospectral collocation technique or grid based search techniques [4, 14]. Whereas for linear systems the solutions are trivial straight line calculations. The CCM framework has been extended to H_∞ and nonlinear LQR like frameworks using L_2 gain bounds and incremental L_2 gain bounds in [10]. Robust techniques for control systems are limited in predicting flying qualities for aircraft. A number of researchers have explored techniques to place eigenvalues via optimisation of the weights of an LQR structure, [20], [12]. The authors in [20] adopted a multi-loop algorithm to generate Q-matrix diagonal terms by transforming the desired closed loop system to a canonical form and then generating a set of Q matrix iterations that are summed together to create the final weighting matrix. The resulting controller achieved convergence charactersitics to meet damping and settling time requirements. A step by step analytical approach with a trial weighting parameter ρ is adopted by [12] in a simultaneous LQR and Eigenvalue Assignment (EA) algorithm. They show that with a design set of eigenvalues and eigenvectors the weighting parameters for the Q and R matrices in the LQR controller achieves both a robust and desired eigenstructure for the system. Their approach is validated on an F-16 longitudinal flight path and pitch attitude tracking simulation. In [23] a Control Anticipation Parameter (CAP) is formulated in

an LMI that ensures a certain flying quality for the closed loop system adopting an H_∞ control structure. The controller achieves both the robustness of the H_∞ bound and eigenvalue constraints imposed by the CAP constraints. The solution is also solved as a convex problem. However the CAP constraints are upper and lower bounded so an explicit solution was not the design goal rather a guarantee on the flying quality limits. In [18] a Multi-Objective Genetic Algorithm (MOGA) searches for solutions for the weights of an LQR controller generating mulitple solutions on a pareto front. The objectives are based on multiple design criteria such as pitch rate, drop back, control effort, comfort and rise time for a civil transport aircraft. The current literature explores the use of CCMs but the focus is on their application to nonlinear control problems. Our work explores a practical implementation of CCMS with the aim to bridge the gap between the nonlinear theory and the certification of linear controllers. We focus on the synthesis of a controller that meets desired flying qualities and robustness constraints in the context of LTI systems.

4.1.5 Layout of the paper

The rest of this paper is structured as follows: Section 5.2 describes the attitude hold longitudinal control architecture; Section 5.3 reviews the CCM technique for static metrics compared to traditional LTI approaches; Section 5.4 describes the inclusion of flying qualities in the CCM optimisation and proposes a loop shaping approach as well as a hybrid Robust CCM approach; Section 5.5 studies a traditional root locus design approach; Section 5.6 describes the step response analysis used to compare the controller designs; Section 4.7 details the results of the study; Section 4.8 concludes the study and proposes future work.

4.2 Definition of the Aircraft Longitudinal Attitude Hold Controller

In this section we describe a three state attitude hold proportional feedback control structure for the aircraft simulator. The controller is composed of an inner loop full state feedback in angle of attack and pitch rate with feedback gains in the feedback loop as in a regulator control structure. The outer loop is a negative feedback in θ to control the pitch angle of the aircraft for a short period mode. Since the feedback in the outer loop is in the forward loop and the θ state is the integral of the pitch rate it ensures that the closed loop controller in pitch rate has a steady state error of zero. The three gain terms arbitrarily decide the closed loop poles for the system and so is fully controllable and stabilisable. It must be noted that typically the angle of attack is not included in pitch attitude controllers because the state is not easily measured practically but for this study it is used as an example of a controller using higher dimensions in a two loop layout to understand the use of contraction metrics.

The full state proportional feedback controller for the short period is shown in Fig. 5.2.2. The cascade framework is applicable to any control problem in higher dimensions or with a higher number of loops. The state feedback can also be augmented with integral, error or derivative terms to be similar in design to standard PID or other feedback techniques.

4.3 Review of Control Contraction Metrics (CCM)

In this section, we briefly outline the control contraction method to define a stability certificate for a linear time-invariant system with dynamics described by:

$$\dot{x} = Ax + Bu$$

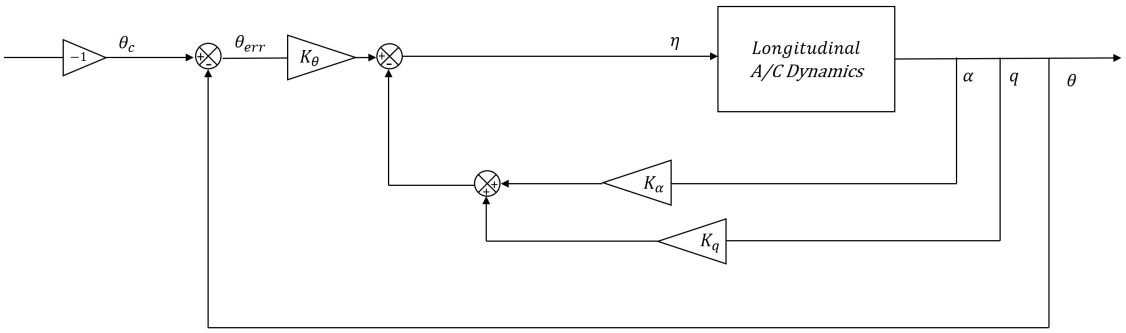


Figure 4.2.1 – Two-Loop 3rd Order Full State Proportional Feedback Controller

where A is an $n \times n$ matrix, B is an $n \times m$ matrix, and x is a vector of states ($x \in \mathbb{R}^n$) u is a vector of control input ($u \in \mathbb{R}^m$).

Full state feedback with a control input including nominal inputs such as a pilot stick input or disturbances is given as:

$$u = v - K\delta x$$

$$\dot{x} = Ax + B(v - K\delta x) = (A - BK)\delta x + Bv$$

In the modified control input, $u = v - K\delta x$, where v is an exogenous input. The CCM method incorporates trajectory differentials to define the feedback inputs $\delta x = (x^* - x)$ where x^* is a reference trajectory. The CCM technique was developed by Manchester and Slotine [7] for nonlinear systems to certify contraction of pairwise trajectories for a system by searching for metric and contraction rate that validates an LMI similar to the established Control Lyapunov Functions (CLFs). In this study the work considers the use of such a metric for LTI dynamics as a first approximation with the view that the architecture and techniques may be extended to nonlinear dynamics for which CCMs are most attractive.

4.3.1 Conditions for stability for a valid CCM

The search for a static metric for linear dyanamics in the CCM technique reduces to an LMI that can be solved as a semi-definite programme to obtain the coefficients of the matrix W satisfying:

$$\min \qquad -trace(W) \tag{4.1}$$

$$s.t. \qquad W \geq 0 \tag{4.2}$$

$$AW + WA^T - BB^T \leq -2\lambda W \tag{4.3}$$

where A is the systems linear dynamics and B is the control effectiveness matrix. The optimisation involves searching for a dual-metric $W = M^{-1}$, a positive definite symmetric matrix, scaled by a positive coefficient lambda that signifies the rate of contraction, where the metric W measures the negative definite properties of the closed loop dynamics. If the closed loop dynamics in the metric are negative definite by more than $2\lambda W$ then the system is said to be exponentially stable with a contraction rate of λ in the metric. The metric further defines the closed loop feedback gains for the system by inverting back to the original co-ordinate space, $K = \frac{1}{2}B^T M$.

In the CCM formulation where a static metric is found for a given contraction rate λ, the control input is of the form in Eqn. (5.4), a differential feedback controller with static gain matrix K.

$$u(t) = -\frac{1}{2}B^T W^{-1}\delta x(t) = -K\delta x(t) \tag{4.4}$$

4.3.2 Standard CCM Optimisation

The standard CCM optimisation is similar to a Lyapunov optimisation to find a decay rate, [6], but in the CCM optimisation, Eqn. (4.5), a dual metric W is used and a maximisation of the trace adopted, [7]. The derivative constraint, Eqn. (4.7) contains an additonal

control matrix and exponential constraint terms parameterised by the contraction rate λ where $\lambda \in \mathbb{R}_{>0}$.

$$\min_{\{w_{11}w_{22}\}\in\mathbb{R}_{>0}} \quad -trace(W) \tag{4.5}$$

$$s.t. \quad W \geq 0 \tag{4.6}$$

$$-(AW + WA^T - BB^T + 2\lambda W) \geq 0 \tag{4.7}$$

$$W = \begin{bmatrix} w_{11} & w_{12} \\ w_{12} & w_{22} \end{bmatrix} \tag{4.8}$$

$$M = W^{-1} \tag{4.9}$$

$$\text{where} \quad \underline{\beta} I_2 \leq W \leq \bar{\beta} I_2 \tag{4.10}$$

$$\underline{\sigma}(M) = \frac{1}{\bar{\beta}}(W), \quad \bar{\sigma}(M) = \frac{1}{\underline{\beta}}(W),$$

where the upper and lower eigenvalues of W are $\bar{\beta}$ and $\underline{\beta}$ are the reciprical eigenvalues of M represented as $\bar{\sigma}$ and $\underline{\sigma}$ respectively, so that the constraints on W, Eqn. 4.10 limit the condition number and norm of W and by definition M^{-1}. Even though the LMI is convex it is bilinear in the decision variables for W and the decay rate λ. A typical approach to find an optimal decay rate is to use a lower and upper bound for the contraction rate λ, λ_{lower},λ_{upper} based on a reasonable estimate for the eigenvalues for the system. The metric is then optimised with a bisection over λ to find the largest λ feasible while the trace of the dual metric W is maximised, Algorithm 3.

Algorithm 3: CCM Optimisation Using Bisection Algorithm

Input: A and B matrix coefficients of the learned dynamics equation linear approximation.
Initialize: $\lambda_{lower}, \lambda_{upper}$
tol $\leftarrow \Delta\lambda$
$\lambda_{works} \leftarrow \lambda_{lower}$
while $\lambda_{upper} - \lambda_{lower} > tol$ **do**
    $\lambda_{loop} \leftarrow (\lambda_{upper} + \lambda_{lower})/2$
    $W \leftarrow \{w_{11}, w_{12}, w_{22}\} \leftarrow$ **CCM**
    if $(W_{feas} == true)$ **then**
        $\lambda_{upper} = \lambda_{loop}$
    else
        $\lambda_{lower} = \lambda_{loop}$
        $\lambda_{works} = \lambda_{loop}$

4.4 Solution Methodology for CCM Loop Shaping and Hybrid Approaches

To be more explicit in the placement of the inner and outer loop poles and to retain a negative set of feedback gains a new approach that combines traditional loop shaping with the CCM technique is explored. In this section we detail the novel approach to the optimisation technique as a series of two loops optimising the CCM controller for desired contraction rates.

4.4.1 CCM Optimisation Specifying Contraction Rate to Meet Flying Qualities

For a static metric M and LTI dynamics the damping ratio and natural frequency of a second order system define the closed loop pole locations. The real part of the poles determines the desired contraction rate λ.

The general solution for an underdamped 2^{nd} order system transfer function is descibed by:

$$\frac{\alpha(s)}{\delta_h(s)} = \frac{B_\alpha(s - M_q + B_q K_q) + B_q(Z_q - B_\alpha K_q)}{s^2 + s(B_\alpha K_\alpha + B_q K_q - Z_\alpha - M_q) + K_\alpha(Z_q B_q - M_q B_\alpha) + K_q(B_\alpha M_\alpha - Z_\alpha B_q) + Z_\alpha M_q - M_\alpha Z_q} \tag{4.15}$$

$$\frac{q(s)}{\delta_h(s)} = \frac{B_q(s - Z_\alpha + B_\alpha K_\alpha) + B_\alpha(M_\alpha - B_q K_\alpha)}{s^2 + s(B_\alpha K_\alpha + B_q K_q - Z_\alpha - M_q) + K_\alpha(Z_q B_q - M_q B_\alpha) + K_q(B_\alpha M_\alpha - Z_\alpha B_q) + Z_\alpha M_q - M_\alpha Z_q} \tag{4.16}$$

$$G(s) = \frac{k\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \tag{4.11}$$

where the characteristic equation is given by:

$$\Delta(s) = s^2 + 2\zeta\omega_n s + \omega_n^2 \tag{4.12}$$

The roots are determined by the general quadratic solution. So if $4\omega_n^2 > (2\zeta\omega_n)^2$ or $\zeta^2 < 1$ then the roots are complex with poles:

$$p1 = -\zeta\omega_n + j\omega_n\sqrt{1 - \zeta^2} \tag{4.13}$$

$$p2 = -\zeta\omega_n - j\omega_n\sqrt{1 - \zeta^2} \tag{4.14}$$

Given state space matrices for the longitudinal short period dynamics as:

$$A = \begin{bmatrix} Z_\alpha & Z_q \\ M_\alpha & M_q \end{bmatrix}, \quad B = \begin{bmatrix} B_\alpha \\ B_q \end{bmatrix}$$

and a full state feedback matrix $K = [K_\alpha, K_q]$ where $K \in \mathbb{R}^m$. The two state output transfer functions are defined as Eqns. 4.15 and 4.16.

The transfer functions represent second order characteristic equations with zeros in the numerator given by Eqns. (4.18) and (4.19), [2]. The time constants T_α and T_{θ_2} determine

the angle of attack and incidence lag of the short period mode. Because $B_\alpha << B_q$ the incidence lag is largely determined by $\frac{1}{Z_\alpha}$ and is critical in determining the handling qualities of the aircraft, [2], and the manoeuvrability and controllability of the aircraft is measured by the CAP and the Eqn. (4.17).

$$CAP = \frac{-g\omega_n^2}{Z_\alpha U_e} \tag{4.17}$$

where g is gravitational acceleraltion, U_e is the equilibrium velocity. A high CAP means the aircraft is less responsive to control inputs whereas a smaller CAP means the aircraft "feels" lighter and more responsive.

$$\frac{\alpha(s)}{\delta_h(s)} = \frac{K_\alpha(s + \frac{1}{T_\alpha})}{s^2 + s(2\zeta\omega_n) + \omega_n^2} \tag{4.18}$$

$$\frac{q(s)}{\delta_h(s)} = \frac{K_q(s + \frac{1}{T_{\theta_2}})}{s^2 + s(2\zeta\omega_n) + \omega_n^2} \tag{4.19}$$

The equations for the damping ratio and the natural frequency can be defined from:

$$2\zeta\omega_n = B_\alpha K_\alpha + B_q K_q - Z_\alpha - M_q \tag{4.20}$$

$$\omega_n^2 = K_\alpha(Z_q B_q - M_q B_\alpha) + K_q(B_\alpha M_\alpha - Z_\alpha B_q) + Z_\alpha M_q - M_\alpha Z_q \tag{4.21}$$

From which a relation between the Metric M and ζ and ω_n can be derived.

Also noting that:

$$K = \frac{1}{2} B^T M \tag{4.22}$$

We derive an expression for ω_n as follows:

$$\omega_n^2 = Z_\alpha M_q - Z_\alpha B_q K_q - B_\alpha K_\alpha M_q - M_\alpha Z_q \tag{4.23}$$

This can also be written as:

$$\omega_n^2 = K_\alpha C_1 + K_q C_2 + \det(A) \tag{4.24}$$

Upon rearranging, we get:

$$\omega_n = \sqrt{\frac{1}{2} B^T M C + \det(A)} \tag{4.25}$$

Here:

$$C = \begin{bmatrix} C_1 & C_2 \end{bmatrix}^T \tag{4.26}$$

with:

$$C_1 = Z_q B_q - M_q B_\alpha; \quad C_2 = B_\alpha M_\alpha - Z_\alpha B_q \tag{4.27}$$

Noting that:

$$KB = B_\alpha K_\alpha + B_q K_q = \begin{bmatrix} K_\alpha & K_q \end{bmatrix} \begin{bmatrix} B_\alpha & B_q \end{bmatrix}^T \tag{4.28}$$

Leading to:

$$KB = \frac{1}{2} B^T M B \tag{4.29}$$

$$= \frac{1}{2} \begin{bmatrix} B_\alpha \\ B_q \end{bmatrix}^T \begin{bmatrix} m_{11} & m_{12} \\ m_{12} & m_{22} \end{bmatrix} \begin{bmatrix} B_\alpha \\ B_q \end{bmatrix} \tag{4.30}$$

For second order dynamics and a linear approximation therefore a metric and contraction rate can be used to determine the performance parameters ζ and ω_n.

4.4.2 Overview of the MIL-STD flying qualities definitions and key metrics

The flying qualities of military aircraft are categorised according to classes of aircraft, mission phases, and the required quality levels for each mission. These classifications are detailed in [19] and summarised in Table 4.4.1.

Table 4.4.1 Definitions - Flying Qualities Specifications [21]

Airplane Classes	
Class I	Small, light airplanes.
Class II	Medium weight, low-to-medium-maneuverability airplanes.
Class III	Large, heavy, low-to-medium-maneuverability airplanes.
Class IV	High-maneuverability airplanes.
Flight Phases	
Category A	Nonterminal flight phases generally requiring rapid maneuvering.
Category B	Nonterminal flight phases normally accomplished using gradual maneuvers without precision tracking, although accurate flight-path control may be required.
Category C	Terminal flight phases normally accomplished using gradual maneuvers and usually requiring accurate flight-path control.
Flying Qualities Levels	
Level 1	Flying qualities adequate for the mission flight phase.
Level 2	Flying qualities adequate to accomplish the mission flight phase, but some increase in pilot workload or degradation in mission effectiveness exists.
Level 3	Flying qualities such that the airplane can be controlled safely, but pilot workload is excessive, or mission effectiveness is inadequate, or both.

For this study we use a simple bounding box to specify the required damping ratio and natural frequency limits for Level 1 flying qualities in Category A flight phase for a Class IV aircraft. Specifications for flying qualities for the F-16 can be found in MIL-F-8785C [19]. Table 4.4.2 describes the damping ratio and natural frequency limits for the set point of interest in this study where $\frac{n}{\alpha}$ has a value of 14. Gain and phase margins are described in [24]. A detailed specification for time domain tracking performance can also be found in [13], Fig. 4.4.1. The time domain tracking performance described includes a ratio of the overshoot Δq_1 to undershoot Δq_2, a rise time Δt defined by the gradient intercept t_1, and set point intercept t_2 where t_1 also determines the delay time. The specification limits parameterised by velocity V is shown in Table 4.4.3. We use the time domain template to derive performance metrics that depicts Level 1,2 and 3 characteristics for the overshoot, undershoot and rise time at an altitude of 15,000 ft and velocity of 500 ft/s. The performance specifications for the three levels of flying qualities are shown in Table 4.4.4, we assume a 3s settling time within a 5% margin of the command for the requirement for Level 1 flying qualities.

Table 4.4.2 Short-Period Damping Ratio and Natural Frequency Limits for Class IV Aircraft, Cat A flight phases [21]

Level	ζ		ω_n, ($\frac{n}{\alpha} = 14$)	
	Min.	Max.	Min.	Max.
1	0.35	1.30	2.0	7
2	0.25	2.00	1.5	11.8
3	0.10	No limit	1.5	No limit

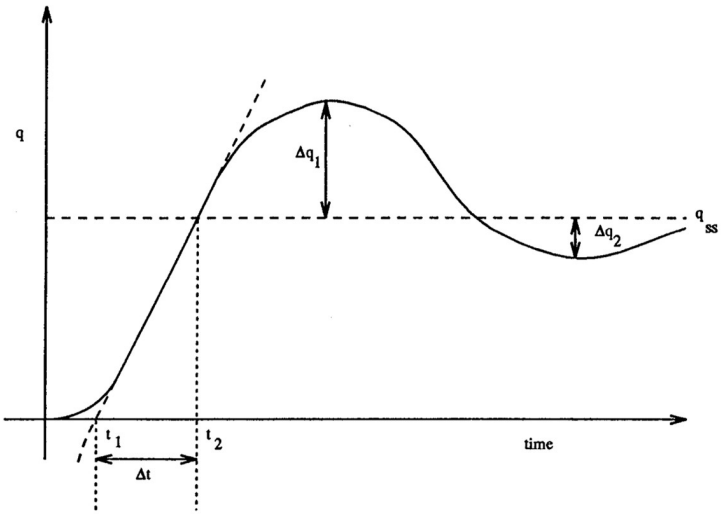


Figure 4.4.1 – MIL-STD Time Domain Tracking Performance Parameters, [13]

Table 4.4.3 MIL-STD Time Domain Tracking Performance Parameters, [13]

Parameter	Specification
t_1	$t_{1_{max}} \leq 0.12$ second
Δt	$\frac{g}{V} \geq \Delta \leq \frac{500}{V}$
$\frac{\Delta q_2}{\Delta q_1}$	$\frac{\Delta q_2}{\Delta q_1} \leq 0.3$

Parameter	Level 1	Level 2	Level 3
t_1 max	0.12s	0.17s	0.21s
$\Delta q_2/\Delta q_1$ max	0.3	0.6	0.85
Δt max	1s	3.2s	-
Δt min	0.018s	0.0064	-
OS %	30	60	-
US %	9	36	-

Table 4.4.4 MIL-STD Time Domain Tracking Specifications at 15,000 ft and 500 ft/s Airspeed

4.4.3 A Double Loop CCM - Optimisation Problem

In the double loop CCM a linear approximation for the dynamics and control effectiveness matrix are first input into the optimisation and in the proposed algorithm the decay rate is chosen as a design condition based on second order flying qualities. If the chosen design

decay rate is infeasible the condition is then relaxed via iterations a distance from the design. Once an inner loop metric is found the inner loop gains are saved and the inner closed loop state space matrix is augmented with the outer loop state space representation of the plant dynamics. The outer loop optimisation then searches for a second CCM dual metric W as a convex Semi-Definite Programme (SDP) and optimises the decision variables to find a solution for the contraction rate a minimum distance to the reference contraction rate. The outer loop gains are then combined with the saved inner loop gains to create the final controller solution. This is an extension of the technique described in works by [5] using the software Yalmip a Linear Matrix Inequality Parser and Mosek [1] an optimisation solver but is generic to most LMI solvers. The technique is also closely related to the works by [15, 11, 25] in determining an optimisation strategy for manipulating the constraints on the Eigenvalues of the W metric.

Let W be a positive definite symmetric matrix with components w_{11}, w_{12}, and w_{22}, let M be the inverse of W, and let λ be the product of the damping ratio ζ_d and the natural frequency ω_{n_d}. The objective of the optimization problem is to maximise the sum of w_{11} and w_{22}, subject to the constraints that W is positive semi-definite, $AW + WA^T - BB^T + 2\lambda W$ is negative semi-definite, and the eigenvalues values of M are bounded by $1/\bar{\beta}$ and $1/\underline{\beta}$. The optimization problem is formulated as follows:

$$\min_{w_{11}w_{22}\in\mathbb{R}>0} \quad (-w_{11} - w_{22}) \tag{4.31}$$

$$s.t. \quad W \geq 0 \tag{4.32}$$

$$\lambda = \zeta_d \omega_{n_d} \tag{4.33}$$

$$-\left(AW + WA^T - BB^T + 2\lambda W\right) \geq 0 \tag{4.34}$$

$$W = \begin{bmatrix} w_{11} & w_{12} \\ w_{12} & w_{22} \end{bmatrix} \tag{4.35}$$

$$M = W^{-1} \tag{4.36}$$

$$\underline{\sigma}(M) = \frac{1}{\bar{\beta}}(W), \quad \bar{\sigma}(M) = \frac{1}{\underline{\beta}}(W), \tag{4.37}$$

Algorithm 4: Convex Loop Shaping with CCM

Input: A_{inner} and B_{inner} matrix coefficients of the learned dynamics for the inner loop.
$\lambda \leftarrow \zeta_d \omega_{n_d}$ **Set the decay rate to design condition**
for $\zeta_{d_i}\omega_{n_{d_i}}$ **do**
  if $(W_{feas} == true)$ **then**
    $W \leftarrow \{w_{11}, w_{12}, w_{22}\}$ $\leftarrow \textbf{CCM}_{\text{InnerLoop}}$
    $\omega_{n_i} \leftarrow \omega_n(W, A, B)$
    if $\omega_{n_{min}} \leq \omega_{n_i} \leq \omega_{n_{max}}$ **then**
      $K_{\text{OPT}_{\text{inner}}} = 0.5B^T(W)^{-1}$
      $A_{\text{CL}_{\text{inner}}} \leftarrow (A_{\text{inner}} - B_{\text{inner}}K_{\text{OPT}_{\text{inner}}})$
    continue
  else
    $\lambda \leftarrow \zeta_{d_{i+1}}\omega_{n_{d_{i+1}}}$
Input: $A_{\text{closed-loop}_{\text{inner}}}$, A_{outer} and B_{outer}
$A_{\text{outer}} \leftarrow A_{\text{CL}_{\text{inner}}}$
for $i = 1 : 50$ **do**
  $\Delta\lambda = i \times 0.05$
  $\lambda = 0.1 + \Delta\lambda$
  $W \leftarrow \{w_{13}, w_{23}, w_{33}\}$ $\leftarrow \textbf{CCM}_{\text{OuterLoop}}$
  if $(W_{feas} == true)$ **then**
    $M = W^{-1}$
    $ratio \leftarrow \lambda/(\zeta_d\omega_{n_d})$
  else
    $M \leftarrow \mathbb{I}_2$
    $ratio \leftarrow 0$
    continue
Output: W for $\min|ratio - 1|$
$K_{\text{OPT}_{\text{outer}}} = 0.5B^T(W)^{-1}$
$K_{\text{OPT}_{\text{outer}}} \leftarrow K_{\text{OPT}_{\text{inner}}}$
Output: $K_{\text{OPT}_{\text{outer}}}$

4.4.4 Robust CCM and Connections to LQR LTI Controllers

It was shown in [10] that an L_2 gain bound can be defined as a quadratic regulator in combination with the standard formulation of the exponential guarantee for stability of a CCM controller. In this formulation the input output gain is bound via a dissipation inequality. In the most simplified formulation of the nonlinear certification technique as we propose in this study where the outputs are measurable and output disturbances are

treated as inputs to the system dynamics the LTI LMI resembles the LQR architecture for regulating the dynamics. We make a distinction between the RCCM formulation and the LQR formulation by introducing a scaling matrix and a prioritisation matrix for the outputs of the system. Further we test the viability of a novel method for creating a hybrid controller which certifies stability as in the previously proposed RCCM architecture but more explicitly as an LMI combination with mixing parameters. We highlight the short commings of the LMI based approach and the advantage of a linear sum of controllers under certain conditions as a benefit of this approach in offering greater degrees of freedom for the controller while offering a degree of performance with tuning parameters.

Theorem 4.4.1 (Hybrid LMI Controller)**.** *Let $A \in \mathbb{R}^{n\times n}$ be the state matrix, $B \in \mathbb{R}^{n\times m}$ be the input matrix, $W \in \mathbb{R}^{n\times n}$ a positive definite symmetric matrix and $Q \in \mathbb{R}^{n\times n}$ a diagonal matrix with positive entries and invertible. Let $\lambda \in \mathbb{R}_{>0}$ be a positive scalar and $a \in [0, 1]$ be a scalar. If the linear matrix inequality*

$$\begin{bmatrix} WA^T + AW - BB^T + 2\lambda W(1-a) & \sqrt{a}W^T \\ \sqrt{a}W & Q^{-1} \end{bmatrix} < 0$$

holds, then the hybrid controller designed based on this inequality guarantees asymptotic stability of the closed-loop system in the when $0 < a < 1$ and exponential stability when a is equal to zero.

Proof. Considering two cases for the value of a:

1. Case $0 > a \leq 1$: In this case, we have $\bar{W}A^T + A\bar{W} - BB^T < -2\lambda\bar{W}(1-a) - a\bar{W}Q\bar{W}$. Since the inequality $\bar{W}A^T + A\bar{W} - BB^T < \bar{W}Q\bar{W}$ guarantees asymptotic stability and $-2\lambda\bar{W}(1-a) - a\bar{W}Q\bar{W} < -a\bar{W}Q\bar{W}$, it follows that the system remains at least asymptotically stable for $0 < a \leq 1$.

2. Case $a = 0$: In this case, the inequality becomes $WA^T + AW - BB^T < -2\lambda W$, which guarantees exponential stability. Therefore, the system is exponentially stable for $a = 0$.

Hence, the hybrid controller, governed by the parameter a, guarantees both asymptotic stability ($0 < a \leq 1$) and exponential stability when ($a = 0$). □

Remark. *In the case where Q is a diagonal matrix with positive entries, WQW results in a positive definite matrix and all eigenvalues are positive. However, these eigenvalues can approach zero if certain elements in Q are small or if the eigenvectors of W align in certain directions. This can lead to poor scaling of the smallest eigenvalues. So while the positivity of the eigenvalues ensures stability only asymptotic stability is guaranteed.*

The hybrid approach's mixed constraints necessitate a hybrid formulation for the feedback gain matrix K. This formulation involves replacing the 0.5 factor in the conventional expression $K = -0.5B^T M$, formerly used in the CCM technique, with the parameter p. This new parameter varies between values of 0.5 in the limit that the LMI tends to the standard CCM case and 1 in the limit that the LMI tends to the RCCM case as the parameter a varies between 0 and 1.

$$p = a + \frac{(1-a)}{2} \tag{4.38}$$

$$u = v - pB^T M\delta_x \tag{4.39}$$

4.4.5 Scaling and Prioritisation of Output Dimensions

In the standard linear LQR technique the Q matrix accepts arbitrary weightings based on an iterative trial and error testing approach. In the RCCM technique the output matrix C is used to define Q as the square root, $Q = C^T C$. We propose a variation on the standard output matrix by considering the scale and range of the states and a prioritisation matrix to better explain the associated L_2 gain cost function. For the scaling matrix S the range r_i divided by the sensitivity s_i of the state is proposed as the scaling factor which is normalised to keep the largest scaled dimension equal to 1. For the longitudinal controller the α state has a range of 50, being limited to ± 25 degrees and the pitch rate has a range

of 120 being limited to a pitching velocity of ± 60 degrees/s. The sensitivity of the states is given by the B matrix parameters and so the normalised scaling matrix becomes:

$$S = \begin{bmatrix} (\frac{r_1}{s_1})(\frac{s_1}{r_1}) & 0 \\ 0 & (\frac{r_2}{s_2})(\frac{s_1}{r_1}) \end{bmatrix} \tag{4.40}$$

$$= \begin{bmatrix} (\frac{r_1}{B_1})(\frac{B_1}{r_1}) & 0 \\ 0 & (\frac{r_2}{B_2})(\frac{B_1}{r_1}) \end{bmatrix} \tag{4.41}$$

A prioritisation matrix P with a myopic weighting of the identity matrix is given the parameters, P_i.

$$P = \begin{bmatrix} P_1 & 0 \\ 0 & P_2 \end{bmatrix} \tag{4.42}$$

The square of the scaling matrix and prioritisation matrix is then equal to the standard LTI Q matrix in the LQR formulation.

$$Q = P'S'SP \tag{4.43}$$

The RCCM LMI optimisation is then modified to account for the new parameterisation matrices:

$$\begin{bmatrix} WA^T + AW - BB^T + 2\lambda W(1-a) & \sqrt{a}W^T P^T S^T \\ \sqrt{a}SPW & I \end{bmatrix} < 0 \tag{4.44}$$

Figures 4.4.2 to 4.4.5 show the results of a set of control optimisation outputs for a grid of parameters for P1 and P2 and varying hybrid parameter a between 0 and 1 for three different contraction rates in lambda. In our study we fix the prioritisation factors for P1 between values of 1 and 3 and values of P2 between values of 1 and 19 with a step size of 1 in both parameters. Although the values have been found via comparison to the

resulting gains associated with a red bounding box for the damping ratio and natural frequency associated with the design constraints, the range of prioritsation values varies with contraction rate and in proportion to the priority of feedback gains for damping coefficient control with pitch rate feedback and natural frequency control with angle of attack feedback. This can be seen in the grid pattern associated with the LQR like control extreme shown in Fig. 4.4.2. The values therefore for the P matrix offer a more sensible scale compared to the standard Q weighting in LQR control. In the figures the red diamonds represent the controller gains associated with three contraction rates. The grid patterns show the span of the prioritisation values tested for different mixing values of a. It is also possible that the hybrid parameters once data is collected regarding the performance of the controller can be incrementally improved in directions associated with the relevant direction of improvement in performance. For example, a controller suffering from oscillations due to disturbance inputs may incrementally improve robustness by increasing the value of a, and increasing the value of P2 while decreasing the value of P1. In this manner an iterative learning approach could be adopted within the architecture and algorithm framework.

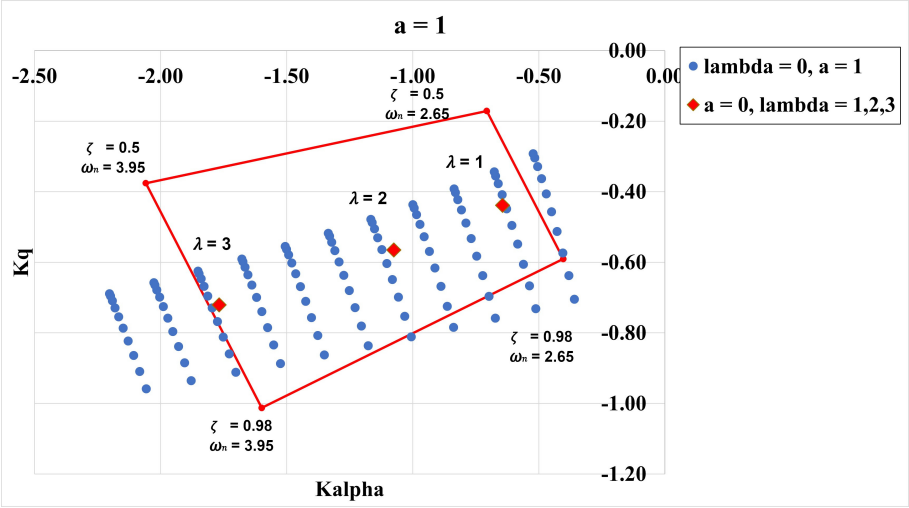


Figure 4.4.2 – Hybrid Controller with a = 1

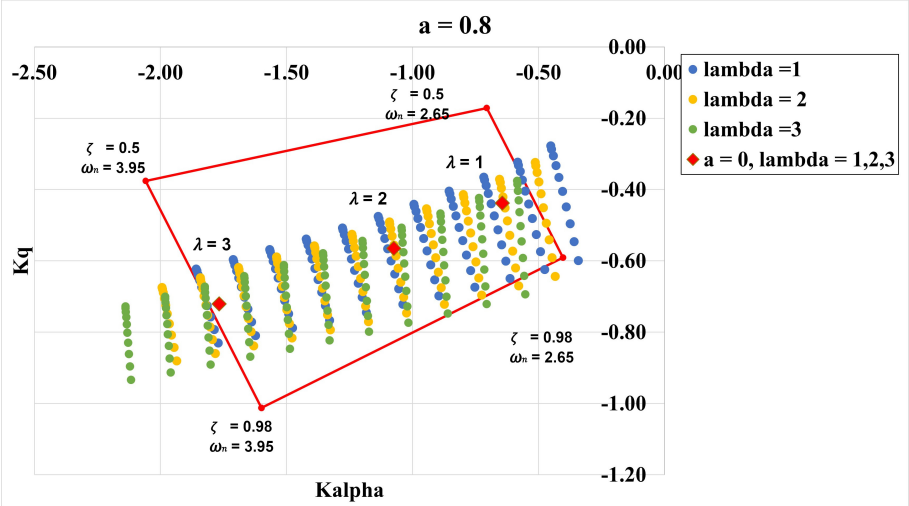


Figure 4.4.3 – Hybrid Controller with a = 0.8

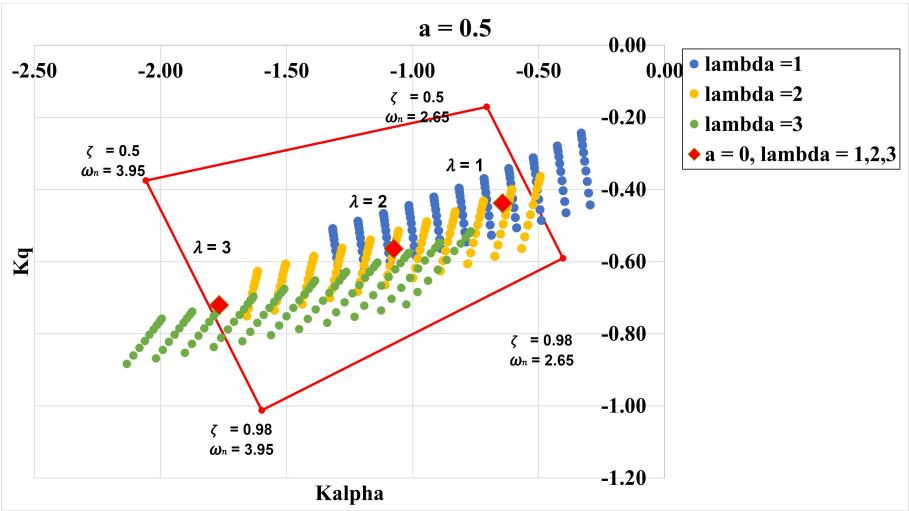


Figure 4.4.4 – Hybrid Controller with a = 0.5

4.4.6 A Hybrid Double Loop

So far we have proposed optimisation of the dual-metric W as a single 3D optimisation using either RCCM LMIs or the standard CCM LMI. We also proposed a two loop solution that better placed the eigenvalues according to an explicit contraction rate in each dimension. A further extension to this approach therefore is to combine the hybrid approach into the double loop formulation to understand how the LMIs might interact together. For this first analysis the hybrid algorithms adopt a weighting $a = 0.5$. The scaling matrix is fixed as $S = [1\,0; 0\,0.04]$ or $S = [1\,0\,0; 0\,0.04\,0; 0\,0\,1]$ in the three state case. The prioriti-

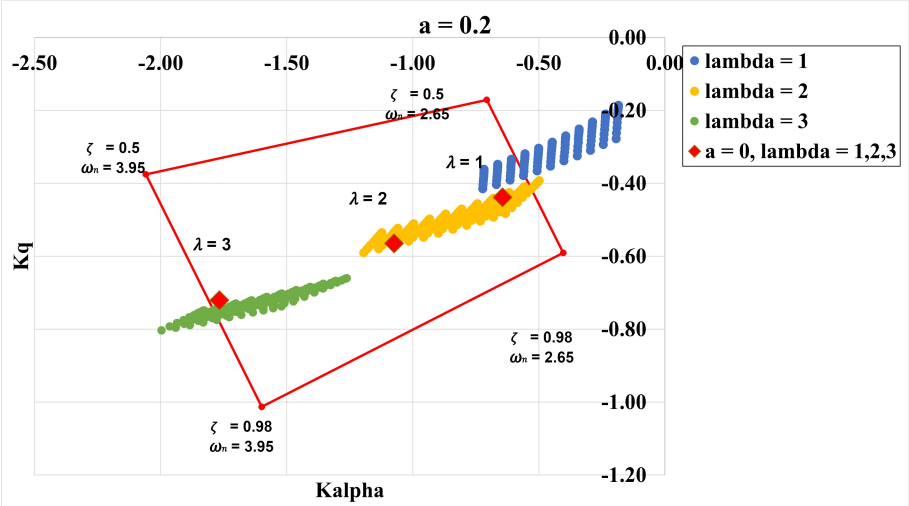


Figure 4.4.5 – Hybrid Controller with a = 0.2

sation matrix is equal to the identity matrix and both optimisation loops adpopt the hybrid RCCM and CCM LMI as in Eqn. 4.44.

4.4.7 A Hybrid Linear Sum

To trade-off performance between two stabilising controllers with different performance objectives an obvious and more attractive combination of controllers to adopt is a linear sum of controllers with a weighting parameter a between 0 and 1:

$$K_{H_{LinSum}} = aK_{lqr} + (1-a)K_{CCM} \tag{4.45}$$

It is well known that for linear systems and two stable controllers that there are no guarantees about the stability for a linear combination of the two, due to the fact that the set of stable matrices is non-convex. For example consider the following case with two 2x2 matrices:

$$A = \begin{bmatrix} -1 & 4 \\ 0 & -1 \end{bmatrix}, \quad B = \begin{bmatrix} -1 & 0 \\ 4 & -1 \end{bmatrix}$$

For matrix A, the eigenvalues are:

$$\lambda_1 = -1 + 2i, \quad \lambda_2 = -1 - 2i$$

For matrix B, the eigenvalues are:

$$\lambda_1 = -1 + 4i, \quad \lambda_2 = -1 - 4i$$

The linear sum of the two matrices, denoted as C:

$$C = A + B = \begin{bmatrix} -2 & 4 \\ 4 & -2 \end{bmatrix}$$

Where the eigenvalues for matrix C, are:

$$\lambda_1 = 2, \quad \lambda_2 = -6$$

The eigenvalues of matrix C have a positive real part and is unstable, whereas the eigenvalues of matrices A and B individually have negative real parts, indicating stability.

4.4.8 Conditions for certifying stability for a linear sum of controllers

We propose under certain conditions that a linear sum of stable controllers is stable for all combinations when the B matrix is negative and the gain matrix K is negative and the system is a single input and hence B and K are both 1-D vectors certifying negative feedback loops. For the conditions specified the case where two stable matrices with off-diagonal terms of opposing magnitude is not possible. Formerly:

Proof. For any two controllers, K_1 and K_2, which are negative vectors, and a negative vector B. For matrices $A - BK_1$ and $A - BK_2$, where A is any full rank square matrix.

If $G_1 = A - BK_1$ is a non-symmetric matrix, hence if there exists at least one pair

of distinct indices i and j such that $g_{ij} \neq g_{ji}$, where g_{ij} represents the element in the i-th row and j-th column of matrix G_1.

$$G_1 = \begin{bmatrix} * & g_{12} & \cdots & g_{1n} \\ g_{21} & * & \cdots & g_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ g_{n1} & g_{n2} & \cdots & * \end{bmatrix}$$

where the diagonal elements (g_{ii}) can be any value and $G_2 = A - BK_2$. It follows that $G_1 \neq G_2^T$ since K_1 is not equal to K_2. The elements of B and K_1 contribute to the asymmetry of $A - BK_1$, making it distinct from its transpose.

Therefore, under the condition that $A - BK_1$ is non-symmetric and all elements of B and K_1 are negative, it can be concluded that $(A - BK_1) \neq (A - BK_2)^T$ for different K_1 and K_2. This proves that under these conditions a linear sum between two controllers K_1 and K_2 will not create a closed loop matrix that lies outside of a convex subset of stable matrices. □

4.4.9 Proof by Inspection of the Linear Sum of Controllers

Since we only provide a partial proof for the stability of the linear sum we further display the properties of the linear sum of two negative controllers via a randomised trial for matrices with the aforementioned properties. Namely a matrix A is randomised with values between -5 and 5, and a B matrix is randomised with values between -0.1 and -5 and two controllers K_1 and K_2 are randomised with values between -0.1 and -5. A third controller that represents the linear sum $K_3 = (K_1 + K_2)/2$ is then designed from the first two. The closed loop eigenvalues for each system is displayed in Fig. 4.4.6. It can be seen in the figure that for all closed loop stable solutions for controllers K_1 and K_2 the linear sum controller K_3 produces poles on their convex hull that are also negative and stable. We trialed over 10,000 instances of randomised matrices with no observed exceptions to the rule. This proof by inspection could be extended to other dimensions or

ranges of state but is left for future work.

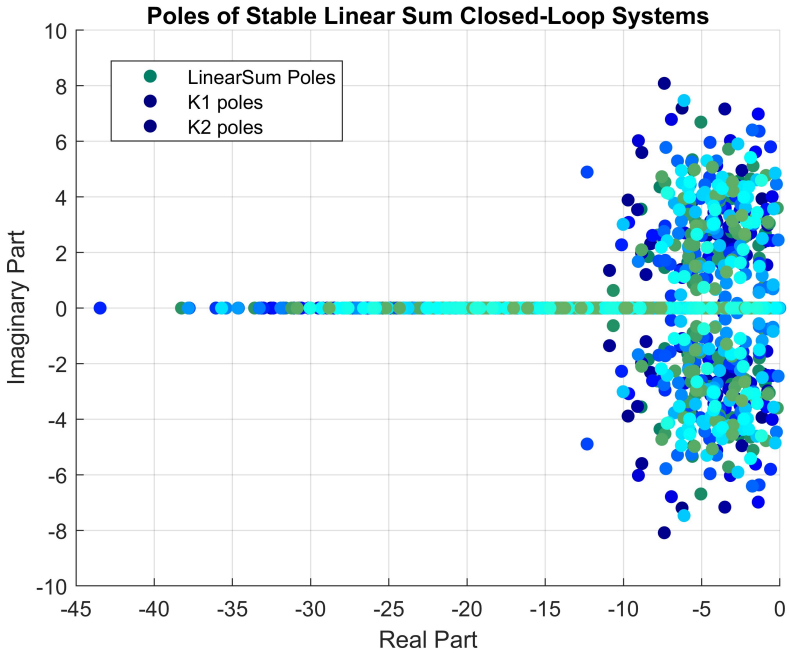


Figure 4.4.6 – Negative Feedback Poles for a Linear Sum Controller vs Controllers K1 and K2

4.5 System Analysis Using the Root Locus Technique

In this section we take a traditional SISO analysis using the root locus technique and transfer functions to create a controller by observing the step response, frequency response and pole locations for a given set of inner loop second order characteristics. The main aims are to:

- Find the maximum outer loop gain to maintain phase margin (PM) and gain margin (GM) limits of 45 degrees and 6 dB, respectively.
- Limit the outer loop gain for a given set of inner loop gains to maintain Level 1 flying qualities with a 30% overshoot and 1-second rise time.
- Investigate the limitations of the 3-state proportional-integral (PI) feedback controller in the absence of using a lead compensator.

$$G_2(s) = \frac{B_q(s - Z_\alpha + B_\alpha K_\alpha) + B_\alpha(M_\alpha - B_q K_\alpha)}{s^2 + s(B_\alpha K_\alpha + B_q K_q - Z_\alpha - M_q) + K_\alpha(Z_q B_q - M_q B_\alpha) + K_q(B_\alpha M_\alpha - Z_\alpha B_q) + Z_\alpha M_q - M_\alpha Z_q} \tag{4.46}$$

To investigate the system properties an inner loop two state transfer function is created for angle of attack and pitch rate dynamics which is then analysed using root locus techniques in MATLAB. Figure 4.5.2 shows the system outer loop controller with actuator dynamics and inner loop transfer functions. The inner loop pitch rate transfer function is derived from the system matrices as follows:

Given the system matrices:

$$A = \begin{bmatrix} Z_\alpha & Z_q \\ M_\alpha & M_q \end{bmatrix}, \quad B = \begin{bmatrix} B_\alpha \\ B_q \end{bmatrix}$$

And the transfer function for the elevator actuator:

$$E(s) = \frac{20.2}{s + 20.2}$$

Compute the desired closed-loop poles:

$$p_1 = -\zeta\omega_n + \omega_n\sqrt{\zeta^2 - 1}, \quad p_2 = -\zeta\omega_n - \omega_n\sqrt{\zeta^2 - 1}$$

A solution for the feedback gain matrix K is found using pole placement and the transfer function for the pitch rate dynamics $G_2(s) = \frac{q(s)}{q_c(s)}$ can be generated from Eqn. 4.46.

The gain K_θ is found by checking that the gain margin is a minimum of 6 dB and the phase margin is a minimum of 45 degrees, the overshoot is less than 30% and the rise time is within 1s. The resulting closed loop transfer function is found from Eqn. 4.47. We analyse set points for the natural frequency and damping ratio characteristics using the military standards range for Level 1 flying qualities. Damping ratio is set to

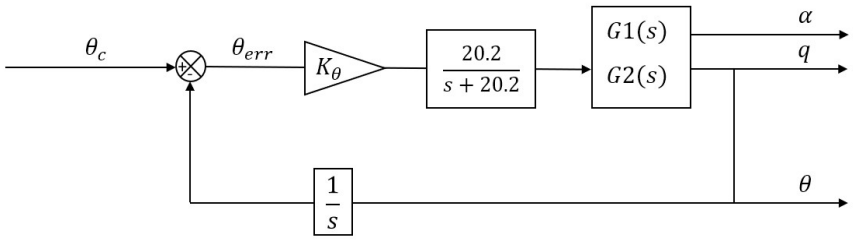


Figure 4.5.1 – Diagram for Root Locus Analysis of Transfer Functions

$0.35, 0.5, 0.7, 0.95, 1.3$ and natural frequency is set to $2, 2.5, 3, 3.6$ rad/s. The upper range for the natural frequency is used because of the rise time requirement of 1s.

$$\frac{\theta(s)}{\theta_c(s)} = \frac{K_\theta E(s) G_2(s)}{1 + \frac{K_\theta}{s} E(s) G_2(s)} \tag{4.47}$$

4.5.1 Results of the Root Locus Analysis

We show a single inner loop pole location and outer loop root locus plot, Fig. 4.5.2 for a damping ratio of 0.7 and natural frequency of 3 rad/s. The locus has been highlighted for the limiting case between overshoot and natural frequency which happens to be the limit of 3.6 rad/s for the natural frequency, giving an outer loop gain of 1.96. Also shown in Fig. 4.5.3 is the bode plot for all inner loop gains tested of the plant model to show the phase margin and gain margin criteria being met for a given choice of outer loop gain. In this example the outer loop gain was set to -3. The step responses are also shown for three different feedback gains for a range of damping ratios and natural frequencies in Figs. 4.5.6,4.5.7. The first figure has an outer loop pole placed at -3 and the second figure a pole at -6. Increasing the outer loop pole reduces the rise time of the step response and the response tends towards the characteristic second order dynamics defined by the inner loop gains. This is more clearly seen in Fig. 4.5.5 for outer loop poles increasing up to a magnitude of -50. The solid green line indicating that a pole of -3 meets the Level 1 rise time criteria of 1s. The results of the root locus analysis are shown in Fig. 4.5.4. The outer loop gain is the maximum gain achievable to meet the phase margin of 45 degrees for each

damping ratio and natural frequency chosen for the inner loop gains. This gives an idea of the limits that might be expected for a controller optimisation using an optimisation technique. By analysing a range of responses against the Level 1 flying qualitites a table can be constructed to identify the optimal operating range for the inner loop and outer loop pole location, Table 4.5.1. From the pass/fail criteria of overshoot and rise time the range of optimum natural frequency and damping ratios for the inner loop is between 0.35 and 0.7 for the damping ratio and between 2.5 and 3.6 for the natural frequency. As the damping ratio increases it becomes increasingly more difficult to achieve the 1s rise time in the absence of additional controller complexity by adding a lead compensator. As the natural frequency reduces it also becomes harder to meet the 1s rise time but the overshoot requirement is easily met until the outer loop pole reaches -10 for a lower damping ratio of 0.35.

4.5.2 Root Locus Controller Design

From the analysis of the pole locations and the root locus feedback gain limits we choose a set of optimal natural frequencies and damping ratios for the aircraft and set of resulting gain designs as shown in Table 4.5.2. From a linear analysis the controllers meet the design criteria but a full nonlinear simulation is required to assess performance as well as a comparison to LQR optimal controller designs and CCM designs which we will explore next.

4.6 Step Response Analysis of Controller Performance

In this study, a nonlinear 3-DOF simulation of an F-16 aircraft, based on NASA wind tunnel data from the study by Nguyen (1979), is utilised to gather trajectory data under various aerodynamic coefficient deviations, as detailed in Appendix 4.A. For each of the controller optimisation methods developed a control policy based on the linear approx-

Table 4.5.1 Analysis of Level 1 Flying Qualities Pass/Fail Criteria for the Inner Loop Damping Ratio and Natural Frequency and Outer Loop Pole Placment

	p3 -2, ζ 0.35		p3 -3, ζ 0.35		p3 -4, ζ 0.35		p3 -6, ζ 0.35		p3 -10, ζ 0.35		p3 -20, ζ 0.35	
ω_n	RT	OS	RT	OS	RT	OS	RT	OS	RT	OS	RT	OS
2												
2.5												
3												
3.6												
	p3 -2, ζ 0.5		p3 -3, ζ 0.5		p3 -4, ζ 0.5		p3 -6, ζ 0.5		p3 -10, ζ 0.5		p3 -20, ζ 0.5	
ω_n	RT	OS	RT	OS	RT	OS	RT	OS	RT	OS	RT	OS
2												
2.5												
3												
3.6												
	p3 -2, ζ 0.7		p3 -3, ζ 0.7		p3 -4, ζ 0.7		p3 -6, ζ 0.7		p3 -10, ζ 0.7		p3 -20, ζ 0.7	
ω_n	RT	OS	RT	OS	RT	OS	RT	OS	RT	OS	RT	OS
2												
2.5												
3												
3.6												

Pass

Fail

p3 - Simple Pole, ζ - Damping Ratio, ω_n - Natural Frequency, RT - Rise Time, OS - Overshoot.

Table 4.5.2 Root Locus Design Options for Controller Gains

ω_n	ζ	K_α	K_q	K_θ
3.6	0.7	-1.47	-0.57	-3.21
3.6	0.5	-1.65	-0.32	-1.69
3	0.7	-0.89	-0.43	-2.29
3	0.5	-1.04	-0.23	-1.21

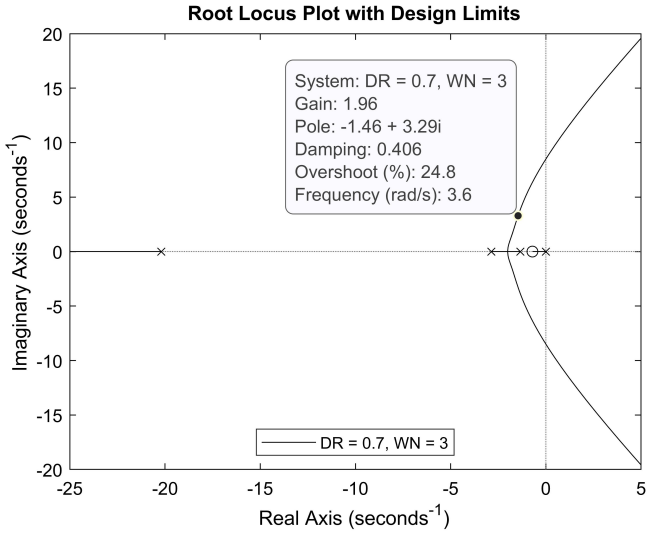


Figure 4.5.2 – Root Locus Analysis of Theta Feedback Gain

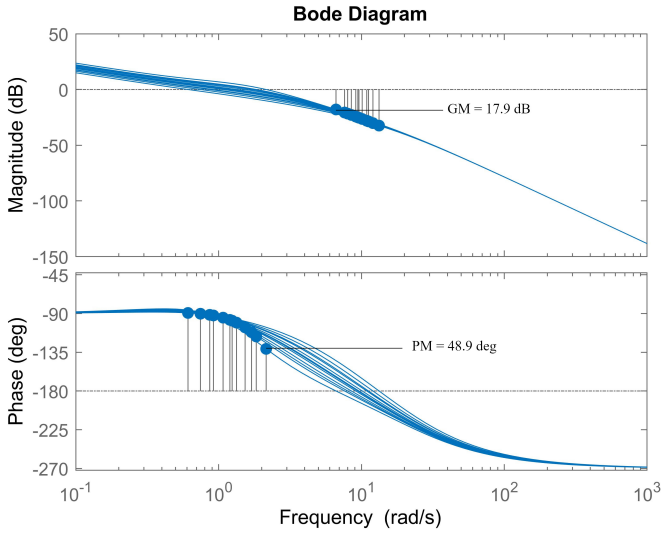


Figure 4.5.3 – Bode Plots for Varying Inner Loop feedback Gains in Ka and Kq

imation of the aircrafts dynamics subject to damage is derived. This policy, expressed as a feedback gain matrix, informs the definition of a transfer function and step response characteristics required for outer loop analysis. Unlike previous studies that concentrated on the inner loop's disturbance rejection capabilities to address short period dynamics, this study employs a step response for the theta command in the outer loop. The step input is a 5-degree theta command, chosen to be large enough to excite the short period

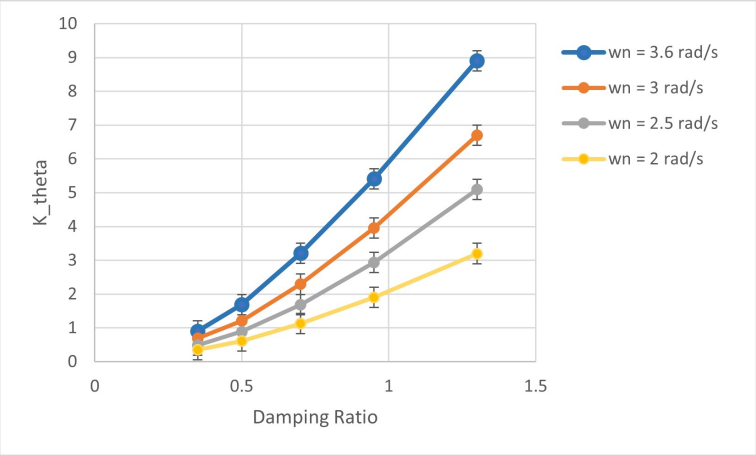


Figure 4.5.4 – Magnitude of Theta Feedback Gain to Meet Gain and Phase Margin Requirements Using a Root Locus Technique. (Error bars indicate that exact PM limits were not used but were within a few degrees of error to sample the Kt values)

mode dynamics, yet small enough to ensure that Phugoid mode dynamics are negligible in the response. This choice also ensures that the dynamics approximation remains valid within the flight envelope used for creating the model approximation and is commensurate with similar studies involing the F-16 short period dynamics, [26, 3]. The output step responses have been normalized to facilitate the assessment of flying qualities. This approach is consistent with flying qualities performance metrics, which are assessed based on step response characteristics. The effectiveness of the outer loop's theta command performance hinges on the precise tracking of attitude commands, which is a direct result of the inner loop's response to short period dynamics. The step responses are evaluated against level 1 and level 2 flying qualities standards. The aircraft test cases considered are detailed in Table 4.6.1.

4.6.1 Analysis of the 6 damaged profiles

The learned state space dynamics from the trajectory data is listed in Appendix 4.B for the seven configurations. The dynamics were derived from a monomial regression using an online convex optimization with L_1 regularisation, isolating the linear terms.

The nonlinear simulation with the nominal root locus designed baseline controller is

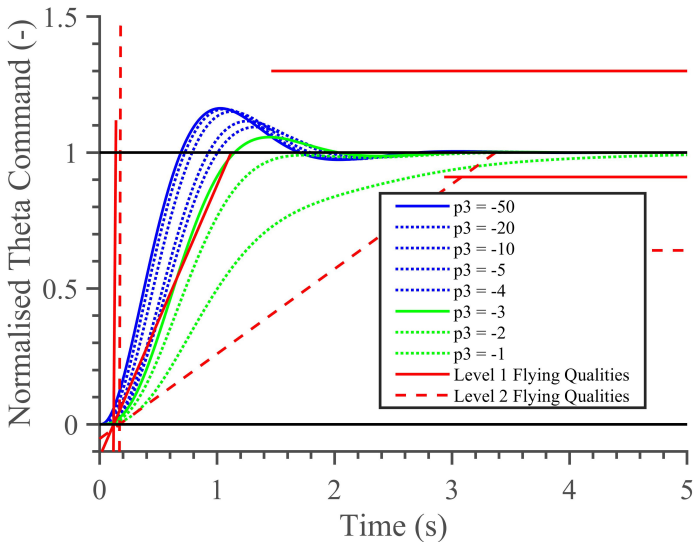


Figure 4.5.5 – Step Response Effect of Increasing Outer Loop Pole Magnitude

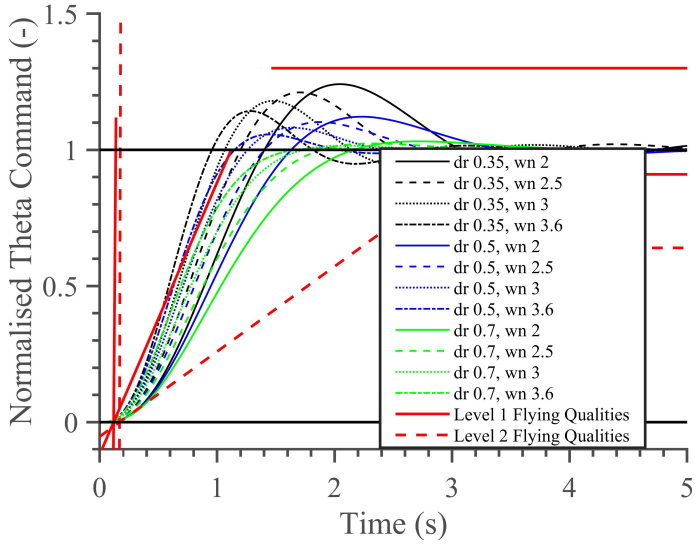


Figure 4.5.6 – Inner Loop Damping Analysis p3 = -3

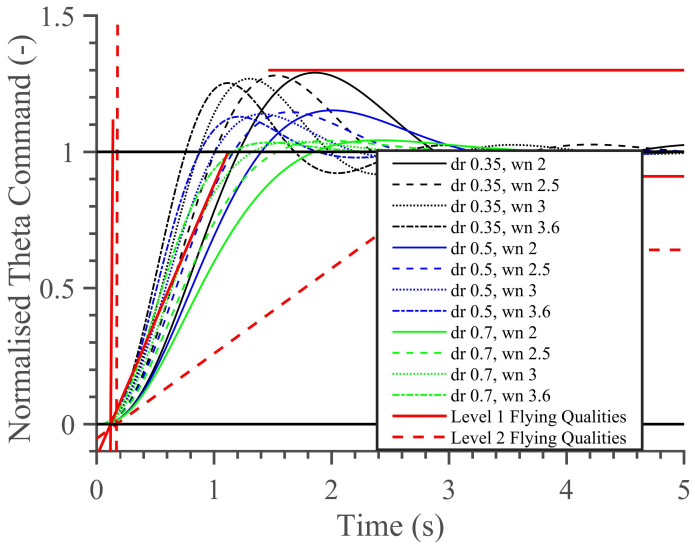


Figure 4.5.7 – Inner Loop Damping Analysis p3 = -6

Aircraft Case	Offset Type (Stability)
Baseline	No changes to coefficients
Damage 1	Nonlinear offset (more stable)
Damage 2	Nonlinear offset (less stable)
Damage 3	Constant offset (more stable)
Damage 4	Constant offset (less stable)
Damage 5	Ramp offset (more stable)
Damage 6	Ramp offset (less stable)

Table 4.6.1 Summary of the Aircraft Aerodynamic Coefficient Modifications between 0 and 20 degrees alpha

used to analyse a step response with the aerodynamic coefficients modified to the damage profiles to show the effect on the proportional feedback controllers response. Figure 4.6.1 shows the step responses for each damaged cases compared to the baseline and Level 1 and Level 2 flying qualities. The baseline step response does not meet the rise time requirement due to the nonlinear dynamics of the system whereas the linear analysis of the response met the requirement adequately. There is also an oscillation after the intial response which means there is a mixed mode in the feedback that could be improved. Looking at the damaged profiles that are more stable all the responses overshoot the set point whereas all the less stable profiles undershoot the set point although the responses have a similar transient dynamics. Although the more stable damaged profiles, (3 and 4), meet the initial rise time and overshoot requirements, all of the damaged profiles lie

outside of the Level 1 flying qualities in terms of a settling time of 3s to be within 5% of the command set point.

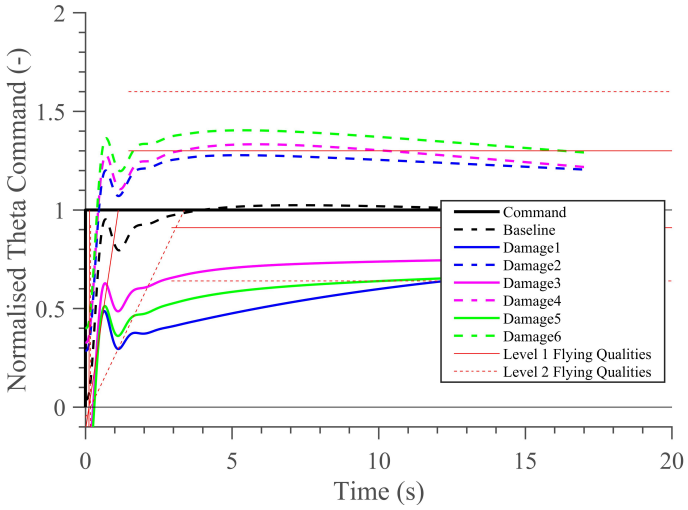


Figure 4.6.1 – Root Locus Controller Subject to Damaged Aircraft Simulation for a Step Size of 5 degrees

4.7 Results

The controller designs are shown in Table 4.7.1 for the baseline case and with the Tables in the Appendix 4.C(4.C.1) to 4.C(4.C.6) for each damaged aircraft case. It can be seen that the CCM double loop controllers have all negative values for the gains which is desirable from a robust point of view that no feedback channel will be contributing to unstable behaviour. The RCCM controllers are also all negative terms whereas the pole placement solution and the single loop optimisation for the CCM controller have positive terms for the alpha feedback. The RCCM theta gain terms directly equal the prioritisation weighting used in the LMI which defines the saturation bound on the elevator actuation. The pole placement gain for theta has the largest gain magnitude out of all controllers. The root locus baseline design being $[-0.89,\ -0.43,\ -2.29]$ correlates most closely with the hybrid linear sum controller.

4.7.1 Linear Approximation Step Responses

Figure 4.7.1 shows the baseline step responses for each controller design. The appendix figures 4.D(a) to 4.D(f) show the step responses for the learned linear dynamics in off-nominal conditions. The step responses are derived using a transfer function from the linear dynamics approximation from trajectory data. The responses are compared to the Level 1 and Level 2 flying qualities rise time and overshoot time domain requirements. It can be seen that the controllers all lie within the Level 1 flying qualities requirements except for the CCM Double Loop and Hybrid Double Loop optimisations that have too large an undershoot due to the oscillatory transient response for the baseline and damaged case 1. Whereas the RCCM controllers are all smooth the nominal RCCM control policy with identity prioritisation matrix has too slow a rise time to meet the level 1 flying qualtities condition. The pole placement controller consistently produces an overshoot consistent with the reference design. The single loop CCM controller has a similar overshoot to the pole placement controller but a slower rise time. Whereas the double loop CCM controller has a similar rise time to the single loop CCM controller but a larger overshoot and undershoot. The hybrid double loop controller has a faster response than the CCM single or double controllers but the overshoot is sometimes greater than the CCM double controllers response and sometimes less than the overshoot of the CCM double controller. The RCCM controllers are both smooth with the optimum controller being the RCCM controller with a theta prioritisation weighting of 4. The RCCM unity controller fails to meet the Level 1 rise time response. It's important to note that although the controllers may appear to perform optimally in the linear case the controller saturation and nonlinear effects must also be taken into consideration to assess the robustness and suitability of each design which is proposed as future work.

Table 4.7.1 Baseline

Test	Ka	Kq	Kt
RCCM 1 0.0016 16	0.111	-1.015	-4.000
RCCM 1 0.0016 1	-0.108	-0.487	-1.000
CCM Single $\lambda = 1.5$	1.134	-0.514	-2.240
CCM Double $\lambda = 1.5$	-0.178	-0.229	-1.478
Hybrid Double $\lambda = 1.5$	-0.394	-0.294	-2.218
LIN Hybrid $\lambda = 1.5$ $\lambda = 1.5$	-0.033	-0.622	-2.739
CCM Single $\lambda = 2.1$	3.85	-0.85	-6.51
CCM Double $\lambda = 2.1$	-0.55	-0.41	-3.80
Place	2.430	-0.833	-4.442

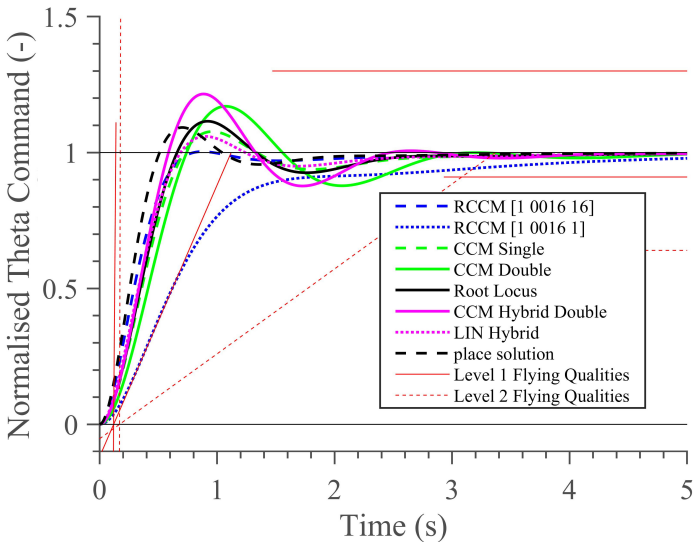


Figure 4.7.1 – Controller Step Responses for the Baseline Case

4.8 Conclusions and Future Work

In this study a set of damaged aircraft coefficient profiles for an F-16 aircraft have been developed and a linear regression performed to approximate the short period mode dynamics as inputs for a control policy optimisation. Six novel control optimisation techniques have been developed and compared to Level 1 and Level 2 flying qualities for 7 different linear approximations of the aircraft dyanamics. The CCM controllers place the poles of the system in a similar arrangement to standard pole placement techniques but the resulting controller has a large overshoot compared to optimal controllers or root locus designs.

The CCM controller also has a positive feedback alpha gain which is undesirable from a robustness perspective. The double loop CCM controller explicitly places both the inner and outer loop poles based on a contraction rate and creates all negative feedback terms but the controller still has a large overshoot and natural frequency which is non-optimal. Robust CCM techniques based on the LQR framework offer smooth transient responses with guarantees on gain and phase margins although are only certified for asymptotic stability a weaker form of stability compared to the exponential stability offered by the CCM controllers. A hybrid LMI framework that trades-off the performance of both RCCM and CCM controllers provides no transient response guarantees for the mixed cases although stability is certifiable the performance is not suitable for flying qualities guarantees. A linear sum hybrid controller under the condition that the B matrix and K matrix are both negative has been certified as stable in mixing an RCCM and CCM controller design. The linear hybrid case therefore provides an effective trade-off of robust performance and exponential stability. Further a parameterisation of the RCCM framework provides for improved scaling in the output dimensions and has been shown to offer greater degrees of freedom in optimisation of the closed loop poles. The extensions and future developments of this baseline approach are:

1. The controllers will be tested with the nonlinear simulation for disturbance inputs to analyse the performance and robustness of each design approach.

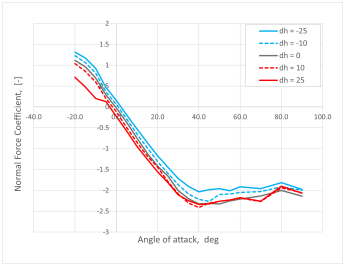
2. In the parameterised RCCM formulation updating the prioritisation matrix with feedback data in a reinforcement style loop could be implemented to adapt the controller to in flight data gathering.

Acknowledgements

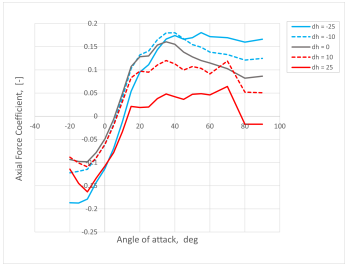
This work was carried out under an Industrial CASE studentship jointly funded by the EPSRC and BAE Systems. The authors would also like to thank Professor Ian Manchester for contributions to the paper.

Appendix

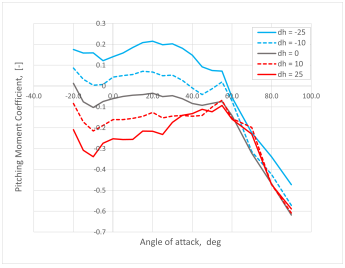
4.A Aerodynamic Coefficient Deviations from Baseline Dynamics



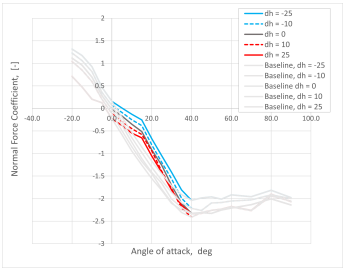
(a) Baseline Cz vs AoA



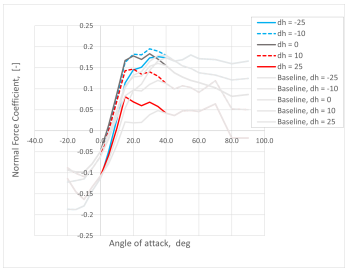
(b) Baseline Cx vs AoA



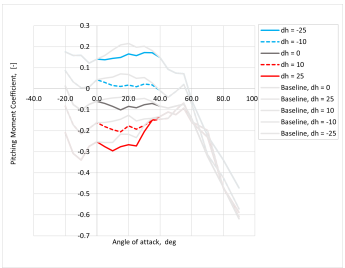
(c) Baseline Cm vs AoA



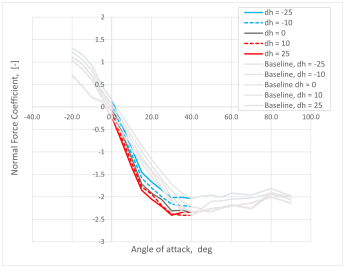
(a) D1 Nonlinear ΔCz, Cz vs AoA



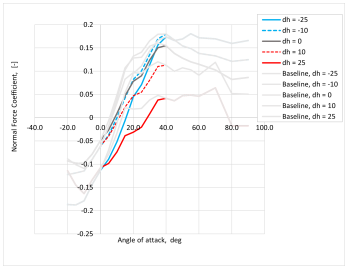
(b) D1 Nonlinear ΔCx, Cx vs AoA



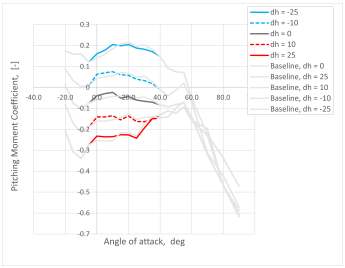
(c) D1 Nonlinear ΔCm, Cm vs AoA



(a) D2 Nonlinear ΔCz, Cz vs AoA

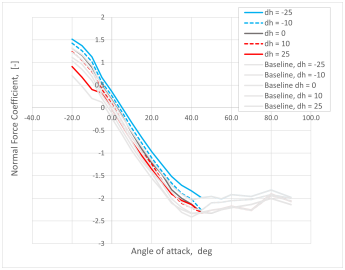


(b) D2 Nonlinear Cx vs AoA

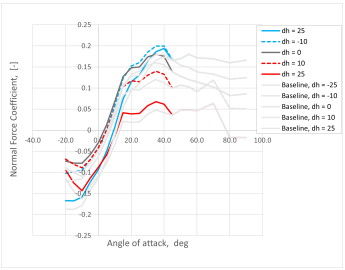


(c) D2 Nonlinear Cm vs AoA

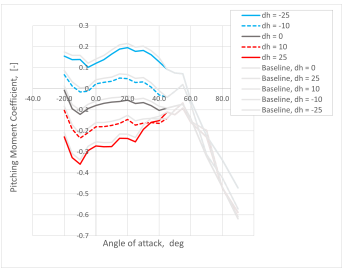
Figure 4.A.1 – Pitching Moment and Axial Aerodynamic Force Coefficients for Varying Angles of Attack (AoA) and Varying Elevator Deflection Angles. Baseline Dynamics and Damage Profiles Baseline to D2



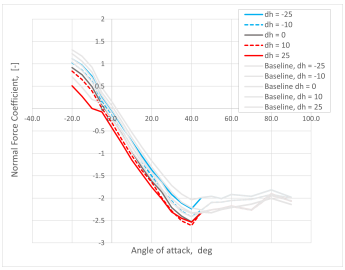
(a) D3 Constant ΔCz, Cz vs AoA



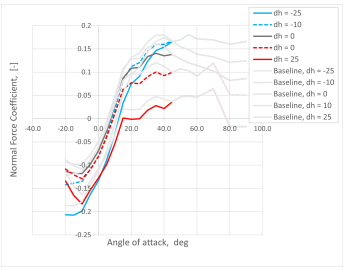
(b) D3 Constant ΔCx, Cx vs AoA



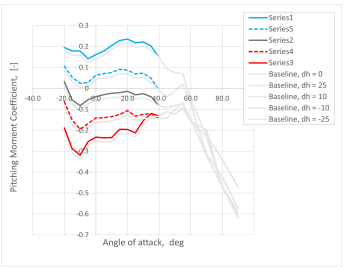
(c) D3 Constant ΔCm, Cm vs AoA



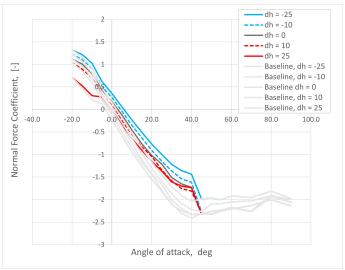
(a) D4 Constant ΔCz, Cz



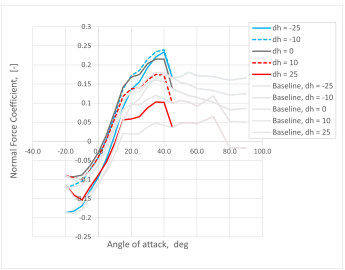
(b) D4 Constant ΔCx, Cx



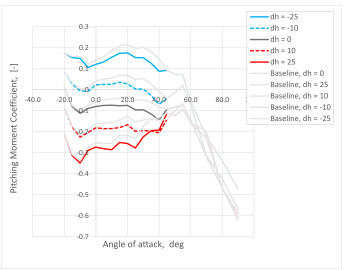
(c) D4 Constant ΔCm, Cm



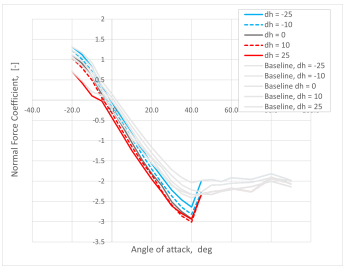
(a) D5 Linear ΔCz, Cz



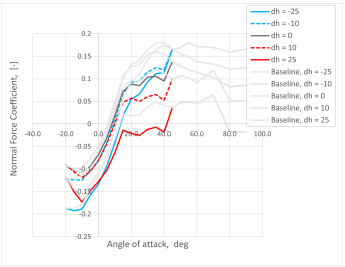
(b) D5 Linear ΔCx, Cx



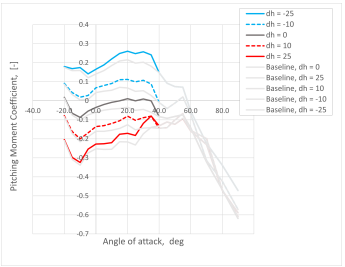
(c) D5 Linear ΔCm, Cm



(a) D6 Linear ΔCz, Cz



(b) D6 Linear ΔCx, Cx



(c) D6 Linear ΔCm, Cm

Figure 4.A.2 – Pitching Moment, and Axial Aerodynamic Force Coefficients for Varying Angles of Attack (AoA) and Varying Elevator Deflection Angles. Baseline Dynamics and Damage Profiles D3 to D6

4.B Linear Approximations for The Short Period Dynamics

Baseline Aircraft:

$$A_1 = \begin{bmatrix} -0.727 & 0.968 & 0 \\ -1.348 & -0.779 & 0 \\ 0 & 1 & 0 \end{bmatrix}, \qquad B_1 = \begin{bmatrix} -0.090 \\ -6.017 \\ 0 \end{bmatrix} \tag{4.48}$$

Damaged Aircraft Case 1:

$$A_2 = \begin{bmatrix} -0.580 & 0.916 & 0 \\ -2.414 & -1.164 & 0 \\ 0 & 1 & 0 \end{bmatrix}, \qquad B_2 = \begin{bmatrix} -0.153 \\ -9.605 \\ 0 \end{bmatrix} \tag{4.49}$$

Damaged Aircraft Case 2:

$$A_3 = \begin{bmatrix} -0.712 & 0.918 & 0 \\ -1.861 & -0.777 & 0 \\ 0 & 1 & 0 \end{bmatrix}, \qquad B_3 = \begin{bmatrix} -0.099 \\ -4.970 \\ 0 \end{bmatrix} \tag{4.50}$$

Damaged Aircraft Case 3:

$$A_4 = \begin{bmatrix} -0.618 & 0.928 & 0 \\ -0.755 & -0.913 & 0 \\ 0 & 1 & 0 \end{bmatrix}, \qquad B_4 = \begin{bmatrix} -0.076 \\ -5.976 \\ 0 \end{bmatrix} \tag{4.51}$$

Damaged Aircraft Case 4:

$$A_5 = \begin{bmatrix} -0.631 & 0.923 & 0 \\ -0.422 & -0.847 & 0 \\ 0 & 1 & 0 \end{bmatrix}, \qquad B_5 = \begin{bmatrix} -0.104 \\ -5.699 \\ 0 \end{bmatrix} \tag{4.52}$$

Damaged Aircraft Case 5:

$$A_6 = \begin{bmatrix} -0.671 & 0.933 & 0 \\ -1.248 & -1.187 & 0 \\ 0 & 1 & 0 \end{bmatrix}, \qquad B_6 = \begin{bmatrix} -0.084 \\ -9.023 \\ 0 \end{bmatrix} \tag{4.53}$$

Damaged Aircraft Case 6:

$$A_7 = \begin{bmatrix} -0.801 & 0.917 & 0 \\ -0.097 & -0.961 & 0 \\ 0 & 1 & 0 \end{bmatrix}, \qquad B_7 = \begin{bmatrix} -0.115 \\ -7.142 \\ 0 \end{bmatrix} \tag{4.54}$$

4.C Comparison of Feedback Gain Matrices for Each Controller Design for Each Aircraft Condition

Table 4.C.1 Damage Case 1

Test	Ka	Kq	Kt
RCCM 1 0.0016 16	0.137	-0.785	-4.000
RCCM 1 0.0016 1	-0.107	-0.369	-1.000
CCM Single $\lambda = 1.5$	1.313	-0.308	-1.921
CCM Double $\lambda = 1.5$	-0.118	-0.118	-1.403
Hybrid Double $\lambda = 1.5$	-0.420	-0.206	-2.105
LIN Hybrid $\lambda = 1.5$	0.009	-0.452	-2.702
CCM Single $\lambda = 2.1$	3.89	-0.54	-5.33
CCM Double $\lambda = 2.1$	-0.37	-0.23	-3.33
Place	2.526	-0.515	-3.636

Table 4.C.2 Damage Case 2

Test	Ka	Kq	Kt
RCCM 1 0.0016 16	0.251	-1.084	-4.000
RCCM 1 0.0016 1	0.013	-0.489	-1.000
CCM Single $\lambda = 1.5$	1.771	-0.641	-3.038
CCM Double $\lambda = 1.5$	-0.239	-0.279	-2.060
Hybrid Double $\lambda = 1.5$	-0.335	-0.286	-3.091
LIN Hybrid $\lambda = 1.5$	0.006	-0.682	-3.030
CCM Single $\lambda = 2.1$	5.58	-1.08	-8.54
CCM Double $\lambda = 2.1$	-0.73	-0.49	-5.09
Place	3.431	-1.036	-5.633

Table 4.C.3 Damage Case 3

Test	Ka	Kq	Kt
RCCM 1 0.0016 16	0.022	-1.011	-4.000
RCCM 1 0.0016 1	-0.186	-0.490	-1.000
CCM Single $\lambda = 1.5$	1.236	-0.512	-2.340
CCM Double $\lambda = 1.5$	-0.207	-0.226	-1.465
Hybrid Double $\lambda = 1.5$	-0.485	-0.307	-2.197
LIN Hybrid $\lambda = 1.5$	-0.092	-0.619	-2.732
CCM Single $\lambda = 2.1$	4.66	-0.86	-7.20
CCM Double $\lambda = 2.1$	-0.62	-0.41	-4.07
Place	3.126	-0.838	-5.197

Table 4.C.4 Damage Case 4

Test	Ka	Kq	Kt
RCCM 1 0.0016 16	-0.027	-1.048	-4.000
RCCM 1 0.0016 1	-0.227	-0.516	-1.000
CCM Single $\lambda = 1.5$	1.142	-0.551	-2.319
CCM Double $\lambda = 1.5$	-0.223	-0.246	-1.388
Hybrid Double $\lambda = 1.5$	-0.515	-0.329	-2.081
LIN Hybrid $\lambda = 1.5$	-0.125	-0.647	-2.694
CCM Single $\lambda = 2.1$	4.62	-0.93	-7.31
CCM Double $\lambda = 2.1$	-0.66	-0.43	-4.03
Place	3.110	-0.903	-5.320

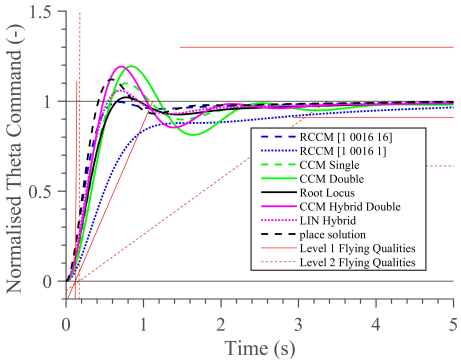
Table 4.C.5 Damage Case 5

Test	Ka	Kq	Kt
RCCM 1 0.0016 16	0.032	-0.816	-4.000
RCCM 1 0.0016 1	-0.186	-0.394	-1.000
CCM Single $\lambda = 1.5$	0.718	-0.299	-1.358
CCM Double $\lambda = 1.5$	-0.097	-0.115	-0.951
Hybrid Double $\lambda = 1.5$	-0.455	-0.226	-1.427
LIN Hybrid $\lambda = 1.5$	-0.033	-0.465	-2.476
CCM Single $\lambda = 2.1$	2.58	-0.52	-4.16
CCM Double $\lambda = 2.1$	-0.34	-0.23	-2.50
Place	1.860	-0.510	-3.177

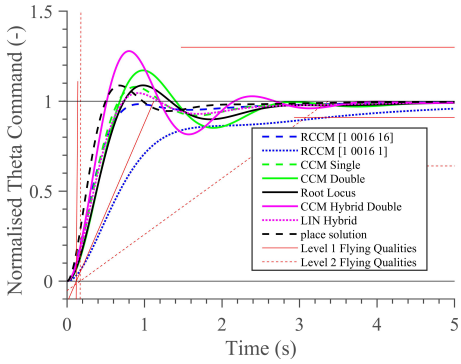
Table 4.C.6 Damage Case 6

Test	Ka	Kq	Kt
RCCM 1 0.0016 16	-0.082	-0.941	-4.000
RCCM 1 0.0016 1	-0.266	-0.468	-1.000
CCM Single $\lambda = 1.5$	0.388	-0.390	-1.200
CCM Double $\lambda = 1.5$	-0.113	-0.157	-0.737
Hybrid Double $\lambda = 1.5$	-0.496	-0.278	-1.106
LIN Hybrid $\lambda = 1.5$	-0.097	-0.549	-2.369
CCM Single $\lambda = 2.1$	2.09	-0.67	-4.08
CCM Double $\lambda = 2.1$	-0.41	-0.31	-2.27
Place	1.567	-0.661	-3.308

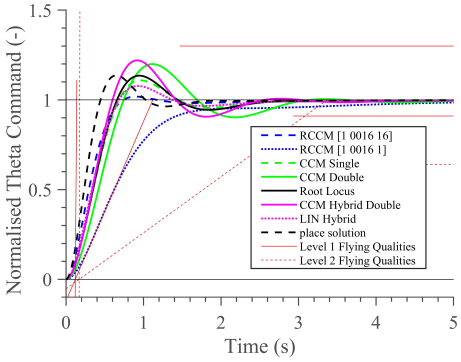
4.D Step Responses for Each Controller Design for the 6 Damaged Aircraft Linear Approximations



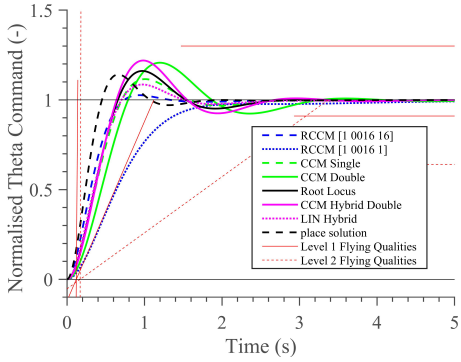
(a) Controller Step Responses for the Damaged Case 1



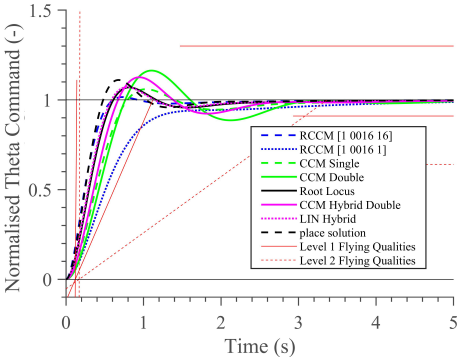
(b) Controller Step Responses for the Damaged Case 2



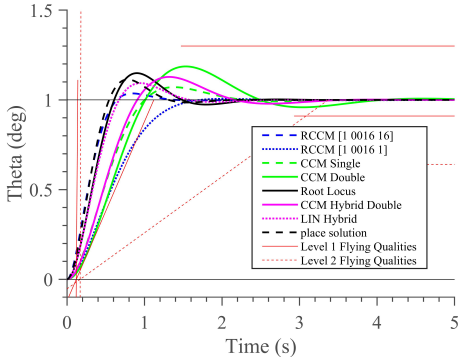
(c) Controller Step Responses for the Damaged Case 3



(d) Controller Step Responses for the Damaged Case 4



(e) Controller Step Responses for the Damaged Case 5



(f) Controller Step Responses for the Damaged Case 6

References

[1] M. Aps. Mosek modeling cookbook, 2020.

[2] M. V. Cook. *Flight dynamics principles: a linear systems approach to aircraft stability and control*. Butterworth-Heinemann, 2012.

[3] T. Keviczky and G. J. Balas. Receding horizon control of an f-16 aircraft: A comparative study. *Control Engineering Practice*, 14(9):1023–1033, 2006.

[4] K. Leung and I. R. Manchester. Nonlinear stabilization via control contraction metrics: A pseudospectral approach for computing geodesics. *Proc. Am. Control Conf.*, pages 1284–1289, 2017.

[5] J. Löfberg. YALMIP: A toolbox for modeling and optimization in MATLAB. *Proc. IEEE Int. Symp. Comput. Control Syst. Des.*, pages 284–289, 2004.

[6] J. Löfberg. Yalmip - decay rate computation in lti system.

[7] I. R. Manchester and J.-J. E. Slotine. Control contraction metrics and universal stabilizability. *IFAC Proceedings Volumes*, 47(3):8223–8228, 2014.

[8] I. R. Manchester and J.-J. E. Slotine. Control contraction metrics, robust control and observer duality. *arXiv*, 2014.

[9] I. R. Manchester and J.-J. E. Slotine. Control contraction metrics: Convex and intrinsic criteria for nonlinear feedback design. *IEEE Transactions on Automatic Control*, 62(6):3046–3053, 2017.

[10] I. R. Manchester and J. J. E. Slotine. Robust control contraction metrics: A convex approach to nonlinear state-feedback control. *IEEE Control Systems Letters*, 2(3):333–338, 2018.

[11] I. R. Manchester, J. Z. Tang, and J.-J. E. Slotine. Unifying robot trajectory tracking with control contraction metrics. In *Robotics Research*, pages 403–418. Springer, 2018.

[12] M. R. Mortazavi and A. Naghash. Pitch and flight path controller design for f-16 aircraft using combination of lqr and ea techniques. *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, 232(10):1831–1843, 2018.

[13] S. N. Sheldon and C. Osmon. Piloted simulation of an f-16 flight control system designed using quantitative feedback theory. *International Journal of Robust and Nonlinear Control: IFAC-Affiliated Journal*, 9(12):841–855, 1999.

[14] S. Singh, A. Majumdar, J. Slotine, and M. Pavone. Robust online motion planning via contraction theory and convex optimization. pages 5883–5890, 2017.

[15] S. Singh, A. Majumdar, J. J. Slotine, and M. Pavone. Robust online motion planning via contraction theory and convex optimization. *Proc. - IEEE Int. Conf. Robot. Autom.*, pages 5883–5890, 2017.

[16] S. Singh, S. M. Richards, V. Sindhwani, J.-J. E. Slotine, and M. Pavone. Learning stabilizable nonlinear dynamics with contraction-based regularization. *The International Journal of Robotics Research*, 2020.

[17] S. Singh, V. Sindhwani, J.-J. E. Slotine, and M. Pavone. Learning stabilizable dynamical systems via control contraction metrics. In *International Workshop on the Algorithmic Foundations of Robotics*, pages 179–195. Springer, 2018.

[18] S. Song, J. F. Whidborne, M. Lone, and A. Molina-Cristobal. Multi-objective optimal longitudinal flight control system design for large flexible transport aircraft. In *2018 UKACC 12th International Conference on Control (CONTROL)*, pages 81–86. IEEE, 2018.

[19] U. M. Specs. Mil-f-8785c: Flying qualities of piloted airplanes. Technical Report November 1980, 1980.

[20] R. M. Stefanescu, C. L. Prioroc, and A. M. Stoica. Weighting matrices determination using pole placement for tracking maneuvers. *UPB Scientific Bulletin, Series D: Mechanical Engineering*, 75(2):31–40, 2013.

[21] B. L. Stevens, F. L. Lewis, and E. N. Johnson. *Aircraft control and simulation: dynamics, controls design, and autonomous systems*. John Wiley & Sons, 2015.

[22] D. Sun, S. Jha, and C. Fan. Learning certified control using contraction metric. In *Conference on Robot Learning*, pages 1519–1539. PMLR, 2021.

[23] H. Tokutake and E. Itoh. Flight controller design taking account of control anticipation parameter. In *Proceedings of the 41st SICE Annual Conference. SICE 2002.*, volume 3, pages 1826–1829. IEEE, 2002.

[24] J. L. Townsend and P. E. Blatt. New mil-f-9490d requirements and implications on future flight control design. *Journal of Aircraft*, 13(9):670–675, 1976.

[25] H. Tsukamoto and S. Chung. Neural contraction metrics for robust estimation and control: A convex optimization approach. *IEEE Control Systems Letters*, 5(1):211–216, 2021.

[26] W. Yang, M. Hammoudi, G. Herrmann, M. Lowenberg, and X. Chen. Two-state dynamic gain scheduling control applied to an f16 aircraft model. *International Journal of Non-Linear Mechanics*, 47(10):1116–1123, 2012.

Chapter 5

Performance Evaluation of 2-Loop CCM and Hybrid RCCM Adaptive Controller

5.1 Introduction

Optimising controllers online for high dimensional systems is inherent to aerospace applications where complex interactions and multivariable system dynamics occur. One modern approach that has guarantees on the feasiblity of solutions and speed of computation is the use of convex optimisation. It is thought that convex optimisation control policies are "the PID controllers of the 21^{st} century" [14]. The benefits of convex solvers are that the solutions are interpretable, the optimisation can incorporate constraints on the system either as invariant sets, inequalities or cost functions, any uncertainties in the dynamics or inputs can be added easily to the optimisation, and the solutions are deterministic. One recent method that exploits these traits in nonlinear control design is the use of control contraction metrics that leverage the use of convex optimisation to generate control certificates. In previous work we developed a learning pipeline adopting a control contraction metric as a certification for stability via convex solvers. We further developed a novel algorithm to explicitly place the closed loop feedback poles of the system relative to flying qualities specifications using a Control Contraction Metric (CCM) optimisation. A method to certify stability adopting a hybrid controller was also explored that traded

off robust control via a Linear Quadratic Regulator (LQR) like framework and exponential control via the CCM controller. In this work we continue the study of the designed controllers by applying them to a nonlinear simulation of an F-16 longitudinal model with 6 different aerodynamic profiles simulating faults to the aircraft. The nonlinear analysis is intended to identify the robustness of each controller design as well as performance in tracking a doublet command and comparison to flying qualities characterstics. In the previous paper we found that a 2-loop CCM controller produced all negative feedback terms whereas a single loop CCM controller created a positive alpha feedback similar to a full state pole placement design. We showed that an LQR like Robust CCM (RCCM) formulation provided more degrees of freedom in optimisation of the control policy via prioritisation weighting of the systems states. A hybrid linear sum controller was also certified as stable under certain conditions on the B matrix and feedback Gain matrix such that a trade-off of robust control and exponential control could be adopted. In this paper we compare the new controllers to a baseline root locus approach and a pole placement technique. The intention is to show the mechanism for controller design from first principles rather than to compare performance with advanced controller techniques. The CCM techniques offer a middle ground in nonlinear control optimisations between LQR techniques and Nonlinear Model Predicitve Control (NMPC) techniques, [7]. The Robust CCM technique is closely related to LQR control and H_∞ control as described in [10].

5.1.1 Approach Overview

In this work a nonlinear F-16 Longitudinal simulator is used to test the adaptive controllers developed subject to deviations from the baseline dynamics. The learning pipeline consists of a batch of 40s of trajectory data excited by a sinusoidal stick input from which a linear approximation of the flight dynamics is formed. The linear approximation is then used to optimise a control policy certified by the CCM controller. A doublet input is used to assess the nonlinear performance of the controller designs. The simulator is based on a previous study by Nguyen at al. [11] that measured the aerodynamic coefficients of an

F-16 from wind tunnel data. The 6 damaged aircraft profiles relate to safe controllable deviations from the baseline aerodynamic coefficients, taking the aircraft out the nominal Level 1 flying qualtities. The deviations are intended to show how the performance of the aircraft can be improved without the need for recovery or control allocation techniques. Our approach aims to provide a methodology that can be tailored to other fixed wing aircraft.

5.1.2 Related Work

Prior research most aligned to this study is that of [6] where the authors use adaptive backstepping for a longitudinal controller and ensure adequate short period mode properties by selection of model parameters via a performance specification model. The approach covers a wide range of the flight envelope and the authors cite that the adaptive control law is much simpler and easier to construct compared to other approaches. Backstepping is also adopted by Sonneveldt et al. [12] to update control laws via B-Spline neural networks scheduled at different flight conditions. The authors demonstrate the robustness and handling qualities of their approach in the presence of uncertain aerodynamic parameters. Previous work that separates adaptive control for the inner and outer loop of a helicopter attitude and trajectory was carried out by Johnson et al. [5]. A Pseudo Control Hedging technique is used in the inner loop to avoid incorrect adaptation when control limits are reached. The approach also considers the correct placement of poles and the mitigation of inner and outer loop interactions. Pole placement using LMIs was proposed by Chilali et al. [2] as complex set constraints to bound the stability properties of dynamic systems. The pole constraints ensure that guarantees are met for transient response characteristics. The approach also provides robust pole placements as LMI optimisations. A useful reference for static feedback LMI strucutres and definitions is the work of Boyd et al. [1] and an extensive overview of convex optimisation applications including the SpaceX rocket landing guidance and other trajectory optimisation use cases is presented in [8]. Static feedback controllers for nonlinear systems have been explored in

recent work adopting state dependent Ricatti Equations (SDRE). The technique is thought to outperform nonlinear control techniques and is an active area of research for nonlinear control problems [3]. The authors in [4] compare LQR and SDRE for a high performance aircraft in longitudinal flight at high angles of attack where nonlinearities start to occur. They showed that the SDRE control was able to return to safe flight conditions where the LQR controller failed. Previous work on control contraction metrics relevant to this study include the works by Manchester et al. [9],[10] and Tsukamoto et al. [13].

5.1.3 Statement of Contributions

The CCM controllers offer exponential control policies for rise time performance whereas the Robust CCM framwork is an optimal control policy that uses a quadratic cost on the states of the system that can be weighted to achieve certain transient performance properties more so than the CCM controller. The hybrid controllers offer a trade-off between the two controllers to chose to prioritise performance over robustness. The aim is to see how the controllers respond in a nonlinear scenario with variations in the nonlinear deviations but using a linear approximating function to derive the controllers. The controllers that best reduce steady state errors, tracking errors and maintain stability in all test conditions are the most suitable.

5.1.4 Statement of Contributions

The main contributions are:

1. An analysis of the robustness properties of six novel controllers adopting a CCM, RCCM and a Linear Sum and Hybrid LMI framework. Our analysis fills a gap in the existing literature by examining six novel controllers under varying nonlinear scenarios.

2. A comparison of two Hybrid approaches in mixing CCM optimisation with RCCM controllers. For each approach an iteration of the mixing parameter in 10 intervals

is compared with the mid-point and bounds highlighted to provide an insight into the mixed transient performance properties for each approach.

3. A comparison to a baseline root locus designed controller and pole placement approach offers a benchmark for evaluating the efficacy of new control strategies.

5.1.5 Layout of the paper

The rest of this paper is structured as follows: Section 5.2 provides a description of the nonlinear simulation and the test methodology to analyse the controllers, Section 5.3 is a review of the algorithms and LMIs used for each controller, Section 5.4 presents the results of the study, Section 5.5 is a discussion and Section 5.6 is the conclusions and proposed future work.

5.2 Testing Methodology Damage Profiles and Simulation Technique

For this study a nonlinear 3-DOF simulation of an F-16 aircraft, created using NASA wind tunnel data from the study by [11] is used to test the adaptive controllers. This work adopts a new input test cases in the form of a 5 degree pitch attitude doublet disturbance where the doublet rise time reduces from $\Delta t = 2s$ in the first period to an impulse step in the second period. The impulse duration represents both a pilot input and a sudden atmospheric distubance to test the control saturation limits. The simulator includes actuator dynamics for the elevator and operates under a constant thrust input for steady level flight. For this study the learning pipeline is reduced to a hard-coded set of linear approximations manually activated and derived from 40 seconds of flight data under 7 different aircraft aerodynamic profiles described in the Appendix 4.A. The damaged profiles are deviations of the baseline aerodynamic coefficients between 0 and 20 degreees angle of attack where the nominal coefficients exhibit fairly linear behaviour. The deviations simulate damage

that would result in three types of behaviour, a constant offset, a ramped offset and a nonlinear offset in with an inflection point maximum at the mid range. The offsets are then both positive and negative to create profiles that cause the aircraft to be more dynamically stable and less dynamically stable in the open loop dynamics respectively. The rationale behind the damage profiles is to create varying degrees of dynamics complexity that takes the aircraft out of the level 1 flying qualities at low angles of attack to assess the learning algorithms ability to modify the control policy to maintain performance during the short period mode oscillations. We list the 7 aircraft test cases summarised here in Table 5.2.1.

Aircraft Case	**Offset Type (Stability)**
Baseline	No changes to coefficients
Damage 1	Nonlinear offset (more stable)
Damage 2	Nonlinear offset (less stable)
Damage 3	Constant offset (more stable)
Damage 4	Constant offset (less stable)
Damage 5	Ramp offset (more stable)
Damage 6	Ramp offset (less stable)

Table 5.2.1 Summary of the Aircraft Aerodynamic Coefficient Modifications between 0 and 20 degrees alpha

The equations of motion for the longitudinal aircraft dynamics are:

$$\begin{aligned}
\dot{u} &= -qw - g\sin\theta + \frac{\bar{q}S}{m}C_{X_t}(\alpha, q, \delta_h) + \frac{F_T}{m} \\
\dot{w} &= qu + g\cos\theta + \frac{\bar{q}S}{m}C_{Z_t}(\alpha, q, \delta_h) \\
\dot{\theta} &= q \\
\dot{V_T} &= \frac{(u\dot{u} + w\dot{w})}{V_T} \\
\dot{\alpha} &= \frac{u\dot{w} - w\dot{u}}{u^2 + w^2} \\
\dot{q} &= \frac{C_{m_t}(\alpha, q, \delta_h)\bar{q}S\bar{c}}{I_{yy}}
\end{aligned} \tag{5.1}$$

where $u, w, \theta, V_T, \alpha$ and q are the aircraft's body x-axis velocity, body z-axis velocity, pitch angle, total velocity, angle-of-attack and pitch-rate respectively. F_T is the single

engine thrust force. m is the mass of the aircraft and I_{yy} the moment of inertia about the y-axis. $\bar{q}$ is the dynamic pressure which is equal to $\frac{1}{2}\rho V_T^2$. S is the wing area and $\bar{c}$ is the mean aerodynamic chord.

Table 5.2.2 Aircraft Properties

Parameter	Symbol	Value	Units
Mass	m	9295.44	kg
Wing Area	S	27.87	m^2
Mean Aerodynamic Chord	$\bar{c}$	3.45	m
Moment of Inertia	I_{yy}	75673.6	$kg.m^2$
Ref CG location	x_{cgr}	$0.35\bar{c}$	m
CG location	x_{cg}	$0.3\bar{c}$	m

The aerodynamic force coefficients in the body axis are C_{Z_t}, C_{X_t} and the aerodynamic moment coefficient is C_{m_t}. All coefficients are functions of the states α and q and the elevator deflection angle δ_h. Values for the coefficents are taken from the high-fidelity model tables listed in [11].

$$
\begin{aligned}
C_{X_t}(\alpha, q, \delta_h) &= C_X(\alpha, q, \delta_h) + \frac{\bar{c}q}{2V} C_{X_q} \\
C_{Z_t}(\alpha, q, \delta_h) &= C_Z(\alpha, q, \delta_h) \frac{\bar{c}q}{2V} C_{Z_q} \\
C_{m_t}(\alpha, q, \delta_h) &= C_m(\alpha, q, \delta_h) + C_{Z_t}(x_{cgr} - x_{cg}) + \frac{\bar{c}q}{2V} C_{m_q}
\end{aligned}
\tag{5.2}
$$

The aerodynamic force coefficents C_Z, C_X and pitching moment coefficient C_M are functions of the the states α, q where higher order terms, $\dot{\alpha}, \dot{q}$, are neglected. The simulation also calculates the coefficients as a nonlinear function of the control input from the elevator δ_h. Implemented into the model are lookup tables with breakpoints taken from the NASA wind tunnel study in α and δ_h so that the test case damage profiles can be switched programmatically in the tests.

The simulator passes the learned dynamics parameters to the CCM optimization certificate. This certificate assures that a stable policy is passed onto the adaptive controller Fig. 5.2.1.

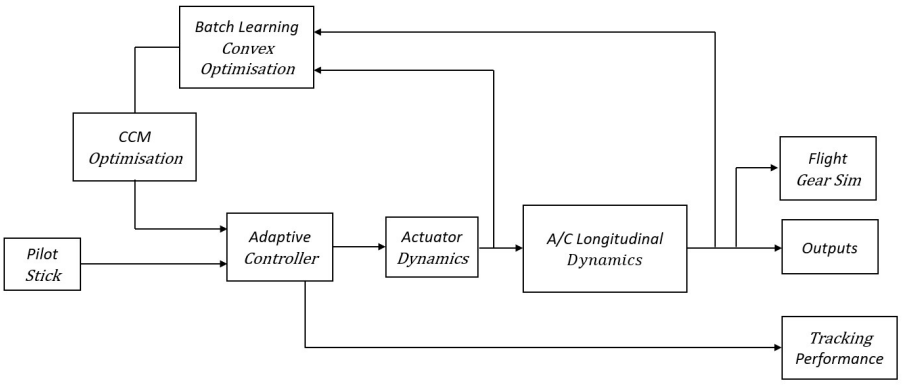


Figure 5.2.1 – Simulator Block Diagram

The adaptive controllers pass learned controller gains to a memory store that updates the proportinal feedback attitude hold controller depicted in Fig. 5.2.2.

5.3 CCM and RCCM Linear Matrix Inequality Optimisation and Algorithmic Controller Designs

There are a total of six designs tested for the control policies adopting variations of the CCM LMI using the parameters shown in Table 5.3.1. The first two controllers are the Robust CCM controllers with quadratic weightings. The first with a weighting of 4 for

Table 5.2.3 Simulation Trim Settings

Parameter	Symbol	Value	Units
Altitude	h	$4572\ (15,000)$	$m\ (ft)$
Velocity	V_T	$152.4\ (500)$	$m/s\ (ft/s)$
Elevator Trim	$\delta_{h_{tr}}$	-4.55	deg
AoA Trim	α_{tr}	4.47	deg
Pitch Rate Trim	q_{tr}	0	deg/s
Thrust Trim	F_T	9433	N

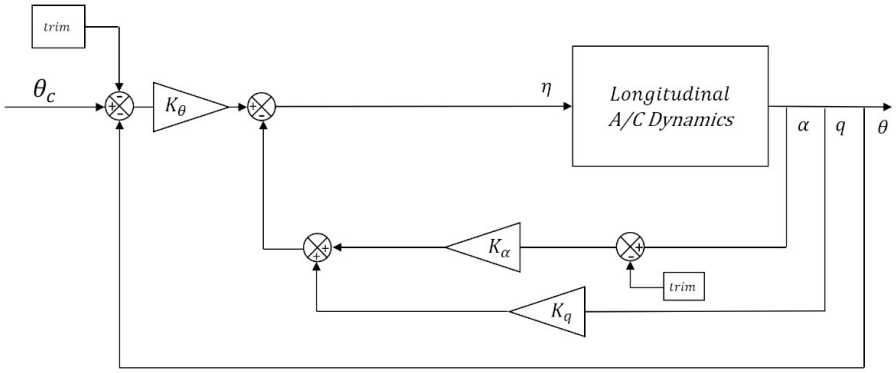


Figure 5.2.2 – Two-Loop Attitude Hold Feedback Controller in Simulator Setup with Trim Correction

the theta channel and the second with unity weighting for all states but both controllers are scaled by a scaling matrix S. The second two controllers tested are the CCM controllers that certify exponential decay of the tracking offset. The single loop as bounds on all states and the double loop explicitly for the inner and outer loops via the feedback gains. The hybrid controllers are tested to understand how a mixed controller between the robust and exponential designs performs depending upon the optimisation. The hybrid LMI controller mixes the linear sum as constraints directly in the LMI optimisation whereas the linear sum mixes the controller gain outputs from separate designs under certain constraints. The hybrid LMI is also designed with both the single loop solution CCM and double loop solutions but the single loop solution is not presented in this paper. The output controller gains for each aircraft test case are listed in Appendix 4.B.

We describe the structure of each linear matrix optimisation here. The algorithms used to optimise the controller gains for this study are derived from the standard static CCM LMI formulation, Equation (5.3).

Table 5.3.1 Parameters Used in Controller Designs

Controller	λ	S	P	a
RCCM_1_0016_16	-	$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 0.04 & 0 \\ 0 & 0 & 1 \end{bmatrix}$	$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 4 \end{bmatrix}$	-
RCCM_1_0016_1	-	$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 0.04 & 0 \\ 0 & 0 & 1 \end{bmatrix}$	$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$	-
CCM Single	1.5	-	-	-
CCM Double	1.5	-	-	-
Hybrid Single	1.5	$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 0.04 & 0 \\ 0 & 0 & 1 \end{bmatrix}$	$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$	0.5
Hybrid Double	1.5	$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 0.04 & 0 \\ 0 & 0 & 1 \end{bmatrix}$	$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 4 \end{bmatrix}$	0.5
LIN Hybrid	1.5	$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 0.04 & 0 \\ 0 & 0 & 1 \end{bmatrix}$	$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$	0.5

$$\begin{aligned} & \min_{w_{ij} \in \mathbb{R}^{nxn}_{>0}} && -trace(W) \\ s.t. \quad & W \geq 0 \\ & AW + WA^T - BB^T \leq -2\lambda W \end{aligned} \tag{5.3}$$

where A is the Jacobian of the linear dynamics of the system and B is the control effectiveness matrix. The optimisation involves searching for a dual-metric $W = M^{-1}$, a positive definite symmetric matrix, scaled by a positive coefficient lambda that signifies the rate of contraction, where the metric W measures the negative definite properties of the closed

loop dynamics. If the closed loop dynamics in the metric are negative definite by more than $2\lambda W$ then the system is said to be exponentially stable with a contraction rate of λ in the metric. Closed loop feedback gains for the system are found from $K = \frac{1}{2}B^T M$. The controller creates a differential feedback with static gain matrix K:

$$u(t) = -K\delta x(t) \tag{5.4}$$

The RCCM LMI derived from [10] used in this study is modified from Eqn. (5.5):

$$\begin{bmatrix} \mathcal{W} & (CW + DY)^T \\ (CW + DY) & I \end{bmatrix} \leq 0 \tag{5.5}$$

where $\mathcal{W} = -\dot{W} + WA^T + AW + BY + B^T Y^T + \frac{1}{\alpha^2} HH^T$

where the $C \in \mathbb{R}^{p\times n}$ matrix represents the state space output matrix, $D \in \mathbb{R}^{p\times m}$ the feedforward matrix and $Y \in \mathbb{R}^{m\times n}$ a gain matrix for the system, where $x(t) \in \mathbb{R}^n$, $u(t) \in \mathbb{R}^m$ and $y(t) \in \mathbb{R}^p$. Since $H = 0$, $D = 0$ and by Finsler's lemma the $BY + Y^T B^T$ becomes ρBB^T. The ρ term can be absorbed into the W metric since for static time invariant metrics $\dot{W}$ is zero, the inequality becomes Eqn. (5.6):

$$\begin{bmatrix} WA^T + AW - BB^T & W^T C^T \\ CW & I \end{bmatrix} < 0 \tag{5.6}$$

where equating C to SP we have:

$$\begin{bmatrix} WA^T + AW - BB^T & W^T P^T S^T \\ SPW & I \end{bmatrix} < 0 \tag{5.7}$$

A mixed constraint hybrid LMI also tested includes a linear weighting parameter a scaled between 0 and 1, Eqn. (5.8):

$$\begin{bmatrix} WA^T + AW - BB^T + 2\lambda W(1-a) & \sqrt{a}W^T P^T S^T \\ \sqrt{a}SPW & I \end{bmatrix} < 0 \tag{5.8}$$

In the hybrid LMI the gain matrix is also modified with a parameter p scaled by the mixing parameter a.

$$p = a + \frac{(1-a)}{2} \tag{5.9}$$

$$u = -pK\delta_x(t) \tag{5.10}$$

The single loop CCM controller is described by Algorithm 5. In the single loop the three state A and B matrix values are input to the optimisation along with the first iteration of the contraction rate λ. If a feasible dual-metric W is found and the natural frequency is within bounds then a controller gain is stored and indexed in a table lookup along with a ratio qualifying the distance the controller is from a designated design condition. For a given range of λ the algorithm outputs the controller for the indexed ratio closest to the design goal. In the double loop algorithm, described in Algorithm (6), the inner loop A and B matrices are first passed to a two state optiisation for the metric W and the resulting gains saved as the inner loop values. Then the closed loop inner system derived from $A_{closedloop_{inner}} = A_{inner} - B_{inner}K_{OPT_{inner}}$ is passed to a second CCM optimisation for the outer loop metric. The outer loop gain for the third state is combined with the gains for the inner loop states to create the final controller.

Algorithm 5: Single Loop Optimisation CCM

Input: A and B matrix coefficients of the learned dynamics; $\zeta_d\omega_{n_d}$ the decay rate design condition
for $\lambda_i = (\zeta\omega)_{min} : \Delta\lambda : (\zeta\omega)_{max}$ **do**
  if $(W_{feas} == true)$ **then**
    $W \leftarrow \{w_{11}, w_{12}, w_{22}\}$ $\leftarrow$ **CCM**
    $\omega_{n_i} \leftarrow \omega_n(W, A, B)$
    if $\omega_{n_{min}} \leq \omega_{n_i} \leq \omega_{n_{max}}$ **then**
      $K_{\text{OPT}_i} = 0.5B^T(W)^{-1}$
      $ratio_i \leftarrow \lambda_i/(\zeta_d\omega_{n_d})$
    else
      $\lambda_i \leftarrow \lambda_{i+1}$
  else
    $\lambda_i \leftarrow \lambda_{i+1}$
Output: K_{OPT_i} for i $\min |ratio_i - 1|$

5.4 Results

5.4.1 Nonlinear Simulation Doublet Responses

In the nonlinear simulation the controllers are first compared in the baseline aircraft condition to compare the performance to the root locus design Fig. 5.4.1. With the CCM controller contraction rate of 2.1 the baseline tracking performance has a smaller error for the double loop but the single loop has a small overshoot for the first more gradual doublet. When the CCM controller is subject to an impulse step, however the gains are too high and the elevator becomes saturated causing an unstable oscillation. For the remaining CCM controllers therefore the contraction rate is reduced from a value of 2.1 to 1.5.

Figure 5.4.2 compares the CCM controllers with a contraction rate of 1.5. Here we are just looking at the performance of the CCM techniques compared to the RCCM techniques with the two prioritisation weightings. Although the CCM controllers outperform the RCCM unity weighting controller for rise time they both exhibit an undesirable mixed modal response that oscillates at the set point. The Double CCM controller has less over-

Algorithm 6: Double Loop Optimisation CCM

Input: A_{inner} and B_{inner} matrix coefficients of the learned dynamics for the inner loop; $\zeta_d \omega_{n_d}$ the decay rate design condition
for $\lambda_i = (\zeta\omega)_{min} : \Delta\lambda : (\zeta\omega)_{max}$ **do**
  if $(W_{feas} == true)$ **then**
    $W \leftarrow \{w_{11}, w_{12}, w_{22}\} \quad \leftarrow \textbf{CCM}_{\text{InnerLoop}}$
    $\omega_{n_i} \leftarrow \omega_n(W, A, B)$
    if $\omega_{n_{min}} \leq \omega_{n_i} \leq \omega_{n_{max}}$ **then**
      $K_{\text{OPT}_{\text{inner}}} = 0.5B^T(W)^{-1}$
      $A_{\text{CL}_{\text{inner}}} \leftarrow (A_{\text{inner}} - B_{\text{inner}} K_{\text{OPT}_{\text{inner}}})$
    continue
  else
    $\lambda_i \leftarrow \lambda_{i+1}$
Input: $A_{\text{closed-loop}_{\text{inner}}}$, A_{outer} and B_{outer}
$A_{\text{outer}} \leftarrow A_{\text{CL}_{\text{inner}}}$
for $\lambda_i = (\zeta\omega)_{min} : \Delta\lambda : (\zeta\omega)_{max}$ **do**
  $W \leftarrow \{w_{13}, w_{23}, w_{33}\} \quad \leftarrow \textbf{CCM}_{\text{OuterLoop}}$
  if $(W_{feas} == true)$ **then**
    $M = W^{-1}$
    $ratio_i \leftarrow \lambda_i/(\zeta_d \omega_{n_d})$
    $K_{\text{OPT}_i} = 0.5B^T(W)^{-1}$
    $ratio_i \leftarrow \lambda_i/(\zeta_d \omega_{n_d})$
  else
    $\lambda_i \leftarrow \lambda_{i+1}$
Output: $K_{\text{OPT}_{\text{outer}\,i}}$
$K_{\text{OPT}_{\text{outer}\,i}} \leftarrow K_{\text{OPT}_{\text{inner}\,i}}$
Output: $K_{\text{OPT}_{\text{outer}\,i}}$ for i $\min |ratio_i - 1|$

shoot on the negative step input compared to the Single CCM controller but oscillates more at the impulse step command compared to the Single CCM controller. Both RCCM controllers have smooth transitions with the highest weighted RCCM controller following the command input precisely. The root locus design performs somewhere inbetween the performance of both RCCM controllers with some multimodal characteritics also similar to the more extreme case of the double CCM controller.

In Figure 5.4.3 we have highlighted the effect of increasing the contraction rate used in the optimisation of the CCM controllers from a value of 0.8 to 2 in 0.2 increments. This highlights the sensitivity in the single loop CCM to excitation of the modal response

whereas the double loop CCM is less sensitive and follows the command signal with less tracking error. This validates somewhat the benefits of a more explicit eigenvalue constraint per loop in the optimisation, although the oscillation is still not as smooth as the RCCM controllers.

In Figure 5.4.4 we have applied the root locus controller design to the airframe and then simulated each aircraft damaged condition with no adaptation in the controller gains. This shows the situation where the aircraft would benefit from controller redesign to better tune the controller gains to match the changing dynamics and sensisivity of the actuator. The dotted lines are all damaged profiles that reduce the stability of the airframe whereas the dashed lines are damage profiles that increase the stability of the airframe. Because the controller is robust enough to maintain stable flight the effect of the damage just causes a steady state error in the responses in the most extreme cases by up to a few degrees in pitch. These responses form a baseline deviation from the command by which the adaptive controller responses can be compared.

Figure 5.4.5 shows the unstable responses of the controller designed using the pole placement technique. In the linear approximation case the controller performed well but with the nonlinear dynamics and actuator dynamics included in the simulation there is instability due to the positive alpha gain and the large theta gain terms. The choice and location of poles therefore requires some trial and error testing or expert knowledge to remain stable under all operating conditions.

In the next figures, Fig. 5.4.6 and Fig. 5.4.7, we apply the CCM Single and Double loop controllers to all damaged configurations to see how well they adapt to the changing dynamics. The single loop controller improves the tracking error compared to the baseline root locus performance but exhibits a larger overshoot especially for the baseline case for a contraction rate of 1.5.

Figures 5.4.8 and 5.4.9 show the effect of reducing the contraction rate to 1.2 from 1.5 in the previous graphs. Although the overshoot is reduced for both the double and single loop CCM controllers there are larger tracking errors for the damaged cases 5 and 6, the

ramp offsets. It highlights an issue with taking a linear approximation for a nonlinear change in the dynamics that the uncertainty in the approximation and the uncertainty due to the controller design can cause larger deviations from the the optimal controller design. The double loop CCM also seems to be more susceptible to this nonlinear uncertainty than the single loop CCM controller as more of the damaged profiles have a tracking error furhter from the command. It also highlights the need to choose a contraction rate that balances performance for tracking error reduction that also does not overshoot or oscillate severely.

The RCCM controllers for all damaged conditions are shown in Figures 5.4.10 and 5.4.11. These controllers being based on optimal quadratic regulator design are much smoother in their transient responses. The steady state errors are larger in the unity weighting controller. The error appears to be due to the linear approximation errors in the dynamics being different to the physical nonlinear dynamics of the system because the baseline controller has more precise tracking. The priority weighting of 4 for the theta feedback appears to be optimal for this system because the rise time and tracking error is small and there is no significant overshoot. It poses the question of whether an exponential controller or an asymptotic controller can or needs to be traded off in terms of performance if the asymptotic controller can be certified as both robust and optimal in performance it appears to be the better choice for flying qualities and reliability.

The next set of figures explore the concept of mixing controllers that were previously defined. Figure 5.4.12 shows a hybrid controller that combines the linear sum of a double loop CCM controller and a unity weighting RCCM controller with a mixing parameter $a = 0.5$. The mixed responses show both characteristics of the RCCM, the smoother transient responses and the smaller overshoot of the CCM controller while having a faster rise time for the RCCM unity controller. The combination of the two controllers therefore provides some benefit of each while also inhibiting to some extent the shortcomings. Figure 5.4.13 shows the baseline responses for a set of hybrid controllers varying between 0 and 1 for the LMI version of the controller. The blue and red solid lines represent the

extremes of the mixing at $a = 1$ when the controller is a unity RCCM controller and $a = 0$ when the controller is a double loop CCM controller. The mid-point in the mixing is highlighted as a magenta line that shows a non-linear transition between responses. Whereas if we look at 5.4.14, where the linear sum of controllers is shown the mid-point is in the middle of the two extremes and so provides a greater guarantee on the performance of the mixed controller. This trend can also be observed in the most extreme case in the damaged case 6 for the LMI mixed controller, Figure 5.4.15 and the linear sum mixed controller 5.4.16, where the LMI controller responses lie outside of the two mixed extremes offering no guarantees on the performance whereas the linear sum controller is bound by the two extremes with a linear transition between the two and hence provides a guarantee at least of the performance based on the performances of the two controllers that are mixed.

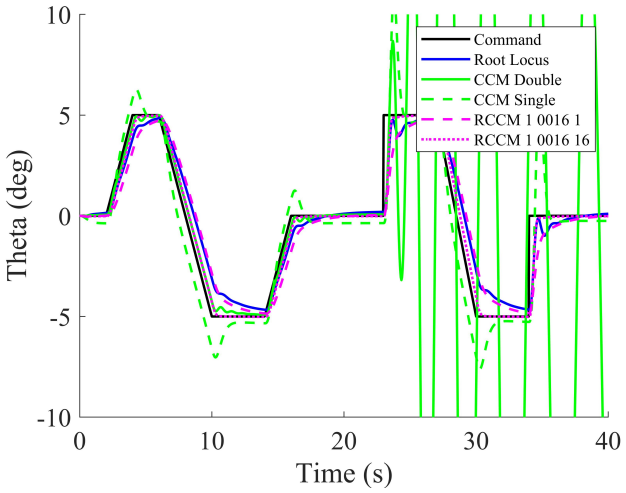


Figure 5.4.1 – Baseline Controllers with CCM contraction rate of 2.1 saturating actuator

5.5 Discussion

We tested 4 new approaches using the CCM technique and the RCCM technique with different parameterisations. We found that the CCM single loop controller provides better performance in terms of rise time than a myopic RCCM controller or an off-nominal

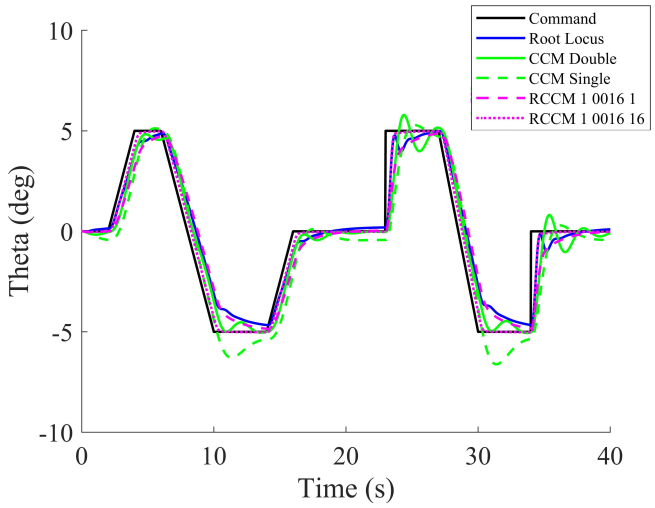


Figure 5.4.2 – Baseline Controllers with CCM contraction rate of 1.5

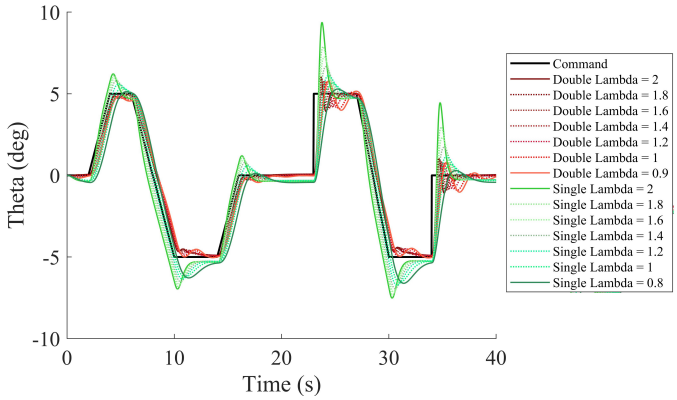


Figure 5.4.3 – Baseline CCM Double and Single increasing lambda

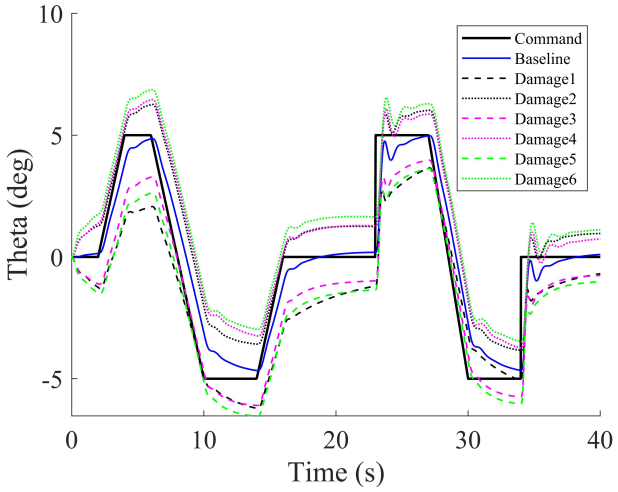


Figure 5.4.4 – Root Locus Baseline Controller for All Damaged Profiles

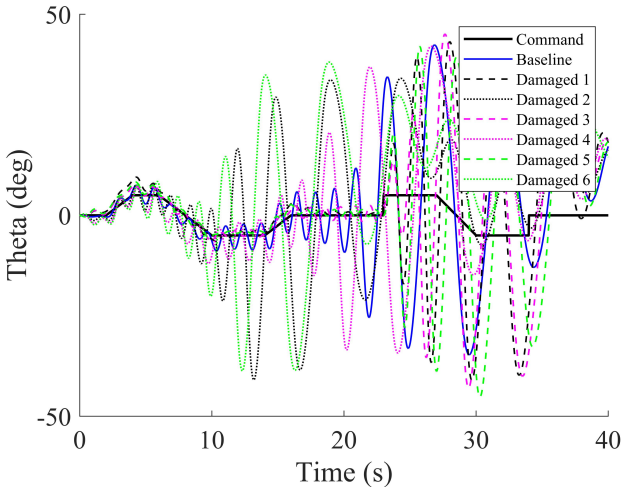


Figure 5.4.5 – Pole Placement Controller for All Damaged Profiles

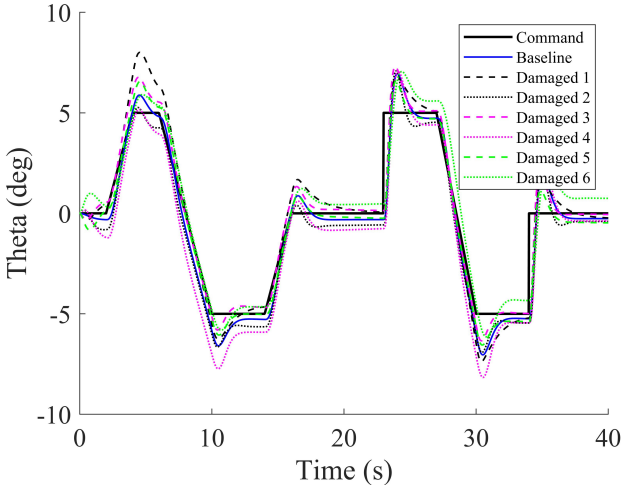


Figure 5.4.6 – CCM Single Loop with $\lambda = 1.5$ Controller for All Damaged Profiles

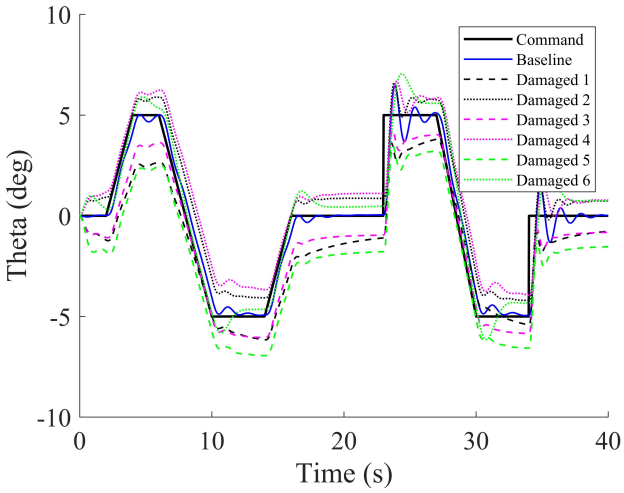


Figure 5.4.7 – CCM Double Loop with $\lambda = 1.5$ Controller for All Damaged Profiles

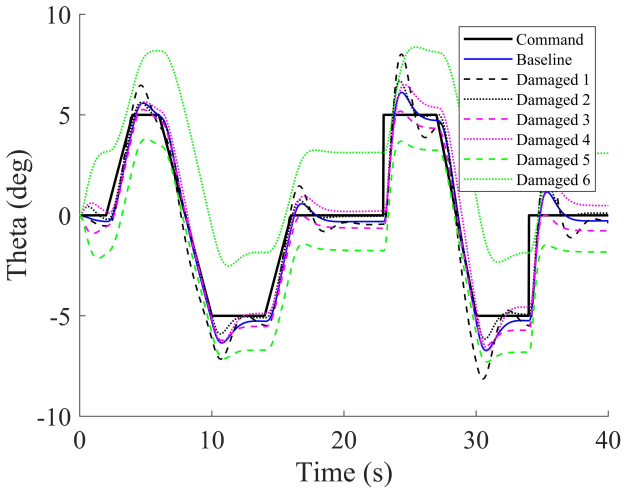


Figure 5.4.8 – CCM Single Loop Controller with $\lambda = 1.2$ for All Damaged Profiles

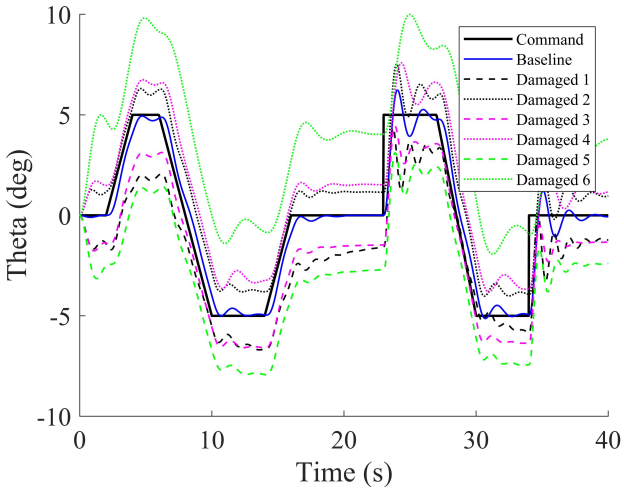


Figure 5.4.9 – CCM Double Loop Controller with $\lambda = 1.2$ for All Damaged Profiles

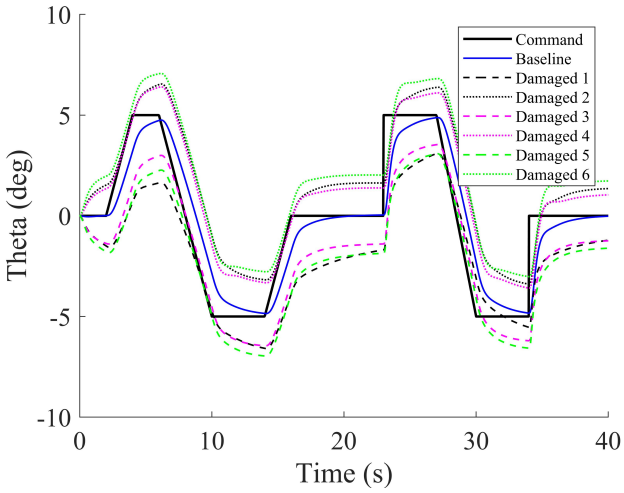


Figure 5.4.10 – RCCM [1 0016 1] Controller for All Damaged Profiles

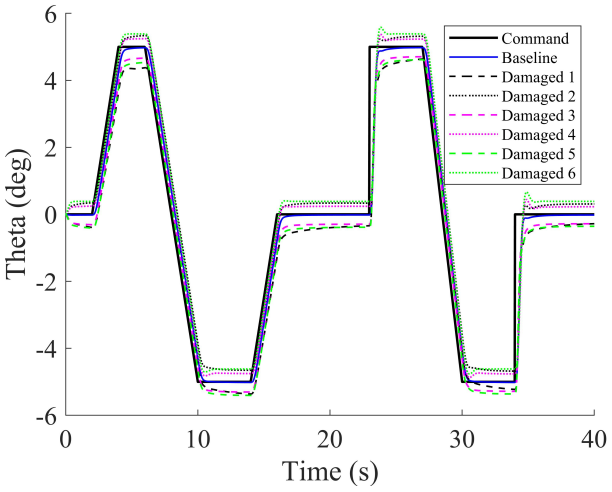


Figure 5.4.11 – RCCM [1 0016 16] Controller for All Damaged Profiles

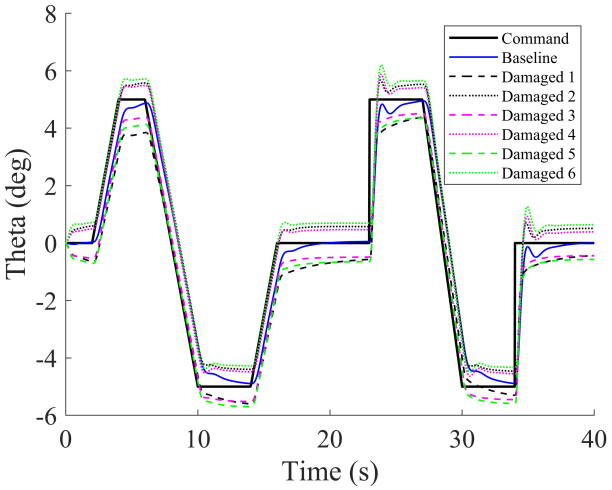


Figure 5.4.12 – Hybrid Linear Sum for All Damaged Profiles

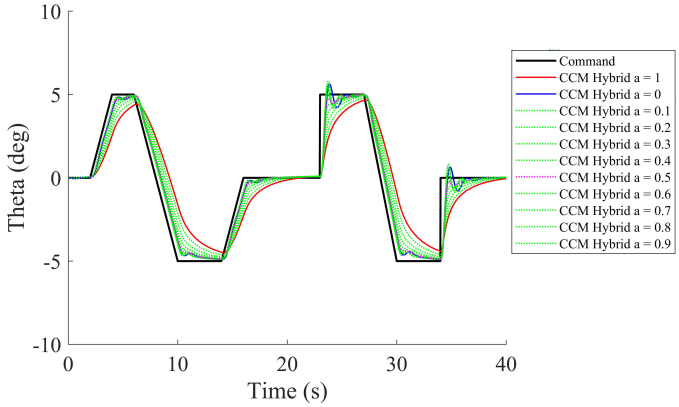


Figure 5.4.13 – Baseline Hybrid CCM Double Loop and RCCM with mixing parameter

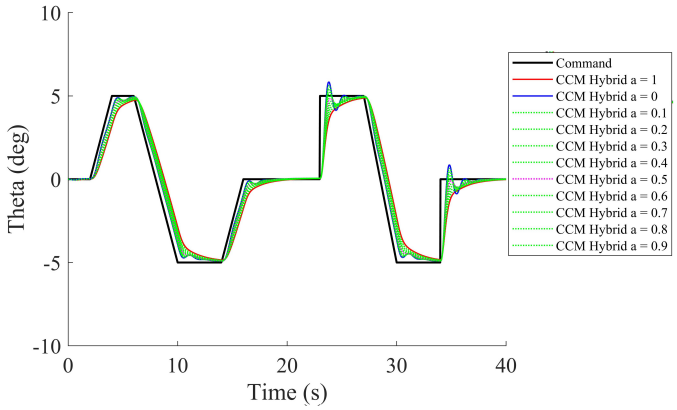


Figure 5.4.14 – Baseline Hybrid Linear Sum CCM Double Loop and RCCM with mixing parameter

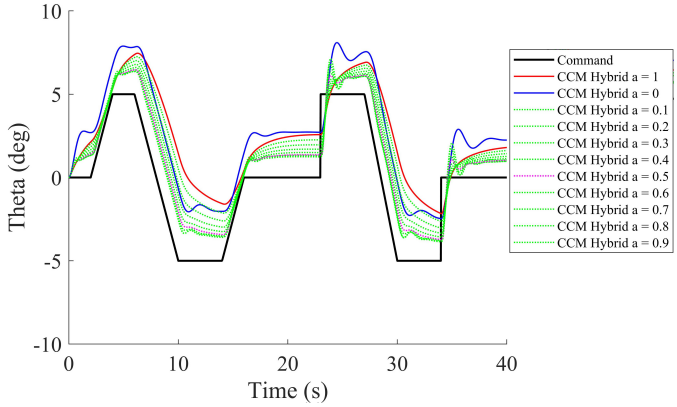


Figure 5.4.15 – Damaged Case 6 Hybrid CCM Double Loop and RCCM with mixing parameter

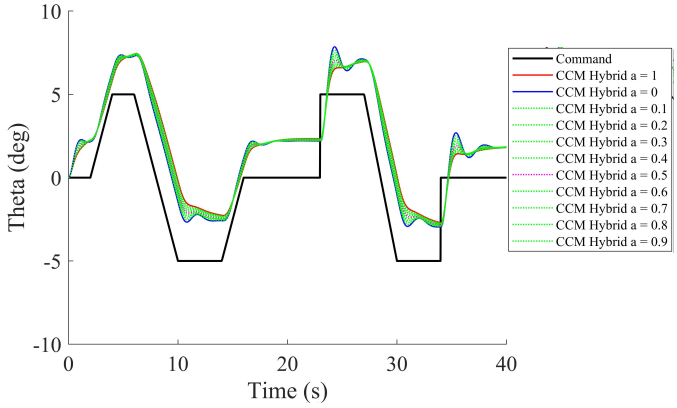


Figure 5.4.16 – Damaged Case 6 Hybrid Linear Sum CCM Double Loop and RCCM with mixing parameter

baseline controller but it lacks robustness in the definition of the feedback gains and suffers from too high a natural frequency and overshoot with multi-modal behaviour. This behaviour is somewhat alleviated by the double loop CCM controller that more explicitly defines the contraction rate for each state feedback and generates negative feedback for all states. This in turn means that some tracking performance is improved via the CCM double loop but some multi-modal behaviour remains. This is most noticeable when the contraction rate is reduced and tested over mutliple damaged test configurations where larger tracking errors are observed compared to the other controllers. The RCCM technique is by nature an optimal control technique with a bound on the L_2 gain of the states. These controllers are much smoother in their transient responses and with the correct weighting can match or better the rise times of the CCM controllers and are hence optimal. A further benefit of the RCCM controllers is that they provide a tuning framework in which a performance for the aircraft can be improved in more degrees of freedom than the CCM technique. By tuning the correct weights, the controller can improve natural frequency or rise time and damping ratio in reference to a 2^{nd} order characteristic. We also compared the root locus design approach and found a controller gain limit that defines a norm bound for the optimisation of the controller metric. The root locus technique provides a good benchmark for controller design but can be improved by the use of an RCCM controller with optimal weighting as the root locus technique can only consider SISO systems, one loop at a time, the inner loop could be further optimised with more iterations.

5.5.1 Advantages and limitations of the presented methods

The CCM single loop controller demonstrates better rise time performance compared to myopic LQR and off-nominal baseline controllers. This suggests that CCM-based controllers can enhance the responsiveness of aircraft control systems, leading to improved maneuverability and agility. However, it is important to note that the CCM single loop controller lacks robustness in defining feedback gains and exhibits high natural frequency

and overshoot, accompanied by multi-modal behavior. This limitation highlights the need for further refinement in the design and tuning of CCM controllers to ensure stability and optimal performance across different flight scenarios.

- **CCM Single Loop Controller:**

  Advantages:

  - Improved rise time performance compared to a myopic RCCM or off-nominal baseline controller.

  Limitations:

  - Lack of robustness in feedback gain definition.
  - High natural frequency and overshoot.
  - Multi-modal behavior.
  - Reduced tracking performance under damaged test configurations.

- **CCM Double Loop Controller:**

  Advantages:

  - Alleviation of multi-modal behavior compared to the single loop controller.
  - Improved tracking performance.

  Limitations:

  - Some residual multi-modal behavior persists, especially with reduced contraction rate.
  - Larger tracking errors observed compared to other controllers under certain conditions.

- **RCCM Technique (Optimal Control):**

  Advantages:

  - Smoother transient responses compared to CCM controllers.
  - Ability to match or exceed the rise time of CCM controllers with appropriate weighting.
  - Tuning framework for improving multiple performance characteristics, including natural frequency, rise time, and damping ratio based on second-order characteristics.
  - Provides more degrees of freedom for performance improvement compared to CCM technique.
  - Certifiability of stability for a linear sum of controllers if both contributing controllers are stable, opening new avenues for controller synthesis.

- **Root Locus Design Approach:**

  Advantages:

  - Serves as a benchmark for comparison.

  Limitations:

  - One-loop-at-a-time consideration limits optimization capability.
  - Can be outperformed by RCCM controllers with optimal weighting.

5.5.2 Implications for aircraft control design

Overall, the study shows that control learning for aircraft remains an iterative process that involves collecting performance data, re-tuning parameters, and evaluating improvements in flight characteristics. Certification of the control policy and learned model is vital to ensure safety and optimality. The key step in the process is certifying a policy as safe and that also involves the certification of the learned model as well as the control policy. A badly learned model or nonlinearities not present in the linear model will mean the controller is non-optimal and can only be certified if robust to all nonlinearities.

5.6 Conclusions and Future Work

A novel 2-loop solution using the CCM technique has been proposed which allows for a more explicit assignment of the eigenvalues for all states of the system and has shown improvements to the single loop design in robustness and reduction in cross coupling of system modes. We showed the connection of the LQR technique in the LTI case to an existing RCCM technique that adopts an L_2 gain bound on performance parameters for the system and defined a normalisation and prioritisation matrix approach to optimise performance. We also showed that a trade-off can be made between controller techniques both as an LMI optimisation and a linear combination of controllers to trade off robust control with performance based controllers. The design of the system was further explored as a SISO using the root locus technique where the inner loop 2nd order model was iterated to find the optimal outer loop gain solution as a full state proportional feedback controller.

In this study we compared six novel approaches to controller design developed in previous studies. We found that the CCM single loop controller surpasses a myopic RCCM and off-nominal baseline controller in terms of rise time performance. However, it exhibits limitations, including lack of robustness in defining feedback gains and a higher natural frequency with multi-modal behavior. Some of these limitations are mitigated by the double loop CCM controller, which more precisely defines the contraction rate for each state feedback, resulting in enhanced tracking performance. Yet, multi-modal behavior persists, especially under multiple damaged test configurations. The RCCM technique, an optimal control approach with a bound on the L_2 gain of the states, offers smoother transient responses. With proper weighting, RCCM controllers can meet or exceed the rise times of CCM controllers and provide a framework for performance tuning that is more flexible than the CCM technique.

5.6.1 Future work

The extensions and future developments of this baseline approach are:

1. Since the performance of the CCM controllers exhibits multi-modal behaviour the structure of the feedback could be improved to isolate the angle or attack from the inner loop to prevent coupling with the theta response.

2. In the parameterised RCCM formulation updating the prioritisation matrix with feedback data in a reinforcement style loop could be implemented to adapt the controller to in flight data gathering.

3. The state feedback can also be augmented with integral states or derivatives to be similar in design to standard PID optimisation techniques.

4. The use of a hybrid linear sum of two controllers could be further tested to show the trade-off in performance of each control policy based on mission requirements or safety margins.

5. It is also possible to include uncertaity in the model approximation to give a robustness to the solution to find a minimum decay rate for all given uncertainty parameters.

6. constraints on the condition number of the metric relative to disturbance bounds could be explored to trade-off controller saturation points with rise time performance.

7. the architechture and techniques may be extended for nonlinear dynamics for which CCMs are most attractive.

Acknowledgements

This work was carried out under an Industrial CASE studentship jointly funded by the EPSRC and BAE Systems.

References

[1] S. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan. *Linear matrix inequalities in system and control theory*. SIAM, 1994.

[2] M. Chilali, P. Gahinet, and P. Apkarian. Robust pole placement in lmi regions. *IEEE transactions on Automatic Control*, 44(12):2257–2270, 1999.

[3] T. Cimen. Survey of state-dependent riccati equation in nonlinear optimal feedback control synthesis. *Journal of Guidance, Control, and Dynamics*, 35(4):1025–1047, 2012.

[4] G. P. Dos Santos, A. Kossoski, J. M. Balthazar, and A. M. Tusset. Sdre and lqr controls comparison applied in high-performance aircraft in a longitudinal flight. *International Journal of Robotics and Control Systems*, 1(2):131–144, 2021.

[5] E. N. Johnson and S. K. Kannan. Adaptive flight control for an autonomous unmanned helicopter. *AIAA Guidance, Navigation, and Control Conference and Exhibit*, (August), 2002.

[6] H.-S. Ju and C.-C. Tsai. Longitudinal axis flight control law design by adaptive backstepping. *IEEE Transactions on Aerospace and Electronic Systems*, 43(1):311–329, 2007.

[7] K. Leung and I. R. Manchester. Nonlinear stabilization via control contraction metrics: A pseudospectral approach for computing geodesics. *Proc. Am. Control Conf.*, pages 1284–1289, 2017.

[8] D. Malyuta, T. P. Reynolds, M. Szmuk, T. Lew, R. Bonalli, M. Pavone, and B. Acikmese. Convex Optimization for Trajectory Generation. 2021.

[9] I. R. Manchester and J.-J. E. Slotine. Control contraction metrics: Convex and intrinsic criteria for nonlinear feedback design. *IEEE Transactions on Automatic Control*, 62(6):3046–3053, 2017.

[10] I. R. Manchester and J. J. E. Slotine. Robust control contraction metrics: A convex approach to nonlinear state-feedback control. *IEEE Control Systems Letters*, 2(3):333–338, 2018.

[11] L. T. Nguyen. *Simulator study of stall/post-stall characteristics of a fighter airplane with relaxed longitudinal static stability*, volume 12854. National Aeronautics and Space Administration, 1979.

[12] L. Sonneveldt, E. R. van Oort, Q. Chu, and J. Mulder. Nonlinear adaptive flight control law design and handling qualities evaluation. In *Proceedings of the 48h IEEE Conference on Decision and Control (CDC) held jointly with 2009 28th Chinese Control Conference*, pages 7333–7338. IEEE, 2009.

[13] H. Tsukamoto and S. J. Chung. Robust controller design for stochastic nonlinear systems via convex optimization. *IEEE Transactions on Automatic Control*, pages 1–1, 2020.

[14] F. Yang. Fast model predictive control using online optimization. *IEEE Transactions on Control Systems Technology*, 18(2):267–278, 2010.

Chapter 6

Discussion on Learning via CCMs

In this chapter the work developed in Chapters 3 to 5 will be discussed for the application of the online learning algorithm to the full flight envelope. This includes an exploration of how uncertainties are addressed within the optimization framework of the controller, particularly through an enhanced LMI formulation. A key aspect of this discussion is the focus on the challenges encountered in the transition from a scheduled controller to a fully nonlinear one, highlighting the complexity involved in developing more sophisticated control mechanisms. Consequently, this chapter provides a comprehensive discussion on aspects of the research that, while not enclosed within the thesis's core scope, significantly contribute to the practical deployment of the learning pipeline. It is important to note that these discussions are distinct from the future work highlighted in the subsequent chapter, as they pertain to considerations beyond the immediate focus of the current study.

6.1 Uncertainty Handling for Polytopic Uncertainty

In Chapter 4, we presented the linear approximations for a set of damaged aircraft profiles, which yielded a range of possible values for the A and B matrices. For the linear Eqn. 6.1 to 6.3, this range forms a convex combination of matrices for both A and B, resulting in a bilinear optimization problem.

$$\dot{x} = Ax + Bu \tag{6.1}$$

$$\text{where} \quad A = \sum_i \beta_i A_i, \quad B = \sum_j \eta_j B_j \quad \text{and} \quad \beta_i, \eta_j > 0, \forall i, j \tag{6.2}$$

$$\text{with} \quad \sum_i \beta_i = 1, \quad \sum_j \eta_j = 1 \tag{6.3}$$

However, this can be solved as a set of LMIs to find a convex set of control gains $K_{i,j}$, which stabilise the system as shown in Eqn. 6.4.

$$\min_{W \in \mathbb{R}^{d \times d}, W > 0} \quad \text{trace}(W) \tag{6.4}$$

$$s.t. \quad -(A_i W + W A_i^T + B_j B_j^T + 2\lambda W) \geq 0, \quad \forall A_i \in \mathcal{A}, B_j \in \mathcal{B} \tag{6.5}$$

Here, A_i and B_j are the vertex matrices of the polytopes that represent the possible values of A and B. The control gains can then be computed as $K_{i,j} = \frac{1}{2} B_j^T W_i^{-1}$, forming an $n \times m \times d$ array. The dimensions n and m correspond to the number of vertices in the polytopes for A and B, respectively, while d refers to the dimensionality of the system state. The specific $K_{i,j}$ to be used depends on the region of A and B selected. For a worst-case bound, an extremum is chosen and for a robust solution an average of the polytopic matrices can be used.

6.2 Nonlinear control synthesis

The mathematics involved in the full nonlinear controller optimisation and synthesis is non-trival. A full description is given for the pseudospectral collocation approach adopted by [2],[5] and a nonlinear convex optimisation approaches discussed in [3]. A neural

network convex approach is discussed in [7]. A nonlinear controller provides a feedback gain of the form in Eqn. 6.6, where the function can be any nonlinear function of the state x and is commonly represented as a polynomial.

$$K(x) = \frac{1}{2}\rho(x)B(x)^T M(X) \tag{6.6}$$

The state dependency however in this formulation can also be scheduled as before. The function approximation chosen is also important to represent the shape of the controller in reference to state parameter variations. For example it was observed that in the case of a sum-of-squares representation of the metric M(x) a second or fourth order polynomial in K(x) means that the policy can suffer from instability when the set points are out of a nominal range due to the exponential nature of the function. A second point about nonlinear controllers is that of blending regions of the state-space together if batches of data are used to learn a control policy for a region of the flight envelope then transitions between these control policies becomes difficult because there are no constraints on function boundaries. These problems have been addressed in other studies such as [6] but is an active area of research. An important limitation we found with polynomial approximating functions like SoS is that the scaleability of the functions is limited by the exponential increase in combinations of monomials. Similarly the degree of the polynomials is typically limited to degree 4 for numerically stable solutions. A study by Morelli [4] found that subsonic wind tunnel data for an F-16 could be approximated by multi-variate functions of up to 4^{th} order in the monomials of the states. This shows some promise for the application of low order models at least for a large proportion of the flight envelope. The regions within the flight envelope that pose the most challenge are where discontinuities such as high angle separations occur or transonic flows where shock wave drag becomes problematic. Another issue with these regions within the flight envelope is the stochastic nature of the onset of flow changes. To fully accomodate all adpative control scenarios

a multi-sensing, tracking, and policy selection is most likely to provide a robust solution for aerospace applications.

6.3 Full Flight Envelope Scheduling

The static metric can be learned at any desired operating point in the flight envelope by using the relevant training data and allocation of the metric to a state dependent function for each operating set point, 6.3.1. We propose the technique adopted in the recent work by [1], in which a linear quadratic integral policy is scheduled via a Continuous Multi-Surface Gain (CMSG) schedule, where the CCM or RCCM gains will be interpolated linearly between each scheduling point to form a hypersurface of gains. This approach means that a lookup table is incorporated into the the controller that depends upon the scheduled parameters, altitude and Mach number. The multi-surface approach comes from further scheduling the gains on angle of attack for greater fidelity. The proposed controller block diagram is shown in Fig. 6.3.2. The numder of scheduling parameters depends upon the sensitivity of the dynamics to the parameter in question but could also include actuator surfaces like the leading edge flaps. In our approach because there are also additional tuning parameters available it is also possible to create surfaces for a robustness tradeoff that could be used as a mission critical flight mode dependent upon flight phase requirements. Due to the rise in computing resources available today it is thought that a data rich scheduling policy is more feasible for modern aircraft. The benefits of the hybrid linear sum controller is that the gain matrix can be directly scheduled given the contraints on the feedback structure to certify stability.

6.4 Controller Augmentation

The controller CCM framework can be improved to include integral errors, feedforward terms or more complex compensators to correct for steady state errors phase lag or delays. This means that the system state space representation must be augmented to include the

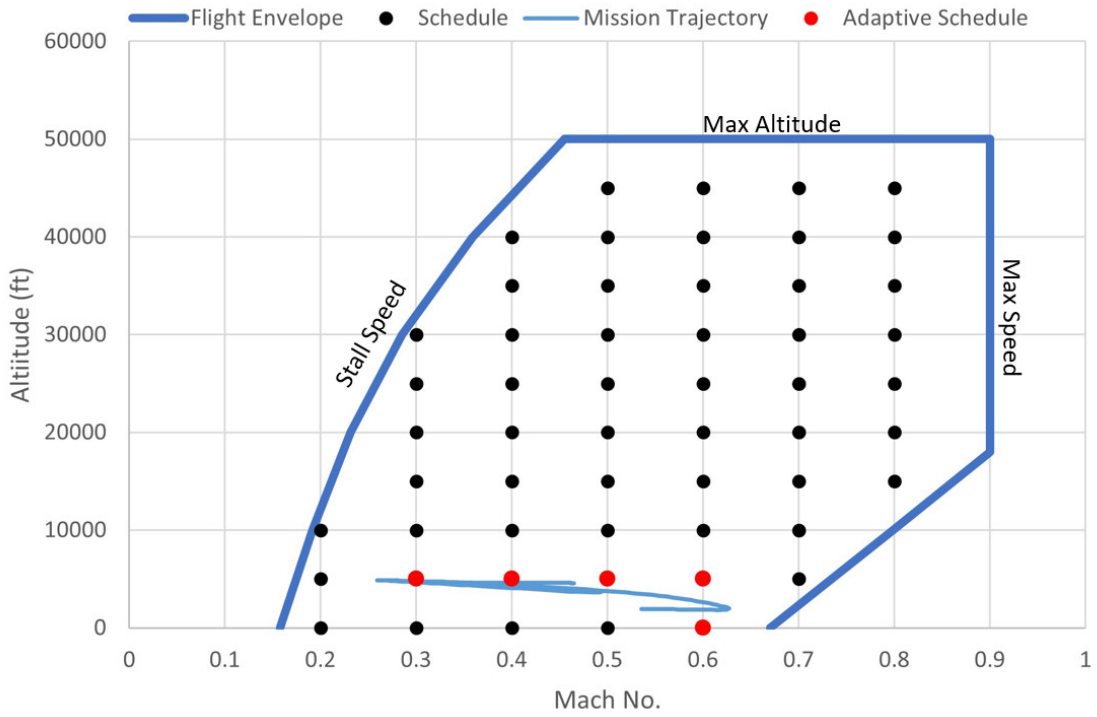


Figure 6.3.1 – Example of an F-16 flight envelope and controller scheduling points. Red points show updated scheduling parameters learned from a flight trajectory.

additional states for consideration. Here we propose an additional integral theta tracking error term as an extension to the basic controller framework to show how this affects the A and B matrix representations. An addititional state $\int \delta_\theta$ and gain $K_{\int \delta_\theta}$ include in Figure 6.4.1 using the same underlying framework as before.

To optimise the new gain term, a new A and B matrix for the attitude hold controller, as shown in Eqn. 6.7, must be used to derive the CCM LMI. Here the additional state $\int \delta_\theta$ has been added and the corresponding integral gain $K_{\int \delta_\theta}$ has also been factored in. This augmentation allows enhanced system performance, stability and robustness in the presence of disturbances and uncertainties.

The updated Figure 6.4.1 shows the expanded framework and how the integral error and its corresponding gain factor can be integrated into the controller CCM framework. Augmenting the controller with an integral error term will lead to a more versatile and adaptive control system, capable of dealing with steady-state errors, phase lag, and delays

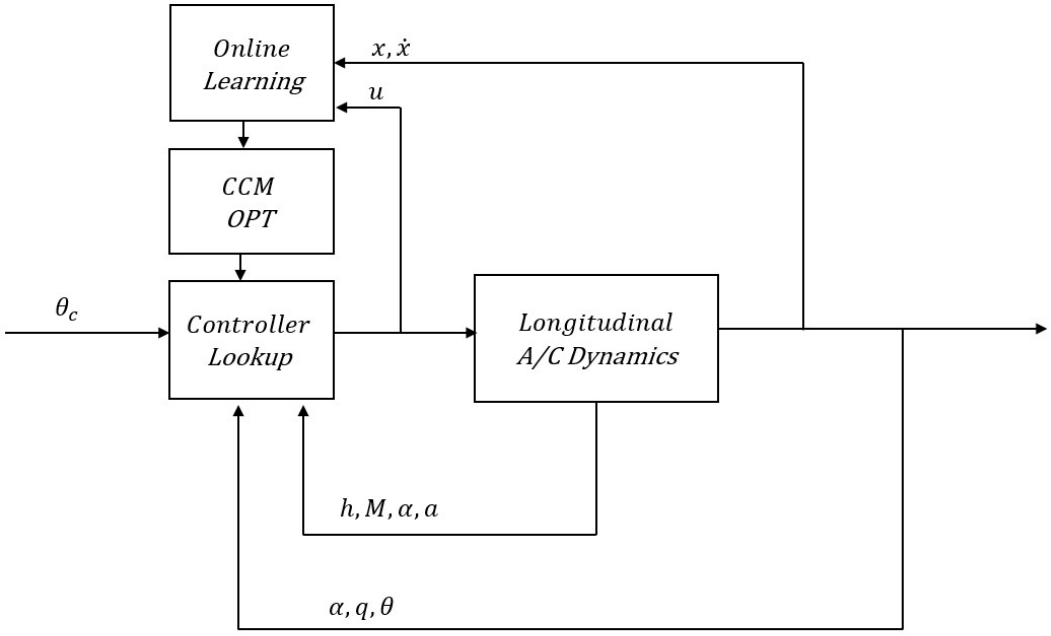


Figure 6.3.2 – Longitudinal Controller CCM Scheduling Example

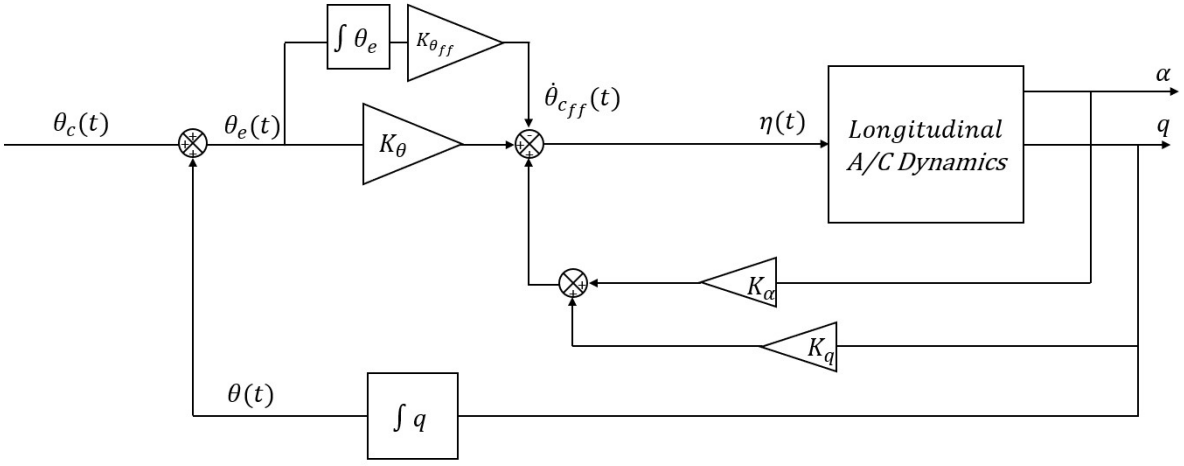


Figure 6.4.1 – Theta integral error controller augmentation

more efficiently.

$$\begin{bmatrix} \dot{\delta}_\alpha \\ \dot{\delta}_q \\ \dot{\delta}_\theta \\ \int \delta_\theta \end{bmatrix} = \begin{bmatrix} Z_\alpha & Z_q & 0 & 0 \\ M_\alpha & M_q & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} \delta_\alpha \\ \delta_q \\ \delta_\theta \\ \int \delta_\theta \end{bmatrix} + \begin{bmatrix} B_\alpha \\ B_q \\ 0 \\ 0 \end{bmatrix} \begin{bmatrix} K_\alpha & K_q & K_\theta & K_{\int \delta_\theta} \end{bmatrix} \begin{bmatrix} \delta_\alpha \\ \delta_q \\ \delta_\theta \\ \int \delta_\theta \end{bmatrix} \tag{6.7}$$

References

[1] M. U. Hassan and A. Ali. Robust aircraft performance using hypersurface gain scheduling lqi controller for supermaneuverable aircraft. *Pakistan Journal of Engineering and Technology*, 5(2):17–22, 2022.

[2] K. Leung and I. R. Manchester. Nonlinear stabilization via control contraction metrics: A pseudospectral approach for computing geodesics. *Proc. Am. Control Conf.*, pages 1284–1289, 2017.

[3] I. R. Manchester and J. J. E. Slotine. Robust control contraction metrics: A convex approach to nonlinear state-feedback control. *IEEE Control Systems Letters*, 2(3):333–338, 2018.

[4] E. A. Morelli. Global nonlinear parametric modeling with application to f-16 aerodynamics. In *Proceedings of the American Control Conference*, pages 1–6, Pennsylvania, 1998.

[5] S. Singh, S. M. Richards, V. Sindhwani, J.-J. E. Slotine, and M. Pavone. Learning stabilizable nonlinear dynamics with contraction-based regularization. *The International Journal of Robotics Research*, 2020.

[6] L. Sonneveldt, E. R. Van Oort, Q. P. Chu, C. C. De Visser, J. A. Mulder, and J. H. Breeman. Lyapunov-based fault tolerant flight control designs for a modern fighter aircraft model. *AIAA Guidance, Navigation, and Control Conference and Exhibit*, (August):1–23, 2009.

[7] H. Tsukamoto and S. Chung. Neural contraction metrics for robust estimation and control: A convex optimization approach. *IEEE Control Systems Letters*, 5(1):211–216, 2021.

Chapter 7

Conclusions and Future Works

This thesis successfully provides an online learning architecture for a flight control system for a fixed wing aircraft based on control contraction metrics and convex optimisation solvers. An in depth analysis into the control synthesis technique for stability certification has been made and the gap in knowledge around linear time invariant flight performance characteristics bridged to allow for integration and application to real world systems. The conclusions to the original objecives in summary are:

- **O1 - A Review of Safe Learning Techniques** We reviewed the state of the art in safe learning of nonlinear systems in chapter 2 with an extensive survey of the state-of-the-art techniques. The survey highlighted the most applicable learning frameworks for high-dimensional and safety critical systems and we made recommendations based on a set of key performance criteria. Based on our survey of approaches we identified two of the most promising techniques, the CCM technique and a hybrid MPC technique with residual dynamics learning.

- **O2 - Benchmarking of safe learning algorithms** A benchmarking exercise of safe learning techniques was carried out for the most suitable techniques for online learning. Safety certificate optimisation, function approximations and control synthesis techniques were compared in up to 10 dimensions.

- **O3 - Development of key performance metrics** Key metrics were developed as an

heuristic exercise in the analysis of learning proposals for six attributes, scaleability, response time, learning time, conservatism, reliability and explainabiliy. The metrics were used to rank techniques with a weighted score to identifiy the most promising avenues for research.

- **O4 - Develop a realistic aircraft testbed simulator** A 3-DOF testbed simulation for a nonlinear F-16 aircraft with lookup tables for aerodynamic coefficients modifications based on damage profiles was developed where the kinematic equations were constructed as a block diagram instead of an embedded C program. Embedded convex optimisation and batch learning components with a memory store for the controllers enables online learning. The simulator provides an excellent tool for experimentation and validation of learning techniques for future research.

- **O5 - Develop novel algorithms to learn aircraft dynamics in the context of design specifications** Two novel adaptations to the current CCM techniques using convex optimisation were developed to explicitly define the eigenvalues of the closed loop system. A 2-loop optimisation improves robustness in the standard technique and a hybrid approach was certified as stable under certain conditions whereby a trade-off between performance and robustness can be adopted.

- **O6 - Develop an algorithm to generate controllers in relation to flying qualities specifications** An algorithm that can learn improvements to the system dynamics in terms of damping ratio and natural frequency was proposed via a parameterisation of a robust CCM controller to provide a mechanism for reinforcement learning algorithm.

- **O7- Show an improvement in aircraft performance using the new approach** Compared to a traditional root locus baseline controller a set of control policies that adapt to deviations in the baseline dynamics increased the aircrafts performance to meet level 1 flying qualities. It was also found that the quality of the linear approximation had a significant impact on the success of the control policy.

- **O8 - Demonstrate the use of an adaptive controller subject to faults or deviations from baseline** Six damaged aircraft profiles were tested as deviations from the baseline to highlight the challenge in learning 3 different nonlinear functions. The updated controller gains adapted to the change in dynamics to maintain Level 1 flying characteristics and the robustness of each design was compared based on the ability of the control policy to adapt to each of the six damage scenarios.

7.1 Conclusion on original hypothesis

The main hypothesis as a result of the benchmarking analysis was that Control Contraction Metrics offer a fast and more expressive framework for stability certification for adaptive flight controllers for reasonable online learning timeframes compared to NMPC or LQR techniques. To answer this question a comparison of learning times was conducted in chapter 1. Learning time frames for up to 8 dimensions in 4 techniques for online learning and safety certificate generation was carried out with offline methods discounted for analysis based on prior studies. We showed that full nonlinear control synthesis using a Pseudospectral technique was slower than LQR calculations or RCCM calcultions in the case of nonlinear functions by an order of magnitude but acceptable for online learning time frames and significantly faster than NMPC. In terms of expressiveness a convex function of nonlinear monomials has been demonstrated to learn the short period mode dynamics. The method of learning is only limited by the choice of monomials selected by the designer and can be based on expert knowledge of the physics of the control system. We demonstrated a 40s trajectory for the training examples of a linear approximation and showed that the quality of the model is highly dependent upon the type of damage and nonlinearities present in the dynamics. The standard CCM optimisation is not well structured to allow for the explicit design of the poles of the system and we proposed a novel variation that alleviates these limitations but still observed multi-modal transient responses in comparison to robust or optimal control techniques such as an LQR or RCCM

controller that adopt a quadratic cost function. Moreover we showed that the RCCM technique architecture provides more degrees of freedom in designing the controller to meet flying qualities constraints and a parameterisation of such a controller could be used as an iterative learning approach in the style of a reinforcment learning algorithm.

7.2 Future Works

- The implementation of a reinforcement algorithm based on the parametrisation of the RCCM and Hybrid controllers will be explored whereby performance data from both the linear approximate models and inflight trajectory data will be usde to inform the direction of adaptation in the control policy.

- Uncertainty estimates for the system model will be incorporated into the LMI optimsations to compare robust control limitations on learning performance.

- Learning of the full flight envelope subject to faults will be explored to understand the limitations of the regression model and synthesis of the controller gains. This will inform the number of set points for a gain scheduling approach subject to aircraft performance.

- More advanced control architectures will be explored to improve the tracking performance of the controllers to reduce steady state errors and modal interactions.

- This intial study focused on the control certification but an important part of the learning pipeline is the certification of the learned model. A study into the validation of trajectory data and model fitting based on the selection of monomials is of interest to understand how confidence can be assured in the updated control policy.

Complete Bibliography

[1] M. Abdolhosseini, Y. M. Zhang, and C. A. Rabbath. An efficient model predictive control scheme for an unmanned quadrotor helicopter. *J. Intell. Robot. Syst. Theory Appl.*, 70(1-4):27–38, 2013.

[2] A. A. Abdullah, P. A. Ioannou, and G. Samaras. Control design for f-16 longitudinal motion. *IFAC Proceedings Volumes*, 37(6):529–534, 2004.

[3] R. J. Adams, J. M. Buffington, and S. S. Banda. Design of nonlinear control laws for high-angle-of-attack flight. *Journal of Guidance, Control, and Dynamics*, 17(4):737–746, 1994.

[4] A. A. Ahmadi, A. Chaudhry, V. Sindhwani, and S. Tu. Safely learning dynamical systems from short trajectories. pages 498–509, 2021.

[5] A. A. Ahmadi and A. Majumdar. Some applications of polynomial optimization in operations research and real-time decision making. *Optim. Lett.*, 10(4):709–729, 2016.

[6] A. A. Ahmadi and A. Majumdar. DSOS and SDSOS optimization: More tractable alternatives to sum of squares and semidefinite optimization. *arXiv*, 3(2):193–230, 2017.

[7] A. A. Ahmadi and A. Majumdar. DSOS and SDSOS Optimization: More Tractable Alternatives to Sum of Squares and Semidefinite Optimization. *SIAM*, 3(2):193–230, 2019.

[8] Air Force Times. F35 high angle of attack, 2019. [Online; accessed April 16, 2021].

[9] A. K. Akametalu, S. Kaynama, J. F. Fisac, M. N. Zeilinger, J. H. Gillula, and C. J. Tomlin. Reachability-based safe learning with gaussian processes. *Proceedings of*

the IEEE Conference on Decision and Control, 2015-Febru(February):1424–1431, 2014.

[10] O. Albostan and M. Gokasan. High angle of attack manoeuvring control of f-16 aircraft based on nonlinear dynamic inversion and eigenstructure assignment. In *7th European Conference for Aeronautics and Space Sciences (EUCASS)*, pages 3–6, 2017.

[11] M. D. Aliyu. Iterative computational approach to the solution of the Hamilton-Jacobi-Bellman-Isaacs equation in nonlinear optimal control. *Control Theory Technol.*, 16(1):38–48, 2018.

[12] M. Althoff, O. Stursberg, and M. Buss. Reachability analysis of nonlinear systems with uncertain parameters using conservative linearization. *Proceedings of the IEEE Conference on Decision and Control*, pages 4042–4048, 2008.

[13] A. D. Ames, S. Coogan, M. Egerstedt, G. Notomista, K. Sreenath, and P. Tabuada. Control barrier functions: Theory and applications. *2019 18th European Control Conference, ECC 2019*, pages 3420–3431, 2019.

[14] J. A. Andersson, J. Gillis, G. Horn, J. B. Rawlings, and M. Diehl. CasADi: a software framework for nonlinear optimization and optimal control. *Math. Program. Comput.*, 11(1), 2019.

[15] A. M. Annaswamy and A. L. Fradkov. A historical perspective of adaptive control and learning. *Annual Reviews in Control*, 52:18–41, 2021.

[16] M. Aps. Mosek modeling cookbook, 2020.

[17] N. Aronszajn. Theory of reproducing kernels. *Transactions of the American Mathematical Society*, 68(3):337–404, 2009.

[18] A. B. Arrieta, N. Díaz-Rodríguez, J. Del Ser, A. Bennetot, S. Tabik, A. Barbado, S. García, S. Gil-López, D. Molina, R. Benjamins, et al. Explainable artificial

intelligence (xai): Concepts, taxonomies, opportunities and challenges toward responsible ai. *Information fusion*, 58:82–115, 2020.

[19] E. M. Aylward, P. A. Parrilo, and J.-J. E. Slotine. Stability and robustness analysis of nonlinear systems via contraction metrics and sos. *Automatica*, 44(8):2163–2170, 2008.

[20] A. Ayyad, M. Chehadeh, M. I. Awad, and Y. Zweiri. Real-time system identification using deep learning for linear processes with application to unmanned aerial vehicles. *IEEE Access*, 8:122539–122553, 2020.

[21] G. J. Balas. Flight control law design: An industry perspective. *European Journal of Control*, 9(2-3):207–226, 2003.

[22] S. Bansal, R. Calandra, T. Xiao, S. Levine, and C. J. Tomiin. Goal-driven dynamics learning via Bayesian optimization. *2017 IEEE 56th Annu. Conf. Decis. Control. CDC 2017*, 2018-Janua(1545126):5168–5173, 2018.

[23] S. Bansal, M. Chen, S. Herbert, and C. J. Tomlin. Hamilton-jacobi reachability: A brief overview and recent advances. *2017 IEEE 56th Annual Conference on Decision and Control, CDC 2017*, 2018-January:2242–2253, 2018.

[24] A. Bemporad, M. Morari, V. Dua, and E. N. Pistikopoulos. The explicit linear quadratic regulator for constrained systems. *Automatica*, 38(1):3–20, 2002.

[25] F. Berkenkamp, A. Krause, and A. P. Schoellig. Bayesian optimization with safety constraints: safe and automatic parameter tuning in robotics. *arXiv preprint arXiv:1602.04450*, 2016.

[26] F. Berkenkamp, R. Moriconi, A. P. Schoellig, and A. Krause. Safe learning of regions of attraction for uncertain, nonlinear systems with gaussian processes. *2016 IEEE 55th Conference on Decision and Control (CDC)*, pages 4661–4666, 2016.

[27] F. Berkenkamp and A. P. Schoellig. Safe and robust learning control with gaussian processes. *2015 European Control Conference, ECC 2015*, pages 2496–2501, 2015.

[28] F. Berkenkamp, M. Turchetta, A. P. Schoellig, and A. Krause. Safe model-based reinforcement learning with stability guarantees. *Advances in Neural Information Processing Systems*, 2017-Decem(Nips):909–919, 2017.

[29] R. Bobiti and M. Lazar. Automated-sampling-based stability verification and doa estimation for nonlinear systems. *IEEE Transactions on Automatic Control*, 63(11):3659–3674, 2018.

[30] N. M. Boffi, S. Tu, N. Matni, J. J. E. Slotine, and V. Sindhwani. Learning stability certificates from data. *arXiv*, pages 1–30, 2020.

[31] S. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan. *Linear matrix inequalities in system and control theory*, volume 15. SIAM, 1994.

[32] S. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan. *Linear matrix inequalities in system and control theory*. SIAM, 1994.

[33] S. Boyd and L. Vandenberghe. *Convex optimization*. Cambridge University Press, 2004.

[34] M. Branicky and M. Curtiss. Nonlinear and hybrid control via RRTs. *Proc. Intl. Symp. Math. Theory Networks ...*, pages 1–10, 2002.

[35] E. Brochu, V. M. Cora, and N. de Freitas. A tutorial on bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning. *arXiv*, 2010.

[36] S. L. Brunton, J. Nathan Kutz, K. Manohar, A. Y. Aravkin, K. Morgansen, J. Klemisch, N. Goebel, J. Buttrick, J. Poskin, A. Blom-Schieber, T. Hogan, and

D. McDonald. Data-driven aerospace engineering: Reframing the industry with machine learning. *arXiv*, pages 1–35, 2020.

[37] S. Bubeck et al. Convex optimization: Algorithms and complexity. *Foundations and Trends® in Machine Learning*, 8(3-4):231–357, 2015.

[38] L. F. Buzachero, E. Assuncao, M. C. Teixeira, and E. R. Da Silva. New techniques for optimizing the norm of robust controllers of polytopic uncertain linear systems. *Frontiers in advanced control systems*, pages 75–100, 2012.

[39] T. L. Chaffey and I. R. Manchester. Control Contraction Metrics on Finsler Manifolds. *Proc. Am. Control Conf.*, 2018-June:3626–3633, 2018.

[40] Y.-C. Chang, N. Roohi, and S. Gao. Neural lyapunov control. *Advances in neural information processing systems*, 32, 2019.

[41] M. Chen, S. L. Herbert, M. S. Vashishtha, S. Bansal, and C. J. Tomlin. Decomposition of Reachable Sets and Tubes for a Class of Nonlinear Systems. *IEEE Trans. Automat. Contr.*, 63(11):3675–3688, 2018.

[42] S. Chen, M. Fazlyab, M. Morari, G. J. Pappas, and V. M. Preciado. Learning lyapunov functions for piecewise affine systems with neural network controllers. *arXiv*, 2020.

[43] X. Chen, S. Chen, and V. M. Preciado. Safety Verification of Nonlinear Polynomial System via Occupation Measures. (Cdc):1159–1164, 2019.

[44] R. Cheng, G. Orosz, R. M. Murray, and J. W. Burdick. End-to-end safe reinforcement learning through barrier functions for safety-critical continuous control tasks. *Proceedings of the AAAI Conference on Artificial Intelligence*, 33(01):3387–3395, Jul. 2019.

[45] M. Chilali, P. Gahinet, and P. Apkarian. Robust pole placement in lmi regions. *IEEE transactions on Automatic Control*, 44(12):2257–2270, 1999.

[46] Y. Chow, O. Nachum, A. Faust, E. Duenez-Guzman, and M. Ghavamzadeh. Lyapunov-based safe policy optimization for continuous control. *arXiv e-prints*, 2019.

[47] T. Cimen. Survey of state-dependent riccati equation in nonlinear optimal feedback control synthesis. *Journal of Guidance, Control, and Dynamics*, 35(4):1025–1047, 2012.

[48] G. Cimini, D. Bernardini, S. Levijoki, and A. Bemporad. Embedded model predictive control with certified real-time optimization for synchronous motors. *IEEE Trans. Control Syst. Technol.*, pages 1–8, 2020.

[49] A. I. S. T. Committee et al. Roadmap for intelligent systems in aerospace. *Reston: AIAA*, 2016.

[50] M. V. Cook. *Flight dynamics principles: a linear systems approach to aircraft stability and control*. Butterworth-Heinemann, 2012.

[51] W. B. Cotton, R. Hilb, S. Koczo, and D. J. Wing. A vision and roadmap for increasing user autonomy in flight operations in the national airspace. In *16th AIAA Aviation Technology, Integration, and Operations Conference*, page 4212, 2016.

[52] K. Dalamagkidis, K. P. Valavanis, and L. A. Piegl. Nonlinear model predictive control with neural network optimization for autonomous autorotation of small unmanned helicopters. *IEEE Trans. Control Syst. Technol.*, 19(4):818–831, 2011.

[53] J. Darbon and S. Osher. Algorithms for overcoming the curse of dimensionality for certain hamilton–jacobi equations arising in control theory and elsewhere. *Research in Mathematical Sciences*, 3(1), 2016.

[54] M. C. de Oliveira and R. E. Skelton. Stability tests for constrained linear systems. In *Perspectives in robust control*, pages 241–257. Springer, 2007.

[55] S. Dean, S. Tu, N. Matni, and B. Recht. Safely learning to control the constrained linear quadratic regulator. In *2019 American Control Conference (ACC)*, pages 5582–5588, 2019.

[56] G. P. Dos Santos, A. Kossoski, J. M. Balthazar, and A. M. Tusset. Sdre and lqr controls comparison applied in high-performance aircraft in a longitudinal flight. *International Journal of Robotics and Control Systems*, 1(2):131–144, 2021.

[57] C. Edwards, H. Alwi, and C. Tan. Sliding mode methods for fault detection and fault tolerant control with application to aerospace systems. *International Journal of Applied Mathematics and Computer Science*, 22(1):109–124, 2012.

[58] Electrive. Lillium vtol air taxi, 2020. [Online; accessed April 26, 2021].

[59] U. Eren, A. Prach, B. B. Koçer, S. V. Rakovic, E. Kayacan, and B. Açikmese. Model predictive control in aerospace systems: Current state and opportunities. *Journal of Guidance, Control, and Dynamics*, 40(7):1541–1566, 2017.

[60] S. Fedorov and A. Candelieri. Reachability-based safe learning for optimal control problem. pages 1–13, 2018.

[61] C. Finn, S. Levine, and P. Abbeel. Guided cost learning: Deep inverse optimal control via policy optimization. *33rd International Conference on Machine Learning, ICML 2016*, 1:95–107, 2016.

[62] J. Fisac, A. Akametalu, M. Zeilinger, S. Kaynama, J. Gillula, and C. Tomlin. A general safety framework for learning-based control in uncertain robotic systems. *IEEE Transactions on Automatic Control*, 64(7):2737–2752, 2019.

[63] J. F. Fisac, A. K. Akametalu, M. N. Zeilinger, S. Kaynama, J. Gillula, and C. J. Tomlin. A general safety framework for learning-based control in uncertain robotic systems. *IEEE Transactions on Automatic Control*, 64(7):2737–2752, 2019.

[64] M. Fisher, V. Mascardi, K. Y. Rozier, B.-H. Schlingloff, M. Winikoff, and N. Yorke-Smith. Towards a framework for certification of reliable autonomous systems. *Autonomous Agents and Multi-Agent Systems*, 35(1):1–65, 2021.

[65] F. Forni and R. Sepulchre. A differential lyapunov framework for contraction analysis. *IEEE Trans. Automat. Contr.*, 59(3):614–628, 2014.

[66] N. Fulton and A. Platzer. Safe reinforcement learning via formal methods: Toward safe control through proof and learning. *32nd AAAI Conference on Artificial Intelligence, AAAI 2018*, pages 6485–6492, 2018.

[67] M. Ghavamzadeh, S. Mannor, J. Pineau, A. Tamar, et al. Bayesian reinforcement learning: A survey. *Foundations and Trends® in Machine Learning*, 8(5-6):359–483, 2015.

[68] J. H. Gillula, G. M. Hoffmann, H. Huang, M. P. Vitus, and C. J. Tomlin. Applications of hybrid reachability analysis to robotic aerial vehicles. *The International Journal of Robotics Research*, 30(3):335–354, 2011.

[69] R. Grandia, A. Taylor, A. Singletary, M. Hutter, and A. Ames. Nonlinear model predictive control of robotic systems with control lyapunov functions. *Robotics: Science and Systems XVI*, Jul 2020.

[70] S. Gros. Towards safe reinforcement learning using nmpc and policy gradients: Part ii - deterministic case. *arXiv*, pages 1–12, 2019.

[71] S. Gros, R. Quirynen, and M. Diehl. Aircraft control based on fast non-linear mpc & multiple-shooting. pages 1142–1147, 2012.

[72] W. Gu, K. P. Valavanis, M. J. Rutherford, and A. Rizzo. A Survey of Artificial Neural Networks with Model-based Control Techniques for Flight Control of Unmanned Aerial Vehicles. In *2019 Int. Conf. Unmanned Aircr. Syst.*, pages 362–371. IEEE, 2019.

[73] M. Hachicha, R. B. Halima, and A. H. Kacem. Formal verification approaches of self-adaptive systems: a survey. *Procedia Computer Science*, 159:1853–1862, 2019.

[74] J. J. Harris. F-35 flight control law design, development and verification. In *2018 Aviation Technology, Integration, and Operations Conference*, page 3516, 2018.

[75] J. J. Harris and J. R. Stanford. F-35 flight control law design, development and verification. *2018 Aviat. Technol. Integr. Oper. Conf.*, pages 1–18, 2018.

[76] M. U. Hassan and A. Ali. Robust aircraft performance using hypersurface gain scheduling lqi controller for supermaneuverable aircraft. *Pakistan Journal of Engineering and Technology*, 5(2):17–22, 2022.

[77] S. L. Herbert, M. Chen, S. Han, S. Bansal, J. F. Fisac, and C. J. Tomlin. FaSTrack: A modular framework for fast and guaranteed safe motion planning. In *2017 IEEE 56th Annu. Conf. Decis. Control. CDC 2017*, volume 2018-Janua, pages 1517–1522, 2018.

[78] S. L. Herbert, S. Ghosh, S. Bansal, and C. J. Tomlin. Reachability-based safety guarantees using efficient initializations. *arXiv*, 2019.

[79] L. Hewing, J. Kabzan, and M. N. Zeilinger. Cautious model predictive control using gaussian process regression. *IEEE Transactions on Control Systems Technology*, 28(6):2736–2743, 2020.

[80] L. Hewing, K. P. Wabersich, M. Menner, and M. N. Zeilinger. Learning-based model predictive control: Toward safe learning in control. *Annu. Rev. Control. Robot. Auton. Syst.*, 3(1):269–296, 2020.

[81] S. Heyer, D. Kroezen, and E.-J. Van Kampen. Online adaptive incremental reinforcement learning flight control for a cs-25 class aircraft. In *AIAA Scitech 2020 Forum*, page 1844, 2020.

[82] D. Hill and P. Moylan. The stability of nonlinear dissipative systems. *IEEE transactions on automatic control*, 21(5):708–711, 1976.

[83] N. Hovakimyan, C. Cao, E. Kharisov, E. Xargay, and I. M. Gregory. L 1 adaptive control for safety-critical systems. *IEEE Control Systems Magazine*, 31(5):54–104, 2011.

[84] X. Huang, D. Kroening, W. Ruan, J. Sharp, Y. Sun, E. Thamo, M. Wu, and X. Yi. A survey of safety and trustworthiness of deep neural networks: Verification, testing, adversarial attack and defence, and interpretability. *Computer Science Review*, 37:100270, 2020.

[85] S. Jiffri, J. Mottershead, and J. Cooper. Adaptive feedback linearisation and control of a flexible aircraft wing. In *Topics in Modal Analysis, Volume 7: Proceedings of the 31st IMAC, A Conference on Structural Dynamics, 2013*, pages 683–699. Springer, 2014.

[86] E. N. Johnson and S. K. Kannan. Adaptive flight control for an autonomous unmanned helicopter. *AIAA Guidance, Navigation, and Control Conference and Exhibit*, (August), 2002.

[87] H.-S. Ju and C.-C. Tsai. Longitudinal axis flight control law design by adaptive backstepping. *IEEE Transactions on Aerospace and Electronic Systems*, 43(1):311–329, 2007.

[88] G. Kahn, A. Villaflor, V. Pong, P. Abbeel, and S. Levine. Uncertainty-aware reinforcement learning for collision avoidance. *arXiv e-prints*, 2017.

[89] E. Kaiser, J. N. Kutz, and S. L. Brunton. Sparse identification of nonlinear dynamics for model predictive control in the low-data limit. *Proceedings of the Royal Society A*, 474(2219):20180335, 2018.

[90] E. Kaiser, J. N. Kutz, and S. L. Brunton. Sparse identification of nonlinear dynamics for model predictive control in the low-data limit. *Proc. R. Soc. A Math. Phys. Eng. Sci.*, 474(2219), 2018.

[91] R. Kamalapurkar, P. Walters, J. Rosenfeld, and W. Dixon. *Reinforcement Learning for Optimal Feedback Control*. Springer, 2018.

[92] M. Kaminski. *A rapid robust fault detection algorithm for flight control reconfiguration*. PhD thesis, Technische Universität München, 2019.

[93] T. Keviczky and G. J. Balas. Receding horizon control of an f-16 aircraft: A comparative study. *Control Engineering Practice*, 14(9):1023–1033, 2006.

[94] H. K. Khalil and J. W. Grizzle. *Nonlinear systems*, volume 3. Prentice hall Upper Saddle River, NJ, 2002.

[95] J. Kirschner, M. Mutny, N. Hiller, R. Ischebeck, and A. Krause. Adaptive and safe bayesian optimization in high dimensions via one-dimensional subspaces. In *Proc. 36th Int. Conf. Mach. Learn.*, 2019.

[96] B. Kiumarsi, K. G. Vamvoudakis, H. Modares, and F. L. Lewis. Optimal and autonomous control using reinforcement learning: A survey. *IEEE Transactions on Neural Networks and Learning Systems*, 29(6):2042–2062, 2018.

[97] W. F. . B. U. Koch. *Flight Controller Synthesis Via Deep Reinforcement Learning*. PhD thesis, Boston University, 2019.

[98] M. Krstic, P. V. Kokotovic, and I. Kanellakopoulos. *Nonlinear and adaptive control design*. John Wiley & Sons, Inc., 1995.

[99] T. Krüger, P. Schnetter, R. Placzek, and P. Vörsmann. Fault-tolerant nonlinear adaptive flight control using sliding mode online learning. *Neural Networks*, 32:267–274, 2012.

[100] J. Lenka, Arkdev, and M. K. Sarkar. Nonlinear stabilization: Control contraction metric approach via sum of square programming technique. *2nd International Conference on Energy, Power and Environment: Towards Smart Technology, ICEPE 2018*, pages 1–6, 2019.

[101] K. Leung and I. R. Manchester. Nonlinear stabilization via control contraction metrics: A pseudospectral approach for computing geodesics. *Proc. Am. Control Conf.*, pages 1284–1289, 2017.

[102] F. L. Lewis and D. Liu. *Reinforcement learning and approximate dynamic programming for feedback control*. John Wiley & Sons, 2013.

[103] F. L. Lewis and D. Vrabie. Design Optimal Adaptive Controllers. *IEEE Control Syst.*, 32(November):76–105, 2012.

[104] J. Löfberg. YALMIP: A toolbox for modeling and optimization in MATLAB. *Proc. IEEE Int. Symp. Comput. Control Syst. Des.*, pages 284–289, 2004.

[105] J. Löfberg. Pre- and post-processing sum-of-squares programs in practice. *IEEE Transactions on Automatic Control*, 54(5):1007–1011, 2009.

[106] W. Lohmiller and J.-J. E. Slotine. On contraction analysis for nonlinear systems analyzing stability differentially leads to a new perspective on dynamic systems. *Automatica*, 34(6)(December 1997):683–696, 1997.

[107] W. Lohmiller and J.-J. E. Slotine. On contraction analysis for non-linear systems. *Automatica*, 34(6):683–696, 1998.

[108] B. T. Lopez and J.-J. E. Slotine. Contraction metrics in adaptive nonlinear control. *arXiv preprint arXiv:1912.13138*, 2019.

[109] J. Löfberg. Yalmip - decay rate computation in lti system.

[110] J. Löfberg. Decay rate computation in lti system, 2022.

[111] A. Majumdar, A. A. Ahmadi, and R. Tedrake. Control design along trajectories with sums of squares programming. *Proc. - IEEE Int. Conf. Robot. Autom.*, pages 4054–4061, 2013.

[112] A. Majumdar and R. Tedrake. Funnel libraries for real-time robust feedback motion planning. *Int. J. Rob. Res.*, 36(8):947–982, 2017.

[113] A. Majumdar, R. Vasudevan, M. M. Tobenkin, and R. Tedrake. Convex optimization of nonlinear feedback controllers via occupation measures. *The International Journal of Robotics Research*, 33(9):1209–1230, 2014.

[114] D. Malyuta, T. P. Reynolds, M. Szmuk, T. Lew, R. Bonalli, M. Pavone, and B. Acikmese. Convex Optimization for Trajectory Generation. 2021.

[115] I. R. Manchester and J.-J. E. Slotine. Control contraction metrics and universal stabilizability. *IFAC Proceedings Volumes*, 47(3):8223–8228, 2014.

[116] I. R. Manchester and J.-J. E. Slotine. Control contraction metrics, robust control and observer duality. *arXiv*, 2014.

[117] I. R. Manchester and J.-J. E. Slotine. Control contraction metrics: Convex and intrinsic criteria for nonlinear feedback design. *IEEE Transactions on Automatic Control*, 62(6):3046–3053, 2017.

[118] I. R. Manchester and J. J. E. Slotine. Robust control contraction metrics: A convex approach to nonlinear state-feedback control. *IEEE Control Systems Letters*, 2(3):333–338, 2018.

[119] I. R. Manchester, J. Z. Tang, and J.-J. E. Slotine. Unifying robot trajectory tracking with control contraction metrics. In *Robotics Research*, pages 403–418. Springer, 2018.

[120] Z. Marinho, B. Boots, A. Dragan, A. Byravan, G. J. Gordon, and S. Srinivasa.

Functional gradient motion planning in reproducing kernel hilbert spaces. In *Proceedings of Robotics: Science and Systems*, AnnArbor, Michigan, June 2016.

[121] D. T. McRuer. Pilot-induced oscillations and human dynamic behavior. Technical report, NASA, 1995.

[122] A. Megretski. Spot (systems polynomial optimization tools) manual, 2010.

[123] M. W. Mehrez, G. K. Mann, and R. G. Gosine. Comparison of stabilizing NMPC designs for wheeled mobile robots: An experimental study. *MERCon 2015 - Moratuwa Eng. Res. Conf.*, (September):130–135, 2015.

[124] M. W. Mehrez, K. Worthmann, J. P. Cenerini, M. Osman, W. W. Melek, and S. Jeon. Model Predictive Control without terminal constraints or costs for holonomic mobile robots. *Rob. Auton. Syst.*, 127(February), 2020.

[125] F. Meng, D. Wang, P. Yang, G. Xie, and C. Romero-Meléndez. Application of sum of squares method in nonlinear h control for satellite attitude maneuvers. *Complexity*, 2019, 2019.

[126] I. Mitchell. A toolbox of level set methods, 2004.

[127] I. M. Mitchell and J. A. Templeton. A toolbox of Hamilton- Jacobi solvers for analysis of nondeterministic continuous and hybrid systems. *Lect. Notes Comput. Sci.*, 3414:480–494, 2005.

[128] H. Mo and G. Farid. Nonlinear and adaptive intelligent control techniques for quadrotor uav – a survey. *Asian Journal of Control*, 21(2):989–1008, 2019.

[129] M. A. Mohd Basri, A. R. Husain, and K. a. Danapalasingam. Intelligent adaptive backstepping control for MIMO uncertain non-linear quadrotor helicopter systems. *Transactions of the Institute of Measurement and Control*, 37(3):345–361, 2015.

[130] E. A. Morelli. Global nonlinear parametric modeling with application to f-16 aerodynamics. In *Proceedings of the American Control Conference*, pages 1–6, Pennsylvania, 1998.

[131] E. A. Morelli. Global nonlinear parametric modelling with application to f-16 aerodynamics. In *Proceedings of the 1998 American Control Conference. ACC (IEEE Cat. No. 98CH36207)*, volume 2, pages 997–1001. IEEE, 1998.

[132] M. R. Mortazavi and A. Naghash. Pitch and flight path controller design for f-16 aircraft using combination of lqr and ea techniques. *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, 232(10):1831–1843, 2018.

[133] K. P. Murphy. *Machine Learning A Probabilistic Perspective*. MIT Press, London, England, 2012.

[134] E. Najafi, R. Babuška, and G. A. Lopes. A fast sampling method for estimating the domain of attraction. *Nonlinear Dynamics*, 86(2):823–834, 2016.

[135] NASA/JPL-Caltech. Nasa's insight lander, 2018. [Online; accessed April 16, 2021].

[136] L. Nepomuceno, É. A. Moura, M. A. Morales, R. G. Silva, and L. C. Góes. An lqr-lmi longitudinal stability augmentation system for a subscale fighter aircraft with variable center of gravity position. In *AIAA AVIATION 2022 Forum*, page 4059, 2022.

[137] L. T. Nguyen. *Simulator study of stall/post-stall characteristics of a fighter airplane with relaxed longitudinal static stability*, volume 12854. National Aeronautics and Space Administration, 1979.

[138] D. T. Ocaña, H.-S. Shin, and A. Tsourdos. Development of a nonlinear recon-

figurable f-16 model and flight control systems using multilayer adaptive neural networks. *IFAC-PapersOnLine*, 48(9):138–143, 2015.

[139] A. Papachristodoulou, J. Anderson, G. Valmorbida, S. Prajna, P. Seiler, and P. Parrilo. Sostools version 3.00 sum of squares optimization toolbox for matlab. *arXiv*, 2013.

[140] P. Patrinos and A. Bemporad. An accelerated dual gradient-projection algorithm for embedded linear model predictive control. *IEEE Trans. Automat. Contr.*, 59(1):18–33, 2014.

[141] A. S. Polydoros and L. Nalpantidis. Survey of model-based reinforcement learning: Applications on robotics. *Journal of Intelligent and Robotic Systems: Theory and Applications*, 86(2):153–173, 2017.

[142] A. Ray, J. Achiam, and D. Amodei. Benchmarking safe exploration in deep reinforcement learning. *SemanticScholar*, 2019.

[143] B. Recht. A tour of reinforcement learning: The view from continuous control. *Annual Review of Control, Robotics, and Autonomous Systems*, 2(1):253–279, 2018.

[144] S. M. Richards, F. Berkenkamp, and A. Krause. The lyapunov neural network: Adaptive stability certification for safe learning of dynamical systems. In *Conference on Robot Learning*, pages 466–476. PMLR, 2018.

[145] U. Rosolia and F. Borrelli. Learning how to autonomously race a car: A predictive control approach. *IEEE Transactions on Control Systems Technology*, 28(6):2713–2719, 2020.

[146] C. Rouff, R. Buskens, L. Pullum, X. Cui, and M. Hinchey. The adaptiv approach to verification of adaptive systems. In *Proceedings of the fifth international c* conference on computer science and software engineering*, pages 118–122, 2012.

[147] S. Russell, Richard. Non-linear f-16 simulation using simulink and matlab. *University of Minnesota, Version*, 1, 2003.

[148] S. Saria and A. Subbaswamy. Tutorial: Safe and reliable machine learning. 2019.

[149] M. Schillinger, B. Hartmann, P. Skalecki, M. Meister, D. Nguyen-Tuong, and O. Nelles. Safe active learning and safe bayesian optimization for tuning a pi-controller. *IFAC-PapersOnLine*, 50(1):5967–5972, 2017.

[150] R. Self, M. Harlan, and R. Kamalapurkar. Online inverse reinforcement learning for nonlinear systems. In *2019 IEEE conference on control technology and applications (CCTA)*, pages 296–301. IEEE, 2019.

[151] M. Selim, A. Alanwar, M. W. El-Kharashi, H. M. Abbas, and K. H. Johansson. Safe reinforcement learning using data-driven predictive control. In *2022 5th International Conference on Communications, Signal Processing, and their Applications (ICCSPA)*, pages 1–6. IEEE, 2022.

[152] S. Seshagiri and E. Promtun. Sliding mode control of f-16 longitudinal dynamics. In *2008 American control conference*, pages 1770–1775. IEEE, 2008.

[153] B. Settles. Active Learning Literature Survey. Technical Report 26 January, University of Wisconsin-Madison, 2010.

[154] S. N. Sheldon and C. Osmon. Piloted simulation of an f-16 flight control system designed using quantitative feedback theory. *International Journal of Robust and Nonlinear Control: IFAC-Affiliated Journal*, 9(12):841–855, 1999.

[155] V. Sindhwani, S. Tu, and M. Khansari. Learning Contracting Vector Fields For Stable Imitation Learning. *arXiv e-prints*, 2018.

[156] S. Singh, A. Majumdar, J. Slotine, and M. Pavone. Robust online motion planning via contraction theory and convex optimization. pages 5883–5890, 2017.

[157] S. Singh, A. Majumdar, J. J. Slotine, and M. Pavone. Robust online motion planning via contraction theory and convex optimization. *Proc. - IEEE Int. Conf. Robot. Autom.*, pages 5883–5890, 2017.

[158] S. Singh, S. M. Richards, V. Sindhwani, J.-J. E. Slotine, and M. Pavone. Learning stabilizable nonlinear dynamics with contraction-based regularization. *The International Journal of Robotics Research*, 2020.

[159] S. Singh, V. Sindhwani, J.-J. E. Slotine, and M. Pavone. Learning Stabilizable Dynamical Systems via Control Contraction Metrics. *arXiv e-prints*, page arXiv:1808.00113, July 2018.

[160] S. Singh, V. Sindhwani, J.-J. E. Slotine, and M. Pavone. Learning stabilizable dynamical systems via control contraction metrics. In *International Workshop on the Algorithmic Foundations of Robotics*, pages 179–195. Springer, 2018.

[161] S. W. Smith, H. Yin, and M. Arcak. Continuous abstraction of nonlinear systems using sum-of-squares programming. pages 8093–8098, 2019.

[162] S. Song, J. F. Whidborne, M. Lone, and A. Molina-Cristobal. Multi-objective optimal longitudinal flight control system design for large flexible transport aircraft. In *2018 UKACC 12th International Conference on Control (CONTROL)*, pages 81–86. IEEE, 2018.

[163] L. Sonneveldt, E. R. van Oort, Q. Chu, and J. Mulder. Nonlinear adaptive flight control law design and handling qualities evaluation. In *Proceedings of the 48h IEEE Conference on Decision and Control (CDC) held jointly with 2009 28th Chinese Control Conference*, pages 7333–7338. IEEE, 2009.

[164] L. Sonneveldt, E. R. Van Oort, Q. P. Chu, C. C. De Visser, J. A. Mulder, and J. H. Breeman. Lyapunov-based fault tolerant flight control designs for a modern fighter aircraft model. *AIAA Guidance, Navigation, and Control Conference and Exhibit*, (August):1–23, 2009.

[165] U. M. Specs. Mil-f-8785c: Flying qualities of piloted airplanes. Technical Report November 1980, 1980.

[166] R. M. Stefanescu, C. L. Prioroc, and A. M. Stoica. Weighting matrices determination using pole placement for tracking maneuvers. *UPB Scientific Bulletin, Series D: Mechanical Engineering*, 75(2):31–40, 2013.

[167] B. L. Stevens, F. L. Lewis, and E. N. Johnson. *Aircraft control and simulation: dynamics, controls design, and autonomous systems*. John Wiley & Sons, 2015.

[168] Y. Sui, A. Gotovos, J. W. Burdick, and A. Krause. Safe exploration for optimization with gaussian processes. *32nd International Conference of Machine Learning, ICML 2015*, 2:997–1005, 2015.

[169] Y. Sui, vincent Zhuang, J. Burdick, and Y. Yue. Stagewise safe Bayesian optimization with Gaussian processes. *Proceedings of the 35th International Conference on Machine Learning*, 80:4781–4789, 10–15 Jul 2018.

[170] D. Sun, S. Jha, and C. Fan. Learning certified control using contraction metric. *arXiv preprint arXiv:2011.12569*, 2020.

[171] D. Sun, S. Jha, and C. Fan. Learning certified control using contraction metric. In *Conference on Robot Learning*, pages 1519–1539. PMLR, 2021.

[172] R. S. Sutton and A. G. Barto. *Reinforcement learning: An introduction*. MIT press, 2018.

[173] R. Tedrake. Underactuated robotics: Algorithms for walking, running, swimming, flying, and manipulation (course notes for mit 6.832).

[174] R. Tedrake. *Underactuated Robotics*. 2022.

[175] R. Tedrake, I. R. Manchester, M. Tobenkin, and J. W. Roberts. LQR-trees: Feedback motion planning via sums-of-squares verification. *Int. J. Rob. Res.*, 29(8):1038–1052, 2010.

[176] H. Tokutake and E. Itoh. Flight controller design taking account of control anticipation parameter. In *Proceedings of the 41st SICE Annual Conference. SICE 2002.*, volume 3, pages 1826–1829. IEEE, 2002.

[177] J. L. Townsend and P. E. Blatt. New mil-f-9490d requirements and implications on future flight control design. *Journal of Aircraft*, 13(9):670–675, 1976.

[178] H. Tsukamoto and S. Chung. Neural contraction metrics for robust estimation and control: A convex optimization approach. *IEEE Control Systems Letters*, 5(1):211–216, 2021.

[179] H. Tsukamoto and S. J. Chung. Robust controller design for stochastic nonlinear systems via convex optimization. *IEEE Transactions on Automatic Control*, pages 1–1, 2020.

[180] A. Tunik and O. Nadsadna. Robust digital gain-scheduling control of the uav altitude. In *2017 IEEE 4th International Conference Actual Problems of Unmanned Aerial Vehicles Developments (APUAVD)*, pages 175–179. IEEE, 2017.

[181] J. Umenberger, M. Ferizbegovic, T. B. Schön, and H. Hjalmarsson. Robust exploration in linear quadratic reinforcement learning. *Advances in Neural Information Processing Systems*, 32, 2019.

[182] H. van Hasselt, Y. Doron, F. Strub, M. Hessel, N. Sonnerat, and J. Modayil. Deep reinforcement learning and the deadly triad. *arXiv e-prints*, 2018.

[183] L. Vandenberghe and S. Boyd. Semidefinite programming. *SIAM review*, 38(1):49–95, 1996.

[184] D. Vrabie, O. Pastravanu, M. Abu-Khalaf, and F. L. Lewis. Adaptive optimal control for continuous-time linear systems based on policy iteration. *Automatica*, 45(2):477–484, 2009.

[185] K. P. Wabersich, L. Hewing, A. Carron, and M. N. Zeilinger. Probabilistic model predictive safety certification for learning-based control. pages 1–11, 2019.

[186] K. P. Wabersich and M. N. Zeilinger. Safe exploration of nonlinear dynamical systems: A predictive safety filter for reinforcement learning. *arXiv*, 2018.

[187] K. P. Wabersich and M. N. Zeilinger. Scalable synthesis of safety certificates from data with application to learning-based control. *2018 Eur. Control Conf. ECC 2018*, pages 1691–1697, 2018.

[188] N. Wahid and N. Hassan. Self-tuning fuzzy pid controller design for aircraft pitch control. In *2012 Third International Conference on Intelligent Systems Modelling and Simulation*, pages 19–24. IEEE, 2012.

[189] D. Wang, H. He, and D. Liu. Adaptive critic nonlinear robust control: A survey. *IEEE Transactions on Cybernetics*, 47(10):3429–3451, 2017.

[190] R. Wang, R. Tóth, and I. R. Manchester. Virtual control contraction metrics: Convex nonlinear feedback design via behavioral embedding. *arXiv preprint arXiv:2003.08513*, 2020.

[191] S. Wang, G. Zhu, X. Chen, X. Zhang, J. Xu, X. Li, and H. Cao. Neural Networks-based Robust Adaptive Dynamic Surface Sliding Mode Control of Flight Path Angle with Tracking Error Constraints. *2019 IEEE 28th International Symposium on Industrial Electronics (ISIE)*, pages 587–592, 2019.

[192] Y. Wang and S. Boyd. Fast Model Predictive Control Using Online Optimization. *IEEE Trans. Control Syst. Technol.*, 18(2):267–278, 2010.

[193] C. Wilkinson, J. Lynch, R. Bharadwaj, and K. Woodham. Verification of adaptive systems. *Federal Aviation Administration, DOT/FAA/TC-16/4*, 2016.

[194] W. Xiang and T. T. Johnson. Reachability analysis and safety verification for neural network control systems. *arXiv*, pages 1–21, 2018.

[195] W. Xiang, P. Musau, A. A. Wild, D. M. Lopez, N. Hamilton, X. Yang, J. Rosenfeld, and T. T. Johnson. Verification for machine learning, autonomy, and neural networks survey. *arXiv*, pages 1–51, 2018.

[196] F. Yang. Fast model predictive control using online optimization. *IEEE Transactions on Control Systems Technology*, 18(2):267–278, 2010.

[197] W. Yang, M. Hammoudi, G. Herrmann, M. Lowenberg, and X. Chen. Two-state dynamic gain scheduling control applied to an f16 aircraft model. *International Journal of Non-Linear Mechanics*, 47(10):1116–1123, 2012.

[198] X. Zhao, A. Banks, J. Sharp, V. Robu, D. Flynn, M. Fisher, and X. Huang. A safety framework for critical systems utilising deep neural networks. pages 244–259, 2020.

[199] Y. Zheng, G. Fantuzzi, and A. Papachristodoulou. Sparse sum-of-squares (SOS) optimization : A bridge between DSOS/SDSOS and SOS optimization for sparse polynomials. In *2019 Am. Control Conf.*, pages 5513–5518, 2019.

[200] Y. Zhou, E.-J. Van Kampen, and Q. Chu. Incremental model based heuristic dynamic programming for nonlinear adaptive flight control. *Proceedings of the international micro air vehicles conference and competition (IMAV)*, 2016.

[201] G. Zhu, J. Du, J. Li, and Y. Kao. Robust adaptive nn tracking control for mimo uncertain nonlinear systems with completely unknown control gains under input saturations. *Neurocomputing*, 365:125–136, 2019.