Robustness Analysis of Linear Time-Varying Systems with Application to Aerospace Systems

A Thesis Submitted to The University of Nottingham
for the Degree of Doctor of Philosophy

FELIX BIERTÜMPFEL, MSc

February 2021

In Memory of Matthias Heller.

Abstract

In recent years significant effort was put into developing analytical worst-case analysis tools to supplement the Verification & Validation (V&V) process of complex industrial applications under perturbation. Progress has been made for parameter varying systems via a systematic extension of the bounded real lemma (BRL) for nominal linear parameter varying (LPV) systems to IQCs. However, finite horizon linear time-varying (LTV) systems gathered little attention. This is surprising given the number of nonlinear engineering problems whose linearized dynamics are time-varying along predefined finite trajectories. This applies to everything from space launchers to paper processing machines, whose inertia changes rapidly as the material is unwound. Fast and reliable analytical tools should greatly benefit the V&V processes for these applications, which currently rely heavily on computationally expensive simulation-based analysis methods of full nonlinear models.

The approach taken in this thesis is to compute the worst-case gain of the interconnection of a finite time horizon LTV system and perturbations. The input/output behavior of the uncertainty is described by integral quadratic constraints (IQC). A condition for the worst-case gain of such an interconnection can be formulated using dissipation theory. This utilizes a parameterized Riccati differential equation, which depends on the chosen IQC multiplier. A nonlinear optimization problem is formulated to minimize the upper bound of the worst-case gain over a set of admissible IQC multipliers. This problem can then be efficiently solved using custom-tailored meta-heuristic (MH) algorithms. One of the developed algorithms is initially benchmarked against non-tailored algorithms, demonstrating its improved performance. A second algorithm's potential application in large industrial problems is shown using the example of a touchdown constraints analysis for an autolanded aircraft as was as an aerodynamic loads analysis for space launcher under perturbation and atmospheric disturbance. By comparing the worst-case LTV analysis results with the results of corresponding nonlinear Monte Carlo simulations, the feasibility of the approach to provide necessary upper bounds is demonstrated. This comparison also highlights the improved computational speed of the proposed LTV approach compared to simulation-based nonlinear analyses.

Acknowledgements

I would like to thank my supervisor Professor Harald Pfifer for his support and guidance throughout the doctorate. The numerous discussions on control theory and aerospace were invaluable and shaped this thesis. A thank you also to Professor Atanas Popov, who helped me transition into the UK academic system and find my way inside the University of Nottingham.

Thanks are also due to the European Space Agency (ESA), which provided funding for this doctorate. Within ESA, I would like to especially thank Doctor Samir Bennani. He actively supported this research from the very beginning. Also, he welcomed me for a valuable research visit inside his division at ESA ESTEC.

I further like to thank Doctor Nantiwat Pholdee for his help and support with the meta-heuristic algorithms. His input was priceless for the outcomes of this thesis.

I would like to thank my colleagues and friends who supported me during this journey. Here I want to mention Christian Weiser and Louis Flanagan for their valuable feedback writing this thesis.

I also want to thank my family for their constant love and support. My parents Andrea and Frank, without whose unconditional support, this PhD would not have been feasible.

A special thank you goes to the most valuable person of this PhD, Diana, who carried me through this roller coaster ride. You provided the sunshine in every hour of this PhD.

My final, but greatest, thank you goes to Professor Matthias Heller, a great mentor but a greater man without whom these lines would not be written. He sparked my research interest and paved the way for this thesis. His dedication

was leading by example, and his knowledge invaluable. Words can describe neither the gratefulness I feel for everything you did for me nor the sadness of not writing these lines earlier. I will always be thankful to have called you a friend.

List of Publications

1. Biertümpfel F., Pfifer H., 2018, Worst Case Gain Computation of Linear Time Varying Systems over a Finite Horizon, in: 2nd Conference on Control Technology and Applications [Chapter 5].

2. Biertümpfel F., Pfifer, H., Bennani, S., 2019. Finite Horizon Worst Case Analysis of Launch Vehicles. IFAC-PapersOnLine 52, 31–36. [Chapter 3]

3. Biertümpfel F., Pfifer, H., Bennani, S., 2020. Finite Horizon Analysis of Launch Vehicles Under Mass and Thrust Uncertainty. IFAC-PapersOnLine [Chapter 6]

4. Biertümpfel F., Pholdee, N., Bureerat, S.,Pfifer, H., 2020. Adaptive Boundary Sine Cosine Optimizer With Population Reduction For Robustness Analysis of Finite Time Horizon System, submitted to Applied Soft Computing [Chapter 5]

5. Biertümpfel F., Pholdee, N., Bennani, S.,Pfifer, H., 2021. Finite Horizon Worst Case Analysis of Linear Time Varying Systems Applied to Launch Vehicle, submitted to IEEE Transactions on Control Systems Technology [Chapter 5]

6. Biertümpfel F., Pfifer, H., 2021. Finite Horizon Analysis of Autolanded Aircraft in Final Approach under Crosswind, submitted to Control Engineering Practice [Chapter 6]

7. Biertümpfel F., Pholdee, N., Bennani, S.,Pfifer, H., 2021. Time-Varying Robustness Analysis of Launch Vehicles Under Thrust Perturbations, submitted to Advanced Control for Applications [Chapter 7]

8. Biertümpfel F., Pfifer, H., 2021. Finite Horizon Touchdown Analysis of Autolanded Aircraft under Crosswind, accepted to 4th IFAC Workshop on Linear Parameter-Varying Systems [Chapter 6]

Contents

Abbreviations

BIBO	Bounded Input Bounded Output
BRL	Bounded Real Lemma
ELV	Expendable Launch Vehicle
ESA	European Space Agency
FCS	Flight Control System
FTB	Finite Time Bounded
FTS	Finite Time Stability
GNC	Guidance Navigation and Control
ILS	Instrument Landing System
IQC	Integral Quadratic Constraint
ISA	International Standard Atmosphere
LFT	Linear Fractional Transformation
LMI	Linear Matrix Inequality
LPV	Linear Parameter-Varying
LQR	Linear Quadratic Regulator
LTI	Linear Time-Invariant
LTP	Linear Time-Periodic
LTV	Linear Time-Varying
MH	Meta-Heuristic
MIMO	Multi Input Multi Output
ODE	Ordinary Differential Equation
PD	Proportional-Derivative
PID	Proportional-Integral-Derivative

PSD	Power Spectral Density
RDE	Riccati Differential Equation
SISO	Single Input Single Output
TVC	Thrust Vectoring Control
UAS	Unmanned Aerial Systems
V&V	Validation and Verification
WGS84	World Geodetic System 84

Notation

$\mathbb{R}$	Set of real numbers
$\mathbb{R}^+$	Set of strict positive numbers
$\mathbb{R}_0^+$	Set of real positive numbers including zero
$\mathbb{R}^{n\times m}$	Set of n by m matrices with elements in $\mathbb{R}$
$\mathbb{C}$	Set of complex numbers
$\mathbb{C}^{n\times m}$	Set of n by m matrices with elements in $\mathbb{C}$
$\mathbb{S}^n$	Set of n-dimensional symmetric matrices

$\bar{z}$	Complex conjugate of a complex number
I_n	n-dimensional identity matrix
X^T	Transpose of a matrix
X^*	Complex conjugate transpose of a matrix
X^{-1}	Inverse of a matrix
$X > Y$	X and Y symmetric and $X - Y$ positive definite
$X \geq Y$	X and Y symmetric and $X - Y$ positive semidefinite
$G(j\omega)^\sim$	Para-Hermitian conjugate of a system $(G(s)^\sim(s) := G(-\bar{s})^*)$

$\mathrm{diag}(X, Y)$	Block diagonal matrix, i.e. $\left[\begin{smallmatrix} X & 0 \\ 0 & Y \end{smallmatrix}\right]$
round(.)	Round scalar to next integer

$F_u(.,.)$	Upper LFT
cond(.)	Condition number for inversion of a matrix

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Motivations

In recent years significant resources have been invested into developing guaranteed worst-case analysis tools as a supplement for Verification & Validation (V&V) processes of complex industrial applications. As a model can only represent the actual system to a certain degree, a particular emphasis was put on respecting the influence of perturbations on the system's dynamics. These include, e.g. neglected higher-order dynamics, nonlinearities such as saturations, or infinite-dimensional systems such as time delays. Nevertheless, the centerpiece of V&V processes remained so-called simulation-based approaches such as Monte Carlo simulations or worst-case optimizations conducted on the nonlinear model [1–5]. These methods can be directly deployed on high-fidelity nonlinear system models. However, they require significant computational resources and time as they must cover a large set of possible dynamics. Even more critical, they cannot provide guaranteed worst-cases, but only a probability distribution or a lower bound of the worst case, respectively.

In contrast, linear analysis methods can calculate guaranteed worst-cases. Over the last decade, significant progress has been made for parameter-varying systems, such as (flexible) aircraft whose parameters depend on altitude and airspeed. In this regard, the key development was the systematic extension of the bounded real lemma (BRL) for nominal linear parameter-varying (LPV) systems to integral quadratic constraints (IQCs) [6]. These allowed the user to specify worst-case gain analysis conditions for LPV systems in interconnection with perturbations. The LPV-IQC framework was then successfully applied for worst-case analyses of, e.g. gust loads on flexible aircraft ([7]).

Systems with strictly time-varying dynamics have received significantly less attention, although they are closely related to parameter-varying systems. This is surprising given the fact that this class of systems covers various engineering and control problems. This includes all systems following predefined trajectories between a specified start and endpoint. The linearization of their dynamics along such a specific trajectory results in so-called finite horizon linear time-varying (LTV) systems. Hence, their system matrices are bounded functions of time, defined only between the trajectory's respective start and end point. For these systems, the behavior along the trajectory and their conditions at the final point are of great interest. Prominent aerospace examples are autolanded aircraft in the final approach and space launchers during atmospheric ascent. Focusing on the aircraft example, the final approach and landing is statistically the most dangerous flight segment, accounting for more than 49% of all disastrous accidents, see, e.g. [8]. Autoland systems (AS) were introduced to moderate the risk, primarily for poor visual conditions, at the beginning of the 1950s [9]. These generally employ a runway-based instrument landing system (ILS) to produce a localizer and glideslope signal. These signals provide a reference trajectory tracked by the aircraft's autopilot during the approach. Following the ILS signals presents a classical reference signal tracking problem, which aims to reduce the offset between reference and the corresponding tracked signals. The autoland systems are, in general, designed for specific reference (nominal) dynamics corresponding to a typical aircraft configuration and environmental conditions. However, during the approach and landing, the aircraft's dynamics change depending on the altitude due to the ground effect and altitude-triggered control law changes. This altitude dependence maps to a strict time dependence given the ILS trajectory rendering the problem strictly time-varying over a finite horizon concluded by the touchdown. For certification, the autoland system must satisfy touchdown performance constraints over a large set of different aircraft configurations and environmental conditions. This means for non-nominal dynamics and external disturbances such as wind turbulence. In summary, the autoland problem requires evaluating performance metrics at the end of the trajectory for uncertain time-varying dynamics under wind disturbance.

Given the atmospheric ascent problem, not only the terminal conditions at the separation of the launcher stages are of interest, but also at every point along trajectory. A significant amount of time in the pre-launch preparation of space launchers is spent optimizing the ascent trajectory and the respective tuning of the launcher's controller. One of the primary optimization objectives is reducing the maximal aerodynamic loads on the launcher due to wind disturbance. Additionally, most launch sites impose tight constraints on ascent corridors in the yaw plane to limit or avoid land overflight [10–12]. Until hours before launch, updates are made based on wind data gathered by wind-balloons or launch side specific wind charts to identify a load minimizing trajectory which provides enough safety range regarding land overflight [13]. During the ascent, the launcher's dynamics are highly time-varying as it accelerates through different layers of the atmosphere along the calculated trajectory. At the same time, the launcher's aerodynamic parameters are difficult to estimate, especially in the transonic region. Furthermore, perturbations in the launcher's thrust due to irregularities in the combustion process significantly influence its dynamics. This renders the launcher ascent a tracking problem for which the lateral offset and aerodynamic loads for significantly uncertain time-varying dynamics under wind disturbance must be evaluated for certification.

However, the recent LPV analysis approaches fail to evaluate (strictly) time-varying dynamics as they, by definition, cover an infinite amount of possible trajectories inside the defined parameter set [14]. In addition, their analysis conditions are specified for infinite time horizons, i.e. the systems (steady) behavior approaching infinite times is analyzed. Thus, (terminal) conditions at a certain point of a finite trajectory cannot be evaluated with these methods. These limitations render them insufficient for finite horizon LTV problems such as the mentioned launcher ascent or autolanded aircraft. Thus, the current V&V processes of both autoland systems and space launchers rely solely on simulation-based methods [1, 15]. Both problems alone justify research in the development of suitable LTV analysis tools. For example, more than 44000 flights are covered every day by the FAA alone [16]. Most of the handled approaches are automated, requiring elaborate and very reliable autoland systems, which trigger ongoing research [17, 18]. With an expected growth in revenue from the current $424 billion to more than $1.4 trillion by 2030, space launcher research and development are of renewed interest. This growth has been driven by the emergence of private companies providing advanced space transportation technology [19, 20]. Hence, developing fast and reliable tools to analyze finite horizons LTV under uncertainty and external disturbances is mandatory to support future V&V processes of time-varying systems over finite horizons.

1.2 Literature Review

The introduction of an analysis framework for systems with highly time-varying dynamics over finite trajectories requires a deep understanding of the control problem itself and the evolution of linear system analyses. Particularly when identifying the limitations and shortcomings of the existing linear (worst case) analysis methods. Hence, this Section starts with a short summary of progress in stability and robustness analysis methods for linear systems, from their emergence in the late 19th century to the most recent advances. Afterwards, the rise in prominence of nonlinear optimization methods due to the ever-growing computational power will be discussed. Here, the focus is on meta-heuristics, which present an efficient and flexible approach to solve complex nonlinear problems. Hence, the state-of-the-art analysis of aircraft in final approach and space launchers in atmospheric ascent will be detailed.

1.2.1 History and Recent Developments in Linear System Analysis

Initially formulated for linear time invariant (LTI) systems the following terms are commonly used in the context of system analysis [21]:

1. Nominal stability: The nominal, i.e. unperturbed, system is stable
2. Nominal performance : The nominal system complies with the imposed performance criteria.
3. Robust stability: The perturbed system remains stable up to the worst case model perturbations.
4. Robust performance: The perturbed system fulfills the performance criteria up to the worst case model perturbations.

In general, these criteria are checked in the order given above, as each subsequent point can only be assessed if the previous holds true, i.e. nominal stability is a necessary condition for nominal performance and so forth.

Hurwitz and Lyapunov Stability Criteria

Methods for determining the nominal stability of a linear time-invariant system date back as early as 1876 when Edward Routh showed that a system's stability can be determined via the roots of the characteristic polynomial [22]. Independently, an equivalent approach was proposed by Adolf Hurwitz. This stability criterion is now known as the Routh-Hurwitz stability criterion and is an easy tool to evaluate the stability of an LTI system [23].

In the context of the theory applied in this thesis, the year 1892 is significant as it is when Aleksandr Lyapunov first treated the stability theory of solutions of ordinary differential equations (ODE) in his dissertation [24] and later in [25]. His theory did not analyze a nonlinear system directly. Instead, the linear surrogate's behavior is analyzed. He proposed two methods to prove the stability of a system. Lyapunov's second method is now commonly known as Lyapunov stability criterion, or the direct method [25, 26]. The direct method states that if a system starts close to a set point x_s and remains close to it for all times, the system is Lyapunov stable. In the cases where it approaches x_s, the system is called asymptotically stable. Note that this approach is not limited to continuous-time and linear systems but covers nonlinear systems as well. Given the scope of the thesis, the focus going forward is mainly on linear systems.

For almost half a century, these groundbreaking results on system stability were forgotten until rediscovered by Nikolay Chetaev in the 1930s. Chetaev himself significantly contributed to the mathematical stability theory in [27]. It was further shown in [28] that when a system is Lyapunov stable, it also remains stable under the influence of small disturbances d. This marks the origin of the common bounded input bounded output (BIBO) stability, and performance analyses. Lyapunov's second method rose to prominence in the 1950s. Primarily for use in the stability analysis of guidance systems as it was able to respect their significant system nonlinearities.

Classical Frequency Domain Criteria

In contrast to Lyapunov's time-domain approach, frequency-domain methods analyzing transfer functions of linear systems emerged in western control theory in the 1930s. In 1930 and 1932, independently, the same stability criteria were proposed by Felix Strecker [29, 30] and Harry Nyquist [31], respectively. It presents a graphical method to prove a dynamic system's stability and is widely known as the (Strecker-) Nyquist stability criterion. This criterion is suitable for systems represented by non-rational functions, e.g. including time delays. It allows for the stability analysis of closed-loop (negative) feedback interconnections based on the open-loop transfer. It was later generalized to include multi input multi output (MIMO) systems, see e.g. [32], and systems with time-varying parameters in, see e.g. [33]. Although a very general approach, determining the stability and the respective margins is not very intuitive using the Nyquist criterion.

In 1938, asymptotic phase and magnitude plots were proposed by Hendrik Bode [34]. These showed the stability of systems using frequency domain concepts and quickly assess phase and gain margins, although in two separate plots. Due to their accessibility and fastness, so-called Bode plots are up to now one of the main approaches for LTI system analyses and control design. However, Bode plots cannot handle non-rational functions and transfer functions with right half-plane singularities. Furthermore, it is strictly limited to single input single output (SISO) systems.

In 1947, Nathanial Nichols introduced another graphical method, now commonly known as Nichols plots [21]. Derived from the Nyquist plots, but displayed in real coordinates, they combined most of the Bode and Nyquist Plots' advantages.

Kalman-Yakubovich-Popov Lemma

Based on Lyapunov's second method, the Kalman-Yakubovich-Popov (KYP) lemma, also known as positive real lemma, was formulated in the early 1960s. Firstly, it was stated and proved by Vladimir Yakubovich in [35] ([36]; English reprint) as strict frequency inequality. It can be seen as a generalization of Lyapunov's equations. In [37], Rudolf Kalman proved the lemma for the non-strict frequency inequality. This paper also made a connection to the solution of the Lur'e equation. The extension to MIMO systems was prosed in [38] and independently by Vasile Popov in [39]. A recent summary on the advances of the KYP can be found in [40].

However, all approaches mentioned so far consider the nominal stability and performance of an LTI system. Kalman's proof of the KYP including the Lur'e equation laid the foundation to included uncertainties/perturbations into the analysis. The Lur'e type equations were introduced 1951 by Anatoliy Lure in [41] after introducing the theory of absolute stability earlier in [42]. Solvability conditions on this type of equations appear in the context of dissipativity of linear systems [43, 44], the spectral factorization [45], and balancing-related model reduction [46]. Furthermore, they are central in the solution of the infinite horizon linear optimal control problem [47–49]. Hence, they are one of the cornerstones from which the work in this thesis originates.

Lur'e Problem

Crucial in the context of robust stability analysis is the so-called Lur'e problem. For its thorough historical treatment, the reader is referred to [50]. It describes an LTI system in feedback interconnection with a memoryless, sector nonlinearity. Two absolute stability criteria are directly connected to the Lur'e problem, the circle criterion [51–53] and the Popov criterion original proposed in [54, 55]) and generalized by [56]. The circle criterion can be understood as a generalization of the Nyquist criterion covering Lur'e type problems, see e.g. [57] and thus presents a graphical analysis method.

A generalization to solve the Lur'e problem was proposed in 1966 by R.P O'Shea in [58] and improved in [59] using a class of multipliers. These multipliers are transfer functions, which translate a nonlinear (passivity-type) problem into a linear (passivity-type) problem that is easier to solve. For a historical context and the significance of the approach, the reader is referred to [60]. A formalization of the approach was introduced in 1968 by George Zames and Peter Falb in [61]. Generally, the multiplier approach aims to identify multipliers that hold for a broad range of nonlinearities. Contrarily to the multiplier theory's popularity in combination with Lyapunov theory in the 1960s, see e.g. [62, 63], the textbook approach regarding absolute stability became the circle and Popov criteria. Especially, the work of O'Shea was widely forgotten until the mid-1990s, see e.g. [64].

Small Gain Theorems

Regarding the stability between two interconnected stable systems, another significant analysis method are small gain theorems introduced in e.g. [65] and [66]. This is a crucial theorem concerning the robust stability and performance analysis of systems in interconnection with an uncertainty/ perturbation/ nonlinearity. It is defined by the boundedness of the connections L2 gain by 1. In [66], a passivity theorem was proposed, which provides an equivalent stability statement, as shown in [67]. A more general small gain theorem was later introduced in [68] by Desoer and Vidyasagar.

Structured Singular Value

In the early 1980s, there was a resurgence of frequency-domain methods based on the singular value analyses (H_∞-performance/optimal control) focusing on the question of stability and achievable performance under uncertainty for LTI mulit-input multi-output (MIMO) systems, see e.g. [69–72]. The most considerable impact regarding the robustness analysis of LTI MIMO systems was in 1982 by John Doyle [73] and Michael Safonov [74]. The proposed structured singular value is commonly denoted by μ [75]. It allowed for the first time to evaluate both the robust stability and performance in a single robustness framework. Technically, Safonov's margin k_s was formulated inverse to Doyle's, i.e. $k_s = 1/\mu$, making it more intuitive to interpret. Generally, the value of μ cannot be calculated exactly, only its lower [76, 77] and upper bound [75]. The former states the guaranteed uncertainty value for which the system becomes unstable/violates performance requirements. The latter defines the smallest uncertainty for which instability/performance violations can occur. The lower bounds are especially useful as they provide the values for the most troublesome uncertainty combinations.

Initially, the framework was formulated for structured complex (dynamic) uncertainties (see e.g. [78] for an extensive discussion), and efficient computational tools were commercially available by 1991 [79]. The latter is significant as this was the first time elaborate robustness analysis tools were made readily available for a broad range of engineering problems. It is also a driving factor for its vast popularity and acceptance in control engineering. The framework and computational tools were later extended to cover parametric uncertainties [80], i.e. the uncertainty is bounded and constraint real, and problems with mixed uncertainties [81, 82]. The μ-framework is limited to dynamic and real perturbations/ uncertainties and cannot handle nonlinearities or infinite-dimensional systems (e.g. time delays).

Integral Quadratic Constraints in the Frequency-Domain

In 1995, the integral quadratic constraint (IQC) framework for LTI systems under perturbation was introduced by Alexandre Megretski and Anders Rantzer in [83]. It provides a unified frequency-domain approach to cover various types of perturbations, such as dynamic and real uncertainties, time delays, and real nonlinearities such as saturations. Explicitly, it covers the Lur'e problem, for which it analyzes multiple perturbations at the same time. The proposed stability argument is based on passivity/dissipation arguments, which can be easily tested in the linear matrix inequality (LMI) framework ([84]). It can be understood as the unification of Yakubovichs work with the western multiplier approach e.g. [66].

Numerous computational tools exist for the IQC based stability and performance analysis of LTI systems, see e.g. [85–88]. It has to be noted that an IQC approach can be dated back to Yakubovich [63] who applied frequency and time domain conditions but is strictly limited to Lur'e's problem, see [60] for more information. The dissipation theory was introduced in 1972 by Jan Willems in [43, 89]. It proposes a storage function that functions as a Lyapunov function of the closed-loop system. However, the advantages of Megretski's and Rantzer's approach in [83] are numerous., as it covers multipliers with non-canonical factorization due to a homotopy argument, Zame-Falb multipliers [83], and Popov multipliers [90]. For the latter, it also provides further properties, such as slope restrictions [91] or conic sector conditions [92]. In [83], a library containing IQC multipliers Π is given covering common types of perturbations.

Nevertheless, the analysis condition in [83] is limited to LTI systems due to its formulation in the frequency domain. Thus, they are not directly applicable to cover linear parameter-varying systems or linear time-varying systems. The former became a focal research subject robustness analysis in the early 1990s, due to the work of Jeff Shamma and Michael Athans [93, 94]. These systems are (more) naturally investigated in the time domain.

Linear Time-Varying Systems

Linear time-varying systems are usually divided into the linear time-periodic (LTP) and finite horizon case. The former is covering systems whose system matrices are periodic matrix functions of time. In general, they cover infinite time horizons. Typical examples of this kind of dynamics are the flapping of helicopter rotor blades in forward flight [95, 96], wind turbines [97], and spinning satellites [98]. Results for the nominal stability and performance can be found in [99–102], mainly using a combination of Floquet's [103] and Lyapunov's [25] work on the solution of LTP systems. They are now known as Floquet-Lyapunov theory [104]. The literature covering these systems is rich, mainly due to comparable behavior to LTI systems. This is in sharp contrast to the finite horizon case.

Control systems which can be approximated by finite horizon LTV systems are all nonlinear systems following predefined finite trajectory with changing dynamics along said trajectory. This includes terminal guidance systems [105], controlled swarm robots [106], robotic manipulators with varying loads [107], newspaper presses as the inertias change during unwinding, or the atmospheric flight phase of space launchers [108]. The linearization of these systems leads to a linear system whose system matrices are bounded, continuous matrix functions of time. As opposed to LTI and LTP systems, stability is no longer a meaningful criterion to assess for finite horizon LTV systems. This is a consequence of the finite time extent of the analysis, as no (linear) system can grow unbounded in finite time.

Nevertheless, determining if the system's states stay in prescribed bounds over a given horizon, i.e. that non-observed/observable states grow arbitrarily large, still provides practical information. In this context, *practical stability* was proposed and investigated by Joseph La Salle, and Solomon Lefschetz in [26] using Lyapunov's direct method.

A similar concept, but with a different analysis horizon, called *finite-time-stability* (FTS) was proposed earlier by Kamenkov in [109] (generalized in [110]) and [111]. Both approaches cover nonlinear as well as linear systems. Focusing explicitly on finite horizon LTV systems, FTS was investigated in e.g. [112, 113] and [114] under the notation *short-time-stability* and *finite-time stability*, respectively. However, these approaches are limited to autonomous systems and do not provide information on the system's input/output behavior.

Leonard Weiss overcame this limitation in [115] by introducing perturbation signals into the finite-time analysis of nonlinear systems. This led to the introduction of finite-time BIBO stability. A comparable approach for LTV system, called *finite-time-bounded* (FTB) *stability*, was introduced in [116] based on a LMI feasibility problem. The approach respects non-zero initial conditions but is limited to constant disturbances. It further shows strong similarities to the linear parameter varying systems introduced later in this literature review. A significant drawback are the extremely short time horizons analyzed by the method (max. 1s). In the context of analysis of industrial examples like space launchers, this approach does not appear promising. Moreover, they are not formulated in a worst-case context, which is imperative for the provision of strict upper bounds on the worst-case performance. For interested readers, a thorough review of FTS and FTB is given in [117] as the overview here is for completeness only.

A more suitable approach for industrial problems covered in this thesis is derived from the advances of finite horizon LTV H_∞ stabilizing controllers in [118–120]. These results are based on the solvability of two Riccati differential equations. In the context of H_∞ input/output operator norms, the induced input/output norms for general linear systems with L2 input/output signals are developed in [121]. Hence, the proposed induced norms relate to worst-case problems, which is key for the analyses conducted in this thesis. The results are based on the classical maximum principle [122] for which it is shown that the solution is equivalent to the solvability of a boundary value Hamiltonian system or an equivalent indefinite Riccati differential equation. Note that the application of linear quadratic costs/optimization problems for finite horizon problems and related solvability conditions on RDEs trace back to the linear quadratic regulator (LQR) optimal control problem, see e.g. [123]. However, the definiteness of the matrix functions concerned are hugely different, rendering a significant amount of the available existence theorems for (time-varying) RDEs infeasible [124–126]. Thus, they require the explicit integration of the RDE.

The results in [121] influenced the extension of the BRL to (finite horizon) LTV systems [127, 128], also called the strict BRL. It provides a condition for the upper bound on the worst-case gain from any norm bounded disturbance input to the performance output. This relates the LTV analysis to an actual BIBO performance approach, more suitable for an engineering problem than FTS/FTB. This upper bound's existence is connected to the existence of a time-varying RDE over the complete analysis horizon. Note that due to the finiteness of the analysis, the gain should always exist. However, especially for unstable systems, the RDE can be hard to solve. This matter was recently addressed in [129] by proposing new algorithms solving the analysis condition. The approach in [127] is already closer to the KYP lemma and dissipation theory. Moreover, the approach shows better applicability over larger horizons compared to the FTB approach. Furthermore, the use of bounded rather than constant inputs allows for modeling a broader range of disturbance signals. Hence, the approach in [127] presents the logical foundation for an extension to robust performance analyses in this thesis.

Linear Parameter-Varying Systems

LPV systems are a special case of LTV systems whose system matrices depend on time-varying parameters over an infinite horizon.

Typical examples for engineering problems which can be modeled/approximated as LPV systems are aircraft or helicopters parameterized over altitude and/or velocity or the dynamic pressure, see e.g [130–132], aircraft-servo-elastics and flutter over the same scheduling parameters, e.g. [133, 134], or turbofan engines over thrust levels, e.g. [135].

A typical example of an engineering problem that can be approximated as an LPV system is an aircraft parametrized over altitude, velocity, and dynamic pressure ranges [130–132]. Other common examples include aircraft-servo-elastics and flutter over the same parameters [133, 134], as well as turbofan engines over thrust levels [135].

LPV systems are distinguished into three categories. Firstly, *polytopic* LPV systems, whose parameters are only bounded in polytopes, e.g. [136]. Secondly, *linear fractional transformation* (LFT) LPV systems, whose system matrices depend rationally on the parameters, e.g. [137–139]. Thirdly, so-called *gridded* LPV systems where the system matrices can arbitrarily depend on the parameters, e.g [14, 140]. The latter are the most general and consequently sparked the largest research interest. A generalization of the LTI BRL for gridded LPV systems was proposed by Fen Wu in [14], which shows obvious similarities to the LTV BRL.

Integral Quadratic Constraints in the Time-Domain

Nevertheless, the analysis frameworks introduced so far for both finite horizon

LTV and LPV systems are limited to their nominal dynamics. A major step in resolving this limitation were the advances in connecting the IQC and passivity/dissipation theory translating the IQC framework into the time-domain. Starting with a new definition of dissipativity with respect to quadratic forms in [141], Lyapunov theory was loosened from its absolute stability and autonomous characteristics to a more general form to analyze the input/output behavior of interconnections. In [142], the use of multipliers in both the dissipativity and Lyapunov framework were compared and their equivalence for certain multipliers were proven. By using mild assumptions on the multipliers the work to merge IQC frequency methods and time-domain Lyapunov conditions was started in [143] for dynamic uncertainties. A link between the dissipation theory and Lyapunov's theorem was established in [144]. The final step to translate the IQC frequency domain argument into the time domain was provided by Peter Seiler in [145]. It proves that for most IQC multipliers found in the literature e.g. [83], a J-spectral factorization [146] exists. Given an adequate factorization, the time domain IQC holds for all finite times, and the frequency domain inequality can be transformed using the KYP LMI ([147]) into an LMI with a positive semidefinite solution $P \geq 0$. Consequently, the frequency domain IQC theorem can be written as an equivalent dissipation inequality test. The derived IQCs are called *hard* IQCs, i.e. they have to be fulfilled for every finite time horizon. This in contrast to so-called *soft* IQCs, which only hold over infinite time horizons.

Based on the results in [145] and [148] a worst-case gain analysis for uncertain (gridded) LPV systems was proposed, covering their robust stability and performance. It provides an upper bound on the worst-case gain based on a dissipation inequality. The upper bound condition can be stated as a semidefinite program, which can be efficiently solved via convex optimization tools, e.g. [149]. It is feasible to use this approach to cover parameter varying IQCs [150]. Its feasibility was also demonstrated for different types of perturbations, such as time delays in [150] and sector nonlinearities in [7].

Advanced Methods for Linear Time-Varying System Analysis

Based on the results in [148] an equivalent analysis condition based on LMIs and an equivalent RDE was proposed for uncertain finite horizon LTV systems in [151]. The LMI and RDE conditions are solved iteratively to mitigate the gridding's effect and calculate a less conservative upper bound. However, the initial computational approach given in [151] is flawed and was recently revised in [152].

Nevertheless, this theoretic approach is more promising for the considered problems ahead than the robust analysis approaches presented so far for finite horizon LTV robustness analyses. The work in [153] proposes robustness measures for finite time trajectories using IQCs to represent the uncertainties. It is solved by directly integrating over the IQC parameter. However, this approach does not consider disturbances. Another approach solely based on RDEs concerning IQCs and a finite horizon time-varying minimal quadratic cost control problem was proposed in [154]. In this reference, the problem is limited to a single averaging IQC used to represent the uncertainties for which a computational approach via direct one parameter search is suggested. Covering multiple IQCs was deemed infeasible by the authors at this point. This situation has changed over the last 20 years due to ever-growing computational power and the emergence of elaborate nonlinear programs. Consequently, a direct optimization approach appears attractive again. A review of a class of feasible nonlinear programs is provided in section 1.2.2.

Furthermore, the analysis of finite horizon LTV systems via gap metrics is covered in [155]. An approach based on the small gain theorem and the strict BRL in [127] is proposed in [156] to calculate time-varying robustness margins. For completeness, it shall be mentioned that, in the literature, numerous approaches to calculate the robustness of uncertain periodic LTV systems are given, such as [157–160]. More recently, the IQC framework has been extended to this problem in [161].

1.2.2 Meta-Heuristics

Meta-heuristics (MHs) have gained increasing popularity solving global optimization problems for various engineering applications. They are also known as evolutionary algorithms, as proposed in 1963 by Lawrence Fogel [162]. The majority of these methods are developed by imitating selection processes in nature such as genetic evolution [163], laws of physics e.g. Coulomb law [164] or gravitational law [165]. Other versions imitate animal behavior such as the path finding of insects, e.g. moths [166], dragonflies [167], or fruit flies [168], animal's hunting patterns e.g. wolves [169], or ant-lions [170], etc. They are considered global optimization methods for using population-based concepts and randomization in their search procedure. This allows them to recover from local optima. Also, no derivatives are required in the optimization procedure. Hence, MHs are straightforward to deploy and can be applied to almost every form of cost function and design variables. Their main downsides are a lack of search consistency and low convergence rates. Concerning these problems, significant effort was put into developing and enhancing MHs over the last two decades, focusing primarily on problem exploitation and exploration capabilities. This lead to an increasing success and prominence of MHs reported in literature. There are now examples covering a variety of engineering problems, e.g. truss sizing [171], general constraint mechanical design optimization [172–174], multi-objective design optimization of e.g. car floor-frames [175], manufacturing optimization [176], tuning of proportional-integral-derivative (PID)-type fuzzy logic controllers [177], neural networks [178], path planing of unmanned aerial systems (UASs) [179, 180] or PID tuning [181].

Focusing on robust control and/or aerospace applications, several different MHs schemes were implemented. This includes genetic algorithms for robust finite horizon controller design for uncertain flexible systems [182] and H_∞

controllers for power transfer systems [183]. Differential evolution was applied for flight control system (FCS) clearance [184] and optimal flight control [185]. Particle swarm optimization was utilized for autoland controller tuning [186] and pitch control design [187]. Furthermore, Lévy flight-based pigeon-inspired optimization was applied for automated carrier landings [188] and unmanned aerial system controller tuning [189].

The manifoldness of these problems and approaches make a promising candidate for the direct optimization problem introduced in Section 1.2.1 in the context of IQCs and uncertain LTV systems. This is a nonlinear problem that also must be assumed non-convex and non-smooth with an arbitrarily large search space. The problem size also scales with the number of covered perturbations. However, no literature concerning this matter exists.

1.2.3 Analysis of Systems with Time-Varying Dynamics

This section presents the state-of-the-art approaches to address the industrial analysis problems covered in this thesis. It starts with the finite horizon analysis of aircraft with potentially time-varying dynamics. Here, especially the finite horizon of the analysis problem, including its terminal conditions, is critical. Furthermore, the dynamics are time-varying, mostly late in the approach due to the ground effect. Important certification methods are included for completeness. Afterwards, state-of-the-art analysis methods for the atmospheric flight phase of space launchers are investigated. Here, the varying time dynamics are more significant, and the problem's focus is mainly concerned with their influence on possible violations of certification criteria along the trajectory.

Finite Horizon Analysis of Aircraft

The state-of-the-art approach, to evaluate the touchdown constraints and flight control system clearance in general, are Monte Carlo analyses, e.g. [4, 5, 190, 191], or worst case searches/ optimizations, e.g. [192–194]. These are conducted on the nonlinear model. Due to the large parameter space of the aircraft, environmental conditions, and disturbances, these approaches are computationally expensive. Hence, they are not suitable to provide fast feedback in the design/tuning phase of an autoland controller. Furthermore, they can not provide (guaranteed) worst cases, whose identification is also critical in the design process.

An example calculating linear worst cases applying μ analysis in the design process of an autoland controller can be found in [195]. However, the presented robust performance analysis is limited to frozen grid points in time over finite horizons. Thus, it is unsuitable for evaluating touchdown constraints in any form.

LTI methods with the theoretical potential to analyze touchdown constraints are reachable set analyses. However, these would fail to respect the time-varying dynamics explicitly. General examples for said analysis concerning LTI systems under uncertainty can be found in [196–198]. In [199], the approach in [196] is extended to uncertain linear systems with time varying parameters. For neither the time-invariant nor the time-variant case, applications to the final approach problem can not be found in the literature. Furthermore, they do not present worst-case analyses and are limited to a predefined set of uncertain parameters/system matrices. Additionally, the respected time horizons in the example applications are too short for the problem at hand.

A finite time horizon requirement also renders LPV IQC approaches to cover the uncertain, varying dynamics infeasible. These are a valuable tool for constraint analyses regarding gust loads, e.g. [7, 200]. These analyses cover infinite horizons and determine the worst-case value along all possible trajectories in the parameter set. This also allows the system to remain at a certain point, which is in apparent contradiction to the problem posed by an approach analysis.

Consequently, only a robust finite horizon LTV analysis allows for both the finite horizon and time-varying dynamics along a specific trajectory. So far, there exist no examples for this kind of analysis in literature. Most closely related is the approach in [201]. It presents a backward reachability analysis using IQCs for uncertain nonlinear systems using a sum of squares (SoS) approach. However, SoS approaches scale poorly with the system size. The presented analysis of a simple quad-rotor example with six states and one uncertainty over a time horizon of 2s required between 18min and 10h to complete.

Worst-Case Analysis of Launch Vehicle

The general industrial approaches used to evaluate the effects of turbulence and the estimated perturbation set utilize Monte Carlo analyses and worst-case optimizations conducted on nonlinear launcher models [1–3]. Although these methods can be directly deployed on the high fidelity nonlinear launcher model, they require significant computational resources and time. Even more critically, they can only provide a lower bound on any worst-case performance measure, such as aerodynamic loads, pitch/yaw tracking, or deviation from the flight path.

Linear worst-case analyses are conducted to provide strict upper bounds. These are mainly based on the structured singular value μ and corresponding LTI worst-case gains. Therefore the systems are analyzed at frozen gird points, i.e. all matrix coefficients are frozen at a certain time, and the systems are treated like an LTI system. Consequently, the linear worst-case analysis considers the parameter variation as ”slowly enough” and infinite time horizons. Examples for this approach can be found in, e.g. [202–206]. However, there are also counterexamples demonstrating that the LTI assumption is invalid [207]. This includes examples for LTV systems with unstable poles whose system responses are not unstable.

More recently, LPV synthesis methods were applied to launch vehicles in [208] using the non-gravitational velocity as a varying parameter with strictly positive parameter variation rates. As for the approach problem, this approach suffers from an infinite time horizon. Additionally, the launcher closely follows a predefined trajectory, rendering analyzed parameter combinations infeasible/over-conservative.

Consequently, there is no guarantee that the common linear worst-case methods provide correct nor meaningful results as they ignore the actual system's behavior due to their infinite horizon definition. The motivating example in Chapter 2 visualizes the LTI approach's insufficiency for clearly time-varying systems with finite horizons.

1.3 Thesis Aims and Objectives

This thesis aims to develop linear analytical worst-case analysis tools and the required theory to provide fast and reliable upper bounds for (aerospace) systems with time-varying dynamics. These tools need to explicitly respect the system's time-varying dynamics along a predefined finite trajectory under uncertainty. These include parametric uncertainty, dynamic uncertainty, and nonlinearities (e.g. time delays). Furthermore, the influence of external disturbances, such as wind, must be respected and accurately modeled.

Guaranteed analysis results shall be provided by extending the finite horizon LTV bounded real lemma to integral quadratic constraints (IQC). Based on the finite horizon LTV framework for IQCs, the worst-case analysis shall cover time-varying dynamics under perturbations and external disturbance and present an alternative to state-of-the-art nonlinear analyses. Due to their finite horizon nature, these tools also provide a practical approach to evaluate performance criteria worst-cases for certain points of specific trajectories.

To achieve these aims, the following objectives are specified:

1. Identify and evaluate existing nominal LTV worst-case analysis methods.

2. Extend the theoretical framework for nominal LTV worst-case analysis to IQCs to incorporate a multitude of perturbations into the analysis.

3. Develop an efficient analysis framework, including fast algorithms to calculate the worst-case gains of uncertain LTV systems, providing reliable upper bounds for the real system.

4. Evaluate the developed algorithms on simple but industry-relevant benchmark models against exiting algorithms.

5. Apply the developed LTV worst-case analysis framework on elaborate industry-sized problems. Use this to demonstrate the feasibility of the approach to provide fast upper bounds via comparison to the results of corresponding nonlinear Monte Carlo simulations.

6. Identify shortcomings of the developed tools and methods to determine future areas of research.

Note that this thesis is partially funded by ESA. Hence, the example applications will focus mainly on space launchers. These are predestined for finite horizon LTV analyses and whose mission success is connected to the greatest monetary and hazard risk. However, an aircraft touchdown analysis is used to demonstrate the versatility of the developed framework for other aerospace applications.

1.4 Thesis Outline

The thesis is divided into seven chapters corresponding to the key areas of the conducted work. They are summarized as:

Chapter 1: Introduction

The motivation for the research is presented. This includes two explicit industrial applications with growing markets, which would benefit from the development of novel analysis tools for their respective V&V processes. Afterwards, existing literature on the robust performance and worst-case analysis of linear time-varying systems is examined. This includes a digression to the historic key developments in (robust) stability, which form the backbone of the theory and methods developed in this thesis. Additionally, state-of-the-art worst-case analysis methods for systems with strictly time-varying dynamics are investigated. Here, the focus is strictly on the two industrial examples covered in the thesis. Based on the motivation and literature review, the aims and objectives of the work are stated.

Chapter 2: Motivating Example

A state-of-the-art LTI worst-case analysis of space launcher, a prototypical time-varying system, under wind disturbance is presented. The results are evaluated against a corresponding nonlinear simulation to validate the inadequacy of existing worst-case approaches for highly time-varying systems. This illustrates motivation for the thesis and emphasizes the necessity to develop dedicated LTV analysis tools.

Chapter 3: Fundamentals on Nominal Robustness Analysis of Linear Time-Varying Systems

The literature review identifies existing (worst-case) analysis conditions and methods for finite horizon LTV systems. These build the origin for the research and new developments in this thesis. Thus, the necessary theoretical background, including relevant signal and system norms, existing theorems, and integral quadratic constraints, is provided. This is essential for the extension of the finite horizon LTV framework in this thesis.

Chapter 4: Worst-Case Analysis of Uncertain Finite Horizon Linear Time-Varying Systems

The extension of the LTV bounded real lemma to IQCs is derived and proved. Subsequently, the necessary steps to convert the theorem into a computationally feasible problem are shown. These are based on the solvability of a parameterized Riccati differential equation. This leads to a constrained nonlinear optimization problem for which two tailored nonlinear programs using meta-heuristics are developed. The novel nonlinear programs are evaluated on benchmark examples to show their feasibility and applicability to industry-sized engineering problems.

Chapter 5: Finite Time Horizon Analysis of an Autolanded Aircraft in Landing Configuration under Crosswind

The developed LTV worst-case analysis framework is applied to identify worst-case touchdown conditions of an autolanded airliner. This presents a critical application for linear worst-case analysis where its finite time horizon is especially critical. It is also of high industrial relevance due to the vast amount of automated landings in transport aviation and the tight touchdown constraints to be met by the autolanding systems under a broad range of environmental and aircraft conditions. A special wind disturbance model for LTV analyses is developed, covering the wind disturbance usually applied in the certification process. It is demonstrated that the developed LTV worst-case analysis framework provides feasible upper bounds for the Monte Carlo simulation in a fraction of time. Therefore, a supplemental tool for the V&V process of autolanding systems is provided.

Chapter 6: Finite Time Horizon Analysis of a Launch Vehicle in Atmospheric Ascent

The last application presents the worst-case aerodynamic loads and lateral drift analysis of a launch vehicle's first stage flight under atmospheric disturbance. Due to the fast progression through the multiple layers of the atmosphere and rapid fuel burn, LTV worst-case analyses are of high interest to support the V&V process and narrow the validation gap between existing linear analytical methods and common nonlinear analysis. Therefore, tailored LTV wind filters are calculated, covering the wind profile of an actual launcher mission in the pitch plane and a certification wind profile in the yaw plane. The LTV worst-case analysis results are validated against a Monte Carlos simulation conducted on the nonlinear launcher model.

1.5 Summary

This chapter provided the motivation for this thesis, namely the development of fast and reliable worst-case analysis tools for highly time-varying systems under perturbation and external disturbance. Based on this motivation, an extensive literature review was presented focusing on theoretical advances and the application of worst-case analysis methods for linear systems. The latter focused on the aerospace examples covered in this thesis. The literature review pointed out the missing theory and, thus, the missing use of LTV analyses tool for highly time-varying systems under uncertainty. Thus, it emphasized the necessity to develop novel analysis tools for such systems. Afterwards, the aims and objectives of this thesis were derived from the motivation and literature review.

Chapter 2

Motivating Example

2.1 Introduction

This chapter presents an LTI analysis as typically used in industrial validation and verification processes of highly time-varying systems. A simple but sufficiently complex nominal linear worst-case performance analysis of a space launcher under wind disturbance is chosen. Firstly, the dynamics of a space launcher along a predefined trajectory are derived. Afterwards, two controllers are designed to minimize the vertical drift from the trajectory and stabilize the pitch motion. The first controller stabilizes the inherently unstable launcher dynamics over the whole trajectory. Inversely, the second controller's design leads to unstable closed-loop dynamics for the drift motion but achieves a sizable performance gain. Subsequently, a nominal LTI worst-case analysis is conducted on frozen grid points covering the trajectory. The analysis aims to predict the worst-cases of multiple performance measures relevant in the control design and V&V process of space launchers. Thus, fast and reliable feedback on the effects of design changes is crucial for the control engineer. To demonstrate the insufficiency of the LTI approach, a corresponding Monte Carlo simulation is conducted on the nonlinear model. Hence, this chapter provides the practical motivation of the thesis.

2.2 Exemplary Launcher Model

The analysis is conducted on a representative expendable launch vehicle (ELV) model during atmospheric flight. It covers a time horizon from $t_\mathrm{s} = 25\mathrm{s}$ to $t_\mathrm{f} = 95\mathrm{s}$ after lift-off, including the most critical flight segments, such as the transonic region, the region of maximum aerodynamic pressure, and the engine burn-out. During this flight phase, the launcher tracks a pre-calculated pitch program, which is designed to minimize the aerodynamical loads on the launcher.

The model's complexity matches the general recommendations for the initial control design process of launch vehicles [209, 210]. In this example, only the rigid body's pitch dynamics are considered with neglected effects of propellant sloshing. The thrust vectoring control's inertias are also neglected. Furthermore, following common practice, the spheric and rotating earth's influence is ignored [209–211]. Additionally, only linear aerodynamics are considered.

A detailed description of the nonlinear equations of motion and how to derive the linear equations with respect to a reference trajectory frame is given in Appendix A. Consequently, the linear equations of motion are stationary with respect to the trajectory reference frame. This does not solve the problem that the parameters are only valid for the discrete analysis points and corresponding instants in time along the trajectory.

2.2.1 Launcher Augmentation

The analyzed space launcher is aerodynamically unstable, and feedback control is required to stabilize the launcher and track the pitch program. A proportional-derivative (PD) control law will be used in the $\Delta\theta$ channel, while solely proportional feedback will be applied in the $\Delta\alpha$ channel:

$$\delta_{\mathrm{TVC,cmd}} = -K_{\mathrm{TVC}}(K_{\dot\theta}\Delta\dot\theta + K_\theta\Delta\theta + K_\alpha\Delta\alpha), \tag{2.1}$$

Table 2.1: Controller gains used for the analysis

Controller	K_{TVC}	$K_{\dot{\theta}}$	K_{θ}	K_{α}
Stable	2	0.4	1	0.2
Unstable	2	0.4	1	1.6

where K_{TVC} is the TVC servo-amplifier, $K_{\dot{\theta}}$ the rate gyro gain, K_{θ} the orientation gyro gain, and K_{α} the angle of attack sensor gain. The controller is specifically chosen as it minimizes the vertical trajectory drift under wind disturbance. Two sets of gains, as summarized in Tab. 2.1, are calculated as detailed in [212, Chapter 3]. The first set corresponds to stable closed-loop dynamics over the whole trajectory. Conversely, the second set belongs to a closed-loop with an unstable flight path pole (vertical drift) over the whole analysis segment. This promises significantly better drift performance as the launcher actively steers into the wind.

2.3 Worst Case Performance Metric

The worst-case energy-to-peak gain , e.g. [213] defined as:

$$\|G\|_{2\to\infty} = \sup_{\substack{d\in L_2(\infty,\infty)\\ d\neq 0, x(0)=0}} \frac{\|e(t)\|_{\infty}}{\|d(t)\|_{2(-\infty,\infty)}}. \tag{2.2}$$

is used to calculate the LTI worst-case performance at frozen points in time. In (2.2), $d(t)$ and $e(t)$ denote the disturbance input and performance output signal, respectively. The notations $\|...\|_{\infty}$ and $\|...\|_{2(-\infty,\infty)}$ represent the infinite horizon ∞-norm and infinite horizon 2-norm, respectively. A detailed definition follows in Section 3.2. It provides a guaranteed upper bound of the maximum peaks of the systems (performance) outputs for an arbitrary norm bounded input. In the case of $\|d(t)\|_{2(-\infty,\infty)} = 1$ (unit norm) and single performance output, (2.2) directly provides the physical worst-case value. It can be readily calculated based on the solution of the Lyapunov equation:

$$AQ + QA^T + B^TB = 0, \tag{2.3}$$

with $A \in \mathbb{R}^{n_x \times n_x}$ denoting the LTI state matrix, $B \in \mathbb{R}^{n_x \times n_d}$ the LTI input matrix, $C \in \mathbb{R}^{n_e \times n_x}$ the LTI output matrix, and $Q \in \mathbb{R}^{n_x \times n_x}$ the controllability Gramian. The variables n_x, n_d, and n_e represent the number of states, disturbance inputs and performance outputs, respectively. Based on (2.3), the worst case gain can be computed as

$$\|G\|_{2\to\infty} = \sqrt{\max(\mathrm{eig}(CQC^T))}. \tag{2.4}$$

In the course of this thesis, it is calculated using the Matlab internal function `gram` (see [214]). However, the LTI energy-to-peak gain requires the analyzed system to be stable with feedthrough matrix $D = 0$. Consequently, neither the vertical deviation, due to the corresponding pole in the origin, nor the non-stabilizing controller can be analyzed.

2.4 Wind Disturbance Model

The evaluated wind disturbance shall resemble Dryden turbulence profiles. These are frequently used aerospace certification processes [215–217].

2.4.1 Wind Filter Nonlinear Analysis

In the nonlinear analysis, the Dryden wind filter G_w for vertical turbulence

$$\begin{aligned} \dot{x}_w(t) &= \begin{bmatrix} 0 & 1 \\ -\left(\frac{V(t)}{L_w}\right)^2 & -2\frac{V(t)}{L_w} \end{bmatrix} x_w(t) + \begin{bmatrix} 0 \\ \left(\frac{V(t)}{L_w}\right)^2 \end{bmatrix} n_w(t) \\ v_w(t) &= \begin{bmatrix} \sigma(h)\sqrt{\frac{L_w}{\pi V(t)}} & \sigma(h)\frac{L_w}{V(t)}\sqrt{\frac{3L_u}{\pi V(t)}} \end{bmatrix} x_w(t) \end{aligned} \tag{2.5}$$

with white noise input n_w, is implemented to generate the wind disturbance w. Here, the white noise signal is calculated by Matlab's internal band-limited white noise block. These signals have a power spectral density Φ of one and are shaped into continuous turbulence profiles statistically matching real turbulence by the filter G_w. In (2.5), V is the launcher's velocity, σ is the altitude-dependent turbulence intensity, and L_w is the turbulence scale length. For the analysis, the values for σ are interpolated over altitude, based on the data for severe turbulence provided in [218]. Contrary to [218], the turbulence scale length is chosen to five-times the recommended value, i.e. $L_w = 2629.2\text{m}$. This results in longer turbulence gusts, which are more critical in the context of the analyzed trajectory disturbances. According to the analyzed trajectory segment, the value of L_u is not altitude dependent following [218].

2.4.2 Wind Filter Linear Analysis

The wind filter G_w is unsuitable for LTI analyses applying worst-case energy to peak gain, see e.g. [213]. This gain only defines an upper bound from an arbitrary analysis input $d(t)$ to the maximum peak of the output signal $e(t)$. Consequently, for the calculated gain to have a physically meaningful value, the wind filter must generate a valid turbulence spectrum for inputs with unit norm. Although some literature exists covering discrete gusts, e.g. [219], the turbulence analysis for LTI systems is relatively unexplored.

In this thesis, the scaling approach proposed in [220] is utilized, i.e. the wind filter is scaled with the maximal expected norm of a white noise signal over a certain analysis horizon. Here, the scaling k_{w_i} is chosen so that the norm of a white noise signal from t_s up to the analyzed frozen point in time t_i is covered.

Accordingly the wind filter $G_{w,\mathrm{LTI}}$ for the LTI analysis is defined as

$$\begin{aligned}\dot{x}_w(t) &= \begin{bmatrix} 0 & 1 \\ -\left(\frac{V(t_i)}{L_w}\right)^2 & -2\frac{V(t_i)}{L_w} \end{bmatrix} x_w(t) + \begin{bmatrix} 0 \\ \left(\frac{V(t_i)}{L_w}\right)^2 \end{bmatrix} n_w(t) \\ v_w(t) &= k_{w_i} \begin{bmatrix} \sigma(t_i)\sqrt{\frac{L_w}{\pi V(t_i)}} & \sigma(t_i)\frac{L_w}{V(t_i)}\sqrt{\frac{3L_w}{\pi V(t_i)}} \end{bmatrix} x_w(t), \end{aligned} \tag{2.6}$$

for a frozen time t_i. Note that due to the predefined trajectory the altitude dependencies in (2.5) convert to strict time dependencies in (2.5). This further highlights a significant issue with LTI analysis, the systematic modeling of appropriate input disturbances.

2.5 Analysis

A nominal LTI worst-case analysis is conducted for both controller designs applying the wind filter. The outcomes are evaluated against the results of a Monte Carlo simulation conducted on the corresponding nonlinear model.

2.5.1 Analysis Setup

The general analysis structure is shown in Fig. 2.1. In the nonlinear analysis, the ELV-block is described by the nonlinear dynamics (A.1) in Appendix A.1. Additionally, the Wind-block represents the unscaled wind filter G_w, with white noise input $n_w(t)$. Depending on the test case, block C represents the stabilizing or non-stabilizing controller with the gains provided in Tab. 2.1. Identical controllers are used in the linear and nonlinear analyses. Furthermore, the performance output y_p includes the signals $\Delta Q\alpha$ (aerodynamic

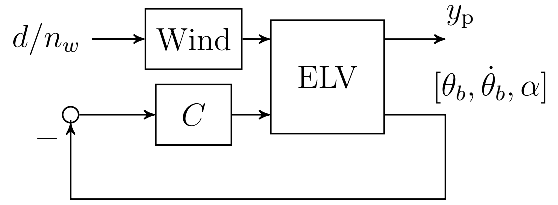


Figure 2.1: *General analysis interconnection used for nominal launcher analysis*

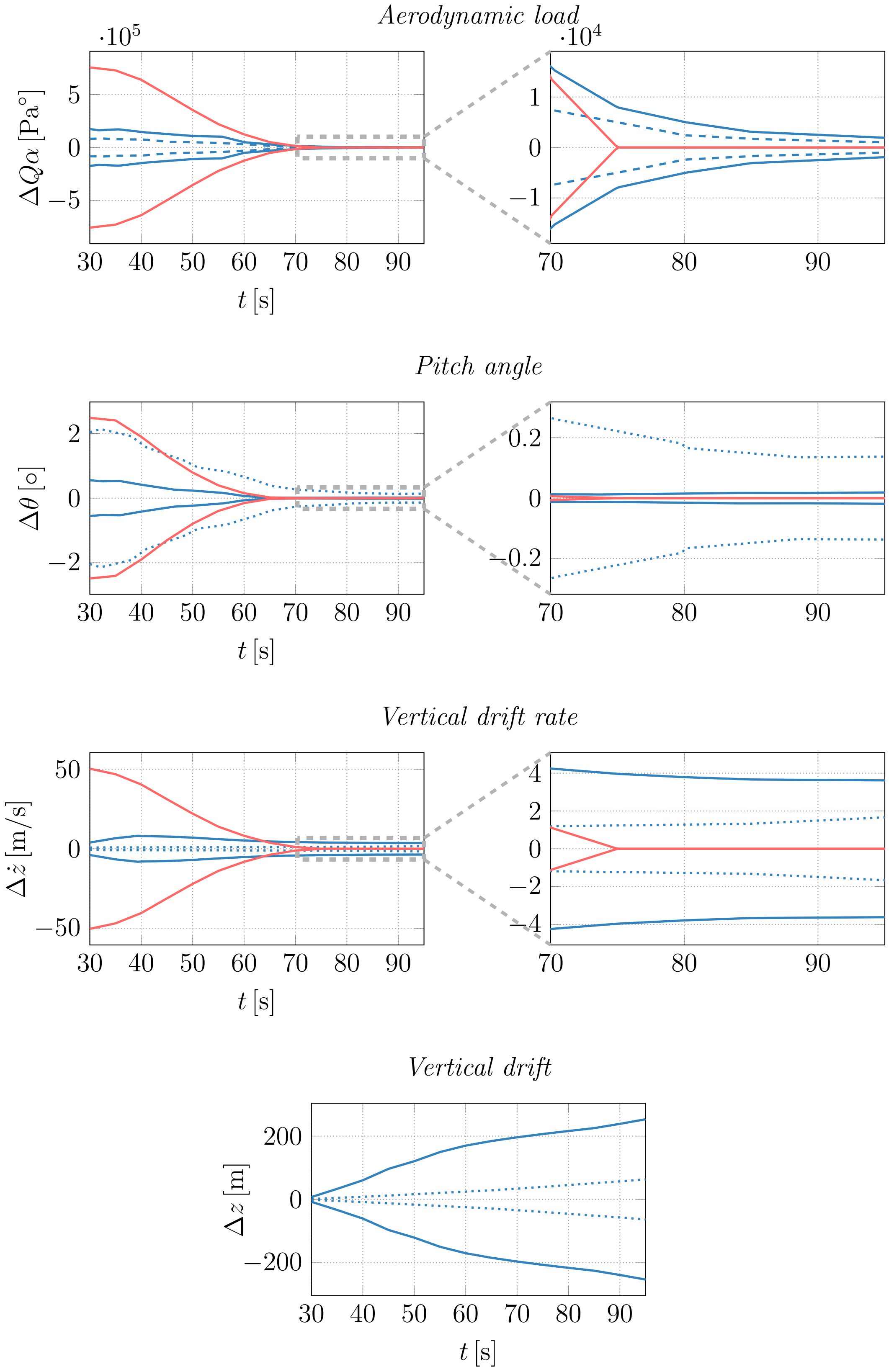
load), $\Delta\theta$ (pitch angle deviation), $\Delta\dot{z}$ (vertical drift rate), and Δz (vertical deviation form trajectory).

In case of the linear analyses, the ELV-block describes the linear dynamics in (A.16) from Appendix A.1. Furthermore, the scaled wind filter $G_{w,\mathrm{LTI}}$ is applied with input d.

2.5.2 Results

Using the frozen time approach, e.g. [204], the worst-case gain is calculated at points in time t_i on the interval [30s, 95s] with a step size of 5s. Here, all performance signals are evaluated individually. As the starting point of the analysis, 30s is chosen so that a suitable wind filter scaling can be calculated. The Monte Carlo simulation of the nonlinear model is conducted in Matlab Simulink using the corresponding analysis interconnection in Fig. 2.1. The simulation starts at $t_\mathrm{s} = 25$s and ends at $t_\mathrm{f} = 95$s after lift-off. 5000 unique white noise signals $n_{w_i}(t)$ are evaluated.

The result of the LTI frozen grid analyses as well as the bound enveloping all Monte Carlo signals for both controllers, are shown in Fig. 2.2. For the LTI analyses, the points in time in-between the analysis points are linearly interpolated. Starting with the stabilizing controller, it can be seen that the LTI analysis is initially more conservative during the initial part of the trajectory but fails to provide an upper bound for times after approximately 70s. This is emphasized with detail windows on the right side of Fig. 2.2. As mentioned before, no results for the worst-case deviation can be calculated. The unsatisfactory results of the LTI analysis, particularly in the later part, expose a major limitation of the frozen grid point approach. Due to the turbulence's altitude dependence, the maximal wind disturbances at later frozen grid points are significantly decreasing. With the analysis limited to the assumption that the system remains on the frozen grid point, also only the worst-case disturbance for dynamics at precisely this point can be modeled. Thus, the influence



Aerodynamic load

Pitch angle

Vertical drift rate

Vertical drift

Figure 2.2: *Results for bounds on performance metrics: Upper bound LTI worst-case analysis (—) $K_\alpha = 0.1$, bound on most critical Monte Carlo $K_\alpha = 0.1$ (—), bound on most critical Monte Carlo $K_\alpha = 1.6$ (- - -)*

of critical points along the trajectory, such as the transonic or engine burn-out, cannot affect later points in time and need to be evaluated explicitly. However, even this analysis will have little meaning as, in reality, the system only passes through that point for an infinitesimal amount of time.

As mentioned before, the vertical deviation from the trajectory cannot be evaluated using the LTI energy to peak gain. This is a significant drawback of the LTI approach as it is not possible to calculate the worst-case value for a central design criterion of the controller. Altogether, the LTI approach fails to analyze realistic system behavior and fails to provide a reliable/guaranteed upper bound. The respective Monte Carlo simulation finished after 1h45min, whereas the LTI analysis was completed in a total of 15s. Thus, the latter would provide a significant time advantage in the V&V process if it provides a reliable upper bound.

Comparing the bounds provided for both controllers by the Monte Carlo simulation in Fig. 2.2, shows a noticeable reduction of the maximal occurring drift rate and total deviation from the trajectory in the nonlinear analysis. The former reduces from 8.17m/s to 1.66m/s as K_α increases from 0.1 to 1.6 and the latter from 253.22m to 63.32m. As the launcher actively steers into the wind, the absolute value of $\Delta\theta$ increases in the Monte Carlo simulation from 0.55° to 2.14° compared to the stabilizing controller. Due to the reduced drift, the gravity turn is better executed, and the maximum load reduces from $1.74 \cdot 10^5$Pa° to $8.48 \cdot 10^4$Pa°.

Consequently, exploiting unstable closed-loop dynamics results in a significant performance gain. However, the state-of-the-art linear analysis approach cannot provide this crucial feedback in the design process. Thus, the control engineer would need to rely fully on Monte Carlo simulations and thus only lower bounds on the performance measures without identifying guaranteed worst cases. Furthermore, the Monte Carlo simulations require a combined 3h30min for a relatively small sample size and do not provide fast feedback in an iterative tuning process.

2.6 Summary

It was shown that LTI worst-case analyses are not suitable for systems with rapidly changing dynamics along a predefined finite trajectory. This is mainly due to the analysis's limitation to frozen points in time, which entirely denies the trajectory characteristics and physical system behavior. The LTI approach was infeasible to provide meaningful insights during the control design process showing a non-comprehensible gap to the nonlinear analysis. Due to the LTI approach's limitation to stable systems, for the non-stabilizing controller, no worst-case gain could be calculated. Here, the nonlinear analysis showed a clear performance improvement. However, already for this simple example, the simulation-based approach required significantly more time than the linear analysis. This example underlines the imperative for developing and applying linear methods explicitly respecting the analyzed system's time-varying characteristics.

Chapter 3

Fundamentals on Nominal Robustness Analysis of Linear Time-Varying Systems

3.1 Introduction

This chapter presents the necessary theoretical preliminaries on the derivation of linearized representations of nonlinear systems explicitly respecting the time variance. Furthermore, the required signal and system norms, nominal worst-case analysis conditions, and IQC theory necessary to extend the LTV analysis framework to perturbed systems are introduced.

3.2 Finite Horizon Linear Time-Varying Systems

As automation becomes more and more prominent in various systems' applications, a significant subset of these systems follows a preprogrammed trajectory leading the system from a fixed starting point to a fixed terminal point. A typical example of trajectory-based operations is a space launcher during atmospheric ascent. The launcher has to tightly follow a predefined trajectory starting from the lift-off and ending with the first stage's burn-out. Consequently, its nonlinear equations of motion are strictly time-dependent.

A less obvious example is the final approach of an auto-landed aircraft. In this scenario, the aircraft has to precisely and fully autonomously track the runway's instrument landing systems' guidance signal. This leads to a specific finite reference trajectory commencing with the aircraft's touchdown. Here, the terminal conditions of the trajectory, especially under adverse environmental conditions, are essential to assure safe landings.

Another example is an industrial robot. These commonly are used for automated assembly [221], materials and quality testing [222], or manufacturing [223, 224]. A robot arm, as shown in Fig. 3.1 is used as an example to show the linearization of nonlinear dynamics along a predefined trajectory. Its planar nonlinear dynamics in the xy-plane concerning the two main links are

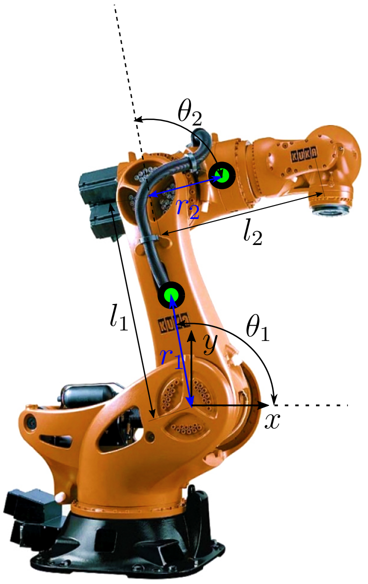


Figure 3.1: *Industry robot (Source: KUKA)*

described by [107]:

$$\begin{bmatrix}\tau_1 \\ \tau_2\end{bmatrix} = \begin{bmatrix}\alpha + \beta\cos(\theta_2) & \delta + \beta\cos(\theta_2) \\ \delta + \beta\cos(\theta_2) & \delta\end{bmatrix}\begin{bmatrix}\ddot{\theta}_1 \\ \ddot{\theta}_2\end{bmatrix} + \begin{bmatrix}-\beta\sin(\theta_2)\dot{\theta}_2 & -\beta\sin(\theta_2)(\dot{\theta}_1+\dot{\theta}_2) \\ \beta\sin(\theta_2)\dot{\theta}_1 & 0\end{bmatrix}\begin{bmatrix}\dot{\theta}_1 \\ \dot{\theta}_2\end{bmatrix}. \tag{3.1}$$

In (3.1), τ_i is the torque applied to the base of the i-th link. Hence, the input vector is chosen as $d = [\tau_1, \tau_2]$. The robot arm's states are represented by $x = [\theta_1, \dot{\theta}_1, \theta_2, \dot{\theta}_2]$ and the outputs by $e(t) = [\theta_1, \theta_2]$. The angles are denoted with respect to joint fixed Cartesian coordinate systems. The parameters α, β, and δ describe substitutes variables combing the mass moments of inertia as follows:

$$\begin{aligned}\alpha &:= I_1 + I_2 + m_1 r_1^2 + m_2(l_1^2 + r_2^2) \\ \beta &:= m_2 l_1 r_2 \\ \delta &:= I_2 + m_2 r_2^2,\end{aligned} \tag{3.2}$$

where l_i and r_i are the total length, and the distance from the joint to it's center of mass, respectively, of the i-th arm. The mass and mass moment of inertia of the i-th arm is denoted by m_i and I_i, respectively. Given a desired state trajectory, $x_0(t)$ the input torque $d(t)$ required can be calculated to achieve the prescribed movement. The robot arm's nonlinear equation of motion in (3.1) along this reference trajectory can be written more generally in the form of the differential equations:

$$\begin{aligned}\dot{x} &= f(x, t, d) \\ e &= g(x, t, d)\end{aligned} \tag{3.3}$$

Under the assumption of small perturbations, the robot arm's movement along this reference trajectory can be modeled as a superposition of a known and perturbed motion. Consequently, the state vector $x(t)$ of the robot arm can be written as nominal vector $x_0 t$ and a perturbation vector $\delta x(t)$:

$$x(t) = x_0(t) + \delta x(t). \tag{3.4}$$

In the same fashion the input vector d can be split into $d(t) = d_0(t) + \delta d(t)$. The subsequent multidimensional Taylor expansion of the state and input vector in (3.3) with respect to the reference trajectory defined by $[x_0, d_0]$ results in the linear approximation:

$$f(x,t,d) = f(x_0,t,d_0) + \nabla_x^T f(x_0,t,d_0)\delta x + \nabla_d^T f(x_0,t,d_0)\delta d + \dots. \tag{3.5}$$

Neglecting higher order terms, the linear perturbation dynamics can be written as:

$$\delta f(t) = \sum_{i=1}^{n} \left(\frac{\partial f(t)}{\partial x_i(t)} \right)_0 \delta x_i(t) + \sum_{i=1}^{n} \left(\frac{\partial f(t)}{\partial d_i(t)} \right)_0 \delta d_i(t), \tag{3.6}$$

where the subscript 0 indicates the differentials evaluated on the reference trajectory. Similarly, a linear representation of the robot arm's output equation g can be derived. The linear dynamics of the robot arm can be arranged in typical state-space form and written as:

$$\begin{aligned} \dot{x}_{G_t}(t) &= A_{G_t}(t) x_{G_t}(t) + B_{G_t}(t) d(t) \\ e(t) &= C_{G_t}(t) x_G(t) + D_{G_t}(t) d(t), \end{aligned} \tag{3.7}$$

where $x_{G_t}(t) \in \mathbb{R}^{n_{xG_t}}$, $d(t) \in \mathbb{R}^{n_d}$ and $e(t) \in \mathbb{R}^{n_e}$ are the state, input, and output vectors, respectively. The system G_t in (3.7) is a finite horizon linear time varying system. Its system matrices are piecewise continuous bounded functions of time t with compatible size to the corresponding vectors, i.e. $A_{G_t} : [0,T] \to \mathbb{R}^{n_{xG} \times n_{xG_t}}$, $B_{G_t} : [0,T] \to \mathbb{R}^{n_{xG_t} \times n_d}$, $C_{G_t} : [0,T] \to \mathbb{R}^{n_e \times n_{xG_t}}$ and $D_{G_t} : [0,T] \to \mathbb{R}^{n_w \times n_d}$.

3.3 Signal Norms

The linear worst-case analyses in this thesis are conducted exclusively in the time domain. A time domain signal is a (Lebesgue) measurable function that maps the real numbers $\mathbb{R}$ to the real vector $\mathbb{R}^n$ and forms the set (see e.g. [127, Chapter 3]):

$$\mathcal{S} := \{d : \mathbb{R} \to \mathbb{R}^n\}. \tag{3.8}$$

This forms a natural vector space under addition and scalar multiplication. Two subspaces can be defined from the set $\mathcal{S}$:

$$\mathcal{S}_+ := \{d \in \mathcal{S} : d(t) = 0 \,\forall\, t < 0\} \tag{3.9}$$

and

$$\mathcal{S}_- := \{d \in \mathcal{S} : d(t) = 0 \,\forall\, t > 0\}. \tag{3.10}$$

In the proceeding, $\|x\| = \sqrt{x^T x}$ defines the common Euclidean vector norm.

3.3.1 Infinite Horizon 2-Norm

The infinite horizon 2-norm is defined as (e.g. [21]):

$$\|d(t)\|_{2(-\infty,\infty)} = \left\{ \int_{-\infty}^{\infty} \|d(t)\|^2 \right\}^{\frac{1}{2}}. \tag{3.11}$$

It is commonly used in the stability and performance analysis of LTI and LPV systems, where the signal's behavior over infinite horizons must be considered. Signals which remain bounded over infinite time horizons are in the infinite Lebesgue 2-space defined as:

$$L_2(-\infty,\infty) = \{d(t) \in \mathcal{S} : \|d(t)\|_{2(-\infty,\infty)}\} \tag{3.12}$$

Based on $L_2(-\infty,\infty)$, the two vector spaces $L_2[0,\infty)$ and $L_2(-\infty,0]$ can be defined/built using the intersections of the two sets $\mathcal{S}_+$ and $L_2(-\infty,\infty)$ and $\mathcal{S}_-$

and $L_2(-\infty,\infty)$, respectively. Proving that a signal is in $L_2[0,\infty)$, is usually done by the stepwise evaluation of increasingly stringent growth conditions, e.g. [127, Chapter 3]. In this context, the extended 2-space is introduced as:

$$L_{2_e} = \{d(t) \in L_2[0,T] \;\forall T < \infty\}. \tag{3.13}$$

Nevertheless, $d(t) \in L_{2e}$ does not imply that $\sup_T \|d(t)\|_{2[0,\infty)} < \infty$, as e.g., $d(t) = 2t^2$ and $d(t) = e^{2t}$ are both in L_{2e}, but not in $L_2[0,\infty)$. In (3.13), $L_2[0,T]$ is the finite horizon Lebesgue 2-space.

3.3.2 Finite Horizon 2-Norm

The finite horizon 2-norm, which will be used for the performance analysis of finite horizon LTV systems in this thesis, is defined as:

$$\|d(t)\|_{2[0,T]} = \left\{ \int_0^T \|d(t)\|^2 \right\}^{\frac{1}{2}}. \tag{3.14}$$

Signals whose finite horizon 2-norm is limited are in the finite horizon Lebesgue 2-space defined by the set:

$$L_2[0,T] = \{d(t) \in \mathcal{S}_+ : \|d(t)\|_{2[0,T]} < \infty\} \tag{3.15}$$

Note that all signals which are continuous on the time horizon $[0,T]$ are bounded and thus in $L_2[0,T]$. Consequently, signals of the form $d(t) = \frac{2}{3t-T}$ are not in $L_2[0,T]$.

3.3.3 Infinite Horizon ∞-norm

The L_∞-norm of time-domain signal $d(t) \in \mathbb{R}^{n_d}$ over an infinite horizon is defined as (e.g. [21]):

$$\|d(t)\|_\infty = \sup_t \left(\max_n |d_n(t)| \right). \tag{3.16}$$

Thus, it describes the signal's peak value over time. In case $d(t) \in L_2(-\infty, \infty)$, it is implied that $\|d(t)\|_\infty < \infty$. The L_∞ norm will be used in the context of nominal LTI worst-case analyses in this thesis.

3.4 System Norm

In the course of the thesis various performance metrics are evaluated using linear (analytical) analysis methods. This section provides the necessary background on the underlying system norms/gain to quantify the performance.

3.4.1 Finite Horizon Induced $L_{2[0,T]}$ Gain

When nominal worst-case tracking performances for LTV systems are evaluated, the finite horizon induced L_2 gain (e.g. [127]):

$$\|G\|_{2[0,T]} = \sup_{\substack{d \in L_2[0,T] \\ d \neq 0, x(0)=0}} \frac{\|e(t)\|_{2[0,T]}}{\|d(t)\|_{2[0,T]}}. \tag{3.17}$$

is utilized in this thesis. It can be interpreted as the maximum energy amplification from the (disturbance) input to the (performance) output along the finite trajectory over all valid signals in $L_2[0,T]$. An efficient way for its calculation is presented in the subsequent section. By $T \leq \infty$ and the causality of G, it is implied that if $d(t) \in L_2[0,T]$, also $x_G(t)$ and $e(t)$ are in $L_2[0,T]$. Hence, for any fixed horizon the induced $L_{2[0,T]}$ gain is finite.

3.4.2 Finite Horizon $L_{2[0,T]}$ to Euclidean Gain

For the quantification of upper bounds on worst-case values for an LTV system's performance outputs at the end of the trajectory, the finite horizon energy-to-Euclidean gain is utilized. Assuming $D(t) = 0$, it is defined as (e.g. [127]):

$$\|G\|_{E[0,T]} = \sup_{\substack{d \in L_2[0,T] \\ d \neq 0, x(0)=0}} \frac{\|e(T)\|}{\|d(t)\|_{2[0,T]}}. \tag{3.18}$$

Consequently, it describes the ball upper bounding the performance output over all valid disturbance input signals in $L_2[0,T]$ at the final point of the trajectory. The restriction $D(t) = 0$, guarantees that the gain is well-defined. Note that the gain only upper bounds the output at the final time T. For intermediate points in time $t \in [0,T]$, the $L_2[0,T]$ to Euclidean gain $\|G\|_{E[0,t]}$ can equivalently be used to upper bound the output. Given $C = I_{n_x}$, $\|G\|_{E[0,T]}$ defines an upper bound on the reachable set, i.e. the set of states that can be reached at final time by a norm bounded disturbance.

3.5 Bounded Real Lemma for Linear Time-Varying Systems

Sufficient conditions to calculate the upper bound of the finite time horizon gains in Section 3.4 are based on the well-known LTI Bounded Real Lemma. An extension to finite horizon LTV systems can be found in [127, Theorem 3.7.4], providing an upper bound on the induced $L_2[0,T]$ gain. An equivalent formulation of the theorem is given below.

Theorem 1. *Let G_t be an LTV system defined by (3.7). Given $x(0) = 0$, iff there exists a time-dependent, continuous differentiable matrix valued function $P : \mathbb{R}_0^+ \to \mathbb{S}^{n_x}$ such that*

$$P(T) = 0 \tag{3.19}$$

and

$$\begin{aligned}\dot{P} = &- PA - A^T P - C^T C \\ &- (PB + C^T D)(D^T D - \gamma^2 I_{n_d})^{-1}(B^T P + D^T C),\end{aligned} \tag{3.20}$$

then γ is an upper bound on the induced $L_2[0,T]$ gain of G_t.

Proof. The proof is based on the definition of a positive definite storage function $V(x,t) = x^T(t)P(t)x(t)$. After perturbing the RDE in (3.20) with an infinitesimal small positive scalar ϵ, the resulting Riccati differential inequality can be rearranged as an equivalent LMI applying Schur's complement (see e.g. ([225]):

$$\begin{bmatrix} \dot{P} + PA + A^T P + C^T C & PB + C^T D \\ B^T P + D^T C & D^T D - \gamma^2 (I_{n_d}(1-\epsilon)) \end{bmatrix} \leq 0. \tag{3.21}$$

Left and right multiplying (3.21) with $[x^T(t), d^T(t)]$ and $[x^T(t), d^T(t)]^T$, respectively, results in the dissipation inequality:

$$\begin{aligned} 0 \geq\, & \underbrace{x^T \dot{P} x + x^T P \overbrace{(Ax + Bd)}^{\dot{x}(t)} + \overbrace{(x^T A^T + d^T B^T)}^{\dot{x}^T(t)} Px}_{\dot{V}(t,x)} \\ & + \underbrace{x^T C^T (Cx + Dd) + d^T D^T (Cx + Dd)}_{e^T(t)e(t)} - (1-\epsilon)\gamma^2 d^T d \end{aligned} \tag{3.22}$$

Integrating (3.22) from 0 to T results in:

$$\begin{aligned} 0 \geq\, & x(T)^T P(T) x(T) - x(0) P(0) x(0) \\ & + \underbrace{\int_0^T e(t)^T e(t) dt}_{\|e(t)\|^2_{2[0,T]}} - (1-\epsilon)\gamma^2 \underbrace{\int_0^T d(t)^T d(t) dt}_{\|d(t)\|^2_{2[0,T]}}. \end{aligned} \tag{3.23}$$

Applying zero initial conditions, norm definitions, and the terminal condition (3.19), the square root of (3.23) becomes:

$$0 \geq \|e(t)\|_{2[0,T]} - \sqrt{(1-\epsilon)}\gamma \, \|d(t)\|_{2[0,T]} \tag{3.24}$$

Consequentially, the upper bound on (3.17) is given by γ. □

Theorem 1 can be easily adjusted to provide the upper bound on the finite horizon $L_2[0,T]$ to Euclidean gain:

Theorem 2. *Let G_t be an LTV system defined by (3.7). Given $x(0) = 0$, if there exists a time-dependent, continuous differentiable matrix valued function*

$P : \mathbb{R}_0^+ \rightarrow \mathbb{S}^{n_x}$ such that

$$P(T) = C(T)^T C(T) \tag{3.25}$$

and

$$\begin{aligned} \dot{P} = &- PA - A^T P \\ &- (PB + C^T)(-\gamma^2 I_{n_d})^{-1}(B^T P + C), \end{aligned} \tag{3.26}$$

then γ is an upper bound on the $L_2[0,T]$ to Euclidean gain of G_t.

Proof. Again, the proof is build on the definition of a positive definite storage function $V(x,t) = x(t)P(t)x(t)$. After perturbing it with an infinitesimal small positive integer ϵ, applying Schur's complement, and left/right multiplying the resulting LMI with $[x^T, d^T]$ and $[x^T, d^T]^T$, respectively, (3.26) can be written as:

$$\begin{aligned} 0 \geq &\underbrace{x^T \dot{P} x + x^T P \overbrace{(Ax + Bd)}^{\dot{x}(t)} + \overbrace{(x^T A^T + d^T B^T)}^{\dot{x}(t)^T} Px}_{\dot{V}(t,x)} \\ &- (1-\epsilon)\gamma^2 d^T d. \end{aligned} \tag{3.27}$$

Integrating (3.27) from 0 to T and applying zero initial conditions results in:

$$0 \geq x(T)^T P(T) x(T) - (1-\epsilon)\gamma^2 \underbrace{\int_0^T d(t)^T d(t) dt}_{\|d(t)\|^2_{2[0,T]}}. \tag{3.28}$$

The left/right multiplication with $x(T)^T/x(T)$ of the terminal condition (3.25) results in:

$$x(T)^T P(T) x(T) = \underbrace{x(T)^T C(T) x(T)}_{e(T)^T e(T)} \tag{3.29}$$

Inserting (3.29) in (3.28) leads to:

$$0 \geq \underbrace{e(T)^T e(T)}_{\|e(T)\|_2^2} - (1-\epsilon)\gamma^2 \left\| d(t) \right\|^2_{2[0,T]} \tag{3.30}$$

Applying the Euclidean vector norm, it can be concluded that γ upper bounds (3.18). □

Consequently, the upper bound γ in Theorem 1 and 2 is based on the existence of (3.20) and (3.26), respectively, over the full horizon $[0, T]$. Thus, it can be easily calculated by bisecting over γ to identify the minimal value for which (3.20) or (3.26) is fully integrable backwards in time given the provided terminal conditions.

3.6 Example of a Linear Time-Varying Worst-Case Analysis

3.6.1 Introduction

Having demonstrated the unsuitability of LTI methods for systems with highly time varying dynamics in Chapter 2, a corresponding finite horizon LTV analysis is conducted. It uses the same analysis setup, but applies the theory presented in Sections 3.4.2 and 3.5. The potential of this approach for systems like space launchers is demonstrated by comparison to the benchmark Monte Carlo simulations.

3.6.2 Analysis

The launcher model is equivalent to the one introduced in Chapter 2, as the linear analysis interconnection shown in Fig. 2.1 is essentially LTV, and finite horizon norms reason the wind filter scaling. Therefore, it can be directly used in the nominal LTV worst-case analysis. Furthermore, the same two controllers are compared.

To calculate upper bounds on the worst-case performance, the nominal finite horizon worst-case $L_{2[0,T]}$ to $\|e(T)\|_2$ gain is applied. The nominal finite horizon worst-case $L_{2[0,T]}$ to $\|e(T)\|_2$ gain can only upper bound the performance output y_p at the respective terminal time T. Therefore, it is necessary to analyze a set of terminal times covering the trajectory. The LTV analysis is performed on final times T_i in the interval [30s, 95s] with a step size of 5s using the approach explained in Section 3.5. For this grid, the wind filter scalings k_{w_i} calculated for $G_{w,\mathrm{LTI}}$ can be directly applied in the current analysis. A relative and absolute tolerance for the bisection of $\epsilon_{\mathrm{BS}_\mathrm{rel}} = 10^{-4}$ and $\epsilon_{\mathrm{BS}_\mathrm{abs}} = 10^{-6}$ is chosen. The RDE is solved using the Matlab internal `ode15s` solver for stiff ordinary differential equations (ODEs) using its default settings.

The results of the LTV finite horizon analyses and Monte Carlo upper bounds from Section 2.5.2 are compared in Fig. 3.2. For the LTV analysis, the points in time in-between the analysis grid are linearly interpolated.

It can be seen that the LTV worst-case envelope encloses all signals of the Monte Carlo simulation for times before 60s for both controllers. These upper bounds are significantly less conservative than for the LTI worst-case analysis. Focusing on the stabilizing controller and times after 60s, only in case of the lateral drift rate and lateral drift, the LTV analysis provides a distinct upper bound. However, the Monte Carlo envelopes for $\Delta Q\alpha$ and $\Delta\theta$ are almost identical to the corresponding LTV worst-cases, with sporadic violations. Regarding the non-stabilizing controller, the upper bounds only holds in case of the lateral drift rate and later drift for times after 60s. In contrast, the LTV worst-case bound is frequently violated for $\Delta Q\alpha$ and $\Delta\theta$ by the Monte Carlo results. However, these infractions happen late along the trajectory, where the overall values of the performance metrics are small and well below their limit values.

Nevertheless, it is apparent from this comparison that the LTV approach presents a more accurate representation of the actual system's behavior. As the system behavior up to the respective terminal time is analyzed, significantly improved results are achieved compared to the preceding LTI analyses. The analysis also covers the wind disturbance characteristic up to the respective final time. This also includes the effects of changing dynamics in the transonic

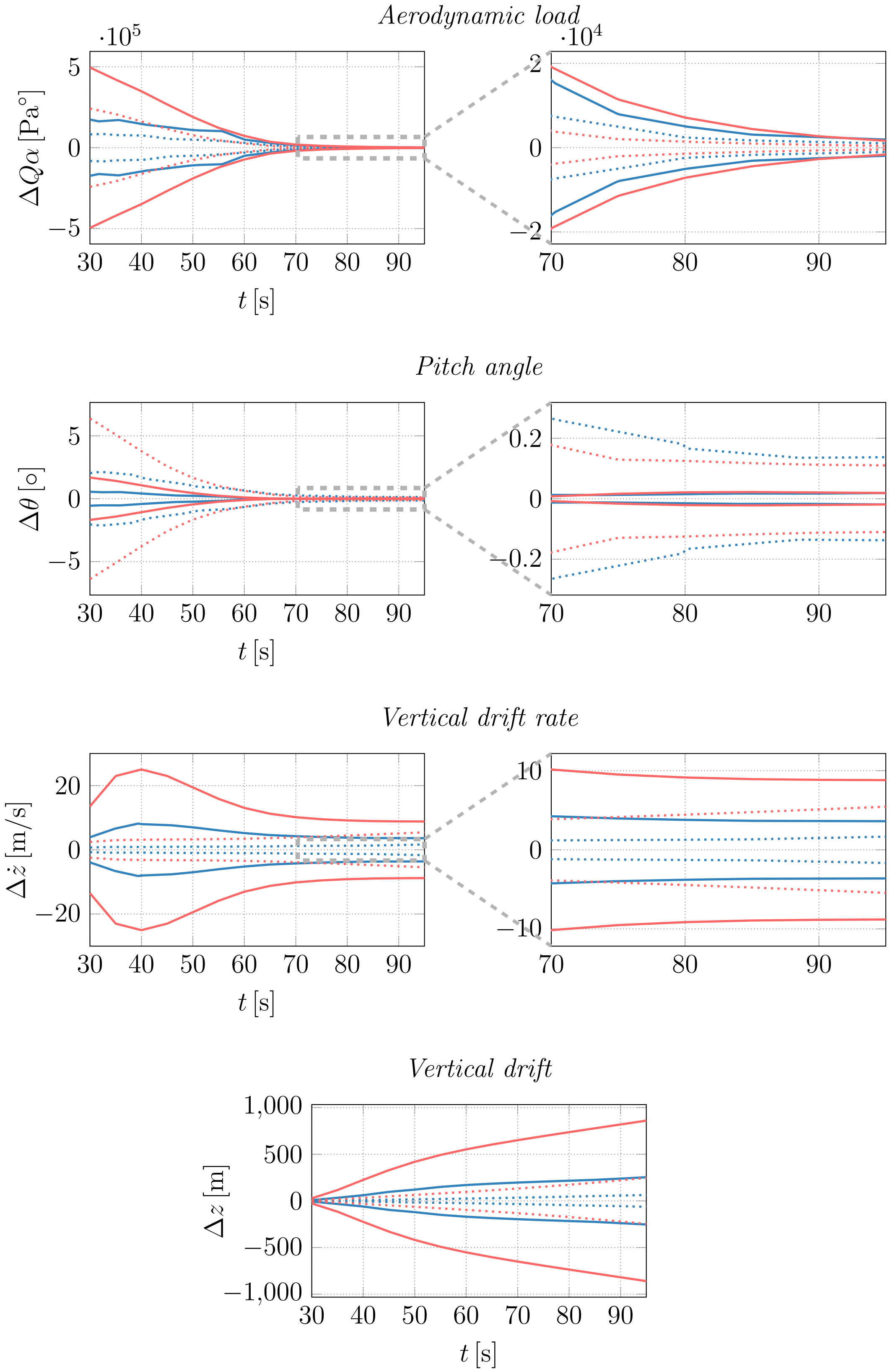


Figure 3.2: *Results for bounds on performance metrics: Upper bound LTV worst-case analysis $K_\alpha = 0.1$ (——) and $K_\alpha = 1.6$ (⋯⋯), upper bound Monte Carlo $K_\alpha = 0.1$ (——) and $K_\alpha = 1.6$ (⋯⋯)*

Table 3.1: Comparison of worst-cases identified by LTV and most critical Monte Carlo analysis results (MC)

	$Q\alpha$[Pa°]		$\Delta\theta$[°]		$\Delta\dot{z}$[m/s]		Δz[m]	
	LTV	MC	LTV	MC	LTV	MC	LTV	MC
$K_\alpha = 0.1$	$4.96 \cdot 10^5$	$1.74 \cdot 10^5$	1.68	0.55	25.02	8.17	859.99	253.22
$K_\alpha = 1.6$	$2.42 \cdot 10^5$	$8.48 \cdot 10^4$	6.38	2.14	5.45	1.66	243.50	63.32

region as well as the engine burnout, if they occur before the analyzed final time.

Furthermore, the LTV analysis provides direct feedback on the influence of the controller design updates. Comparing the results for both controllers in Fig. 3.2 as summarized in Tab. 3.1, shows a noticeable reduction in the maximal drift velocity and total deviation from the trajectory in the nonlinear analysis. The former reduces from 8.17m/s to 1.66m/s and the latter from 253.22m to 63.32m. This reduction is well predicted by the nominal LTV worst-case analysis, whose calculated worst-case drift and deviation reduced from to 25.02m/s to 5.45m/s and 859.99m to 243.50m, respectively. Here, the updated controller gains equate to a reduction by a factor 4.5 and 3.5, respectively.

As the launcher actively steers into the wind, the absolute value of $\Delta\theta$ increases in the Monte Carlo simulation from 0.55° to 2.14° compared to the stabilizing controller. Again, the LTV analysis provides quantitative feedback of the gain tuning showing an increase in $\Delta\theta$ from 1.68° to 6.38°. Surprisingly, the aerodynamic load reduces for the non-stabilizing controller despite steering into the wind. This is a result of the overall lower drift rates and improved trajectory tracking performance. Consequently, the gravity turn is better executed, and the maximum loads are reduced. In the nonlinear analysis, they drop from $1.74 \cdot 10^5$Pa° to $8.48 \cdot 10^4$Pa° and from $4.96 \cdot 10^5$Pa° to $2.42 \cdot 10^5$Pa° in the LTV analysis.

The complete LTV analyses were finished after 4min30s, whereas the Monte Carlo simulation, for a relatively small disturbance set, took 3h30min and will always only provide a lower bound. Thus, the finite horizon LTV approach demonstrates its suitability for analyzing unstable dynamics, providing a fast estimate of the system performance and an accurate match of the actual system's behavior. The insufficiency of the LTV worst-case bound for times after 60s is due to an inadequate wind filter. It is not designed explicitly for the LTV analysis framework inside the BRL.

3.6.3 Conclusion

For the analyzed launcher, only the LTV approach provides an accurate resemblance of the nonlinear system's dynamics independently of the evaluated controller. Furthermore, it predicted the performance improvement of the non-stabilizing controller also seen in the Monte Carlo simulation. Here, the LTV approach was roughly 47 times faster.

The nominal LTV approach can reduce the validation gap between linear and nonlinear analyses. However, an adequate wind filter inside the constraints of the BRL must be designed to assure adequate disturbance levels along the trajectory. This would allow for the provision of strict and not overly conservative upper bounds on simulation-based approaches.

The potential benefits of LTV analyses are even more prominent in the presence of uncertainty as the computational effort of Monte Carlo approaches scales with the covered perturbation sets. Consequentially, the LTV framework's extension to cover perturbed systems is imperative to advance the V&V process for space launchers and other highly time-varying systems.

3.7 Integral Quadratic Constraints

Usually, the linear model used for the controller design or performance analysis differs from the existing system. Various reasons for these differences exist, such as only approximately known parameters, imperfections of sensors, especially at unknown structure and model order at high frequencies, opting to work with a lower order approximation of very detailed models for simplicity and difference between the designed and implemented controller. These kinds of model differences can be divided into two classes, parametric and dynamic model uncertainties. The former covers uncertainties of parameters for models with known order and structure, and the latter cover uncertain dynamics, which are either purposely neglected or unknown. Furthermore, nonlinearities, e.g. saturations or dead-zones, and infinite-dimensional systems, e.g. time delays, cannot be accurately represented in the linear model. All the mentioned model differences are summarized under the notation Δ generally as perturbations. These perturbations are exclusively covered by employing IQCs in this thesis. IQCs present a generalized framework for the robustness analysis covering various perturbation types by bounding their input/output behavior.

3.7.1 Frequency-Domain

IQCs in the frequency domain are introduced in [226] via a multiplier Π. Π is measurable hermitian-valued function, $\Pi : j\mathbb{R} \rightarrow \mathbb{C}^{(n_v+n_w)\times(n_v+n_w)}$. The IQC defined by Π is satisfied by the two signals $v \in L_2[0,\infty)$ and $w \in L_2[0,\infty)$ if

$$\int_{-\infty}^{\infty} \begin{bmatrix} V(j\omega) \\ W(j\omega) \end{bmatrix}^* \Pi(j\omega) \begin{bmatrix} V(j\omega) \\ W(j\omega) \end{bmatrix} d\omega \geq 0, \tag{3.31}$$

where $V(j\omega)$ and $W(j\omega)$ are the Fourier-transforms of the $v(t)$ and $w(t)$, respectively. Consequently, if (3.31) holds for all $v \in L_2[0,\infty)$ and $w = \Delta(v)$, then the bounded, causal operator $\Delta : L_2[0,\infty) \rightarrow L_2[0,\infty)$ satisfies the IQC defined by Π. In [226], IQCs were introduced to analyze the robust stability and performance of uncertain LTI systems. Hence, the respective analysis conditions are formulated in the frequency-domain. Due to the LTV nature of the nominal systems analyzed in this thesis, a respective time-domain condition is required.

3.7.2 Time-Domain

A respective time-domain formulation of IQCs was derived in [145]. It is shown that any $\Pi \in \mathbb{RL}^{(n_v+n_w)\times(n_v+n_w)}$ can be factorized as $\Pi = \Psi^{\sim} M \Psi$, where $M \in \mathbb{S}^{n_z}$ is a symmetric matrix and $\Psi \in \mathbb{RH}_{\infty}^{n_z\times(n_v+n_w)}$. Note that such factorizations are not unique but can be computed via state-space methods such J-Spectral factorizations [146]. Let Ψ be factorized as (Ψ, M), then the IQC in (3.31) is satisfied by $v, w \in L_2(-\infty, \infty)$ iff $Z(j\omega) := \Psi(j\omega)$ satisfies $\int_{-\infty}^{\infty} Z^*(j\omega) M Z(j\omega) \geq 0$. Applying Parseval's theorem, the frequency domain constraint imposed on z can be transformed into an equivalent time domain constraint:

$$\int_0^{\infty} z(t)^T M z(t) dt \geq 0. \tag{3.32}$$

In (3.32), $z = \Psi \left[\begin{smallmatrix} v \\ w \end{smallmatrix}\right]$ is the output of the linear IQC filter Ψ:

$$\begin{aligned} \dot{x}_{\Psi}(t) &= A_{\Psi} x_{\Psi}(t) + B_{\Psi,1} v(t) + B_{\Psi,2} w(t) \\ z(t) &= C_{\Psi} x_{\Psi}(t) + D_{\Psi,1} v(t) + D_{\Psi,2} w(t) \end{aligned} \tag{3.33}$$

, with zero initial conditions. Consequentially, the IQC defined by $\Pi = \Psi^{\sim} M \Psi$ is satisfied by Δ, iff the time domain constraint in (3.32) is fulfilled for all $v \in L_2^{n_v}[0, \infty)$ and $w = \Delta(v)$. In Fig. 3.3, a graphical interpretation of a time-domain IQC is given. Generally, the constraint in (3.32) only holds over infinite horizons. These IQCs are called *soft*. In case the time domain inequality (3.32) holds for all finite times $T \geq 0$, the IQC:

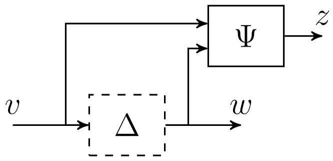


Figure 3.3: *Graphical interpretation of a time-domain IQC*

$$\int_0^{T} z(t)^T M z(t) dt \geq 0. \tag{3.34}$$

is called *hard*. The distinction is relevant for technical reasons. *Hard* IQCs are required for the robustness analysis of *gridded* LPV systems based on dissipativity theory [6]. In the context of LTV robustness analysis, it will only be required that the IQC holds over the considered analysis horizon $[0, T]$. The short notation $\Delta \in IQC(\Psi, M)$ is used if the perturbation Δ satisfies the IQC defined by Ψ and M over the interval $[0, T]$. Note that the *soft/hard* property is imposed by the applied factorization (Ψ, M). In [143], it was shown that under mild assumptions, for most of the multipliers in [83], a hard factorization exists.

The IQC framework allows k different perturbations $\Delta_i \in IQC(\Psi_i, M_i)$ to be included in a single IQC by diagonally combining them. Also, multiple IQCs satisfied by Δ can be combined in a single analysis. The latter is shown in the following example of a dynamic uncertainty:

Example 3.7.1.

For a dynamic LTI uncertainty (LTI system) Δ with $\|\Delta\|_\infty \leq b$, the boundedness of Δ imposes $\|w\|_{2[0,T]} \leq b\,\|v\|_{2[0,T]}$ for any input/output combination $v \in L_2[0, T]$ and $w = \Delta(v)$. Based on the causality of Δ, the constraint on v and w can be equivalently stated as the following time-domain inequality:

$$\int_0^T \begin{bmatrix} v(t) \\ w(t) \end{bmatrix}^T \begin{bmatrix} b^2 & 0 \\ 0 & -1 \end{bmatrix} \begin{bmatrix} v(t) \\ w(t) \end{bmatrix} dt \geq 0. \tag{3.35}$$

Hence, the IQC defined by $IQC_1(\Psi_1, M)$, with $\Psi_1 = I_2$ and $M = \left[\begin{smallmatrix} b^2 & 0 \\ 0 & -1 \end{smallmatrix}\right]$, is satisfied by Δ. Note that $z_1 = \left[\begin{smallmatrix} v \\ w \end{smallmatrix}\right]$ as Ψ has no dynamics. Furthermore, as Δ is LTI, it commutes with any stable minimum phase system $D(s)$, i.e. $\Delta D = D\Delta$. Therefore, also the system $\tilde{\Delta} := D\Delta D^{-1}$ is norm bounded by b. For the related input/output pair defined by $\tilde{v} = Dv$ and $\tilde{w} = Dw$, the inequality $\|\tilde{w}\|_{2[0,T]} \leq b\,\|\tilde{v}\|_{2[0,T]}$ holds. This relation can be equivalent to (3.36) written as:

$$\int_0^T z_2(t)^T M z_2(t) dt \geq 0, \tag{3.36}$$

where $M = \left[\begin{smallmatrix} b^2 & 0 \\ 0 & -1 \end{smallmatrix}\right]$ *and* $z_2 = \left[\begin{smallmatrix} \tilde{v} \\ \tilde{w} \end{smallmatrix}\right] = \left[\begin{smallmatrix} D & 0 \\ 0 & D \end{smallmatrix}\right]\left[\begin{smallmatrix} v \\ w \end{smallmatrix}\right] = \Psi_2\left[\begin{smallmatrix} v \\ w \end{smallmatrix}\right]$. *Thus,* Δ *satisfies the IQC defined by* $IQC_2(\Psi_2, M)$, *with* $M = \left[\begin{smallmatrix} b^2 & 0 \\ 0 & -1 \end{smallmatrix}\right]$ *and* $\Psi_2 = \left[\begin{smallmatrix} D & 0 \\ 0 & D \end{smallmatrix}\right]$. *Moreover, in [6] it was shown that if* Δ *satisfies each IQC separately, it also satisfies any conic combination build of them. This means the outputs* z_i *of the respective IQC filter* Ψ_i *fulfill the quadratic time constraint built by the conic combination of multipliers*

$$\int_0^T z_1(t)^T \lambda_1 M z_1(t) + z_2(t)^T \lambda_2 M z_2(t) dt, \quad \lambda_i > 0 \tag{3.37}$$

for all $v \in L_2[0, T]$ *and* $w = \Delta(v)$ *over the interval* $[0, T]$. *The IQCs in (3.37) can be stacked into the single IQC:*

$$\Psi = \begin{bmatrix} \Psi_1 \\ \Psi_2 \end{bmatrix} \quad \textit{and} \quad M(\lambda) = \begin{bmatrix} \lambda_1 M & 0 \\ 0 & \lambda_2 M \end{bmatrix} \tag{3.38}$$

More valid IQCs can extend the stacked IQC in (3.38) in an obvious fashion. Note that the dynamic system D corresponds to so-called D-scales used in the μ framework.

A time delay presents an infinite-dimensional system, which can not be covered explicitly in classic LTI frameworks such as μ. In [150], a detailed approach to derive and apply time-domain IQCs representation of time delays is given. The representation used in the course of this thesis is described in the following example.

Example 3.7.2. *The IQC representation is built by the conic combination of two IQCs. Both IQCs use the same matrix* $M_\tau = \left[\begin{smallmatrix} 1 & 0 \\ 0 & -1 \end{smallmatrix}\right]$ *but different filters* Ψ_{τ_1} *and* Ψ_{τ_2}. *The respective filters result from the J-spectral factorization* $\Pi_{\tau_i} = \Psi_{\tau_i}^{\sim} M_\tau \Psi_{\tau_i}$ *of the multiplier* $\Pi_{\tau_1} = \left[\begin{smallmatrix} 0 & -1 \\ -1 & 1 \end{smallmatrix}\right]$ *and* $\Pi_{\tau_2} = \left[\begin{smallmatrix} 0 & \phi^*(s) \\ \phi(s) & -1 \end{smallmatrix}\right]$, *with*

$$\phi(s) := \frac{-2.19(\frac{s}{\tau})^2 + 9.02(\frac{s}{\tau}) + 0.089}{(\frac{s}{\tau})^2 - 5.64(\frac{s}{\tau}) - 17.0}, \tag{3.39}$$

In (3.39), τ *is a selected constant time delay. Note that the calculated norm* *bound will also hold for all smaller τ. The parameterization of the combined IQC is confined to the set $\mathcal{M}_\tau$ defined by*

$$\mathcal{M}_\tau := \{diag(\lambda_1 M_\tau, \lambda_2 M_\tau) : \lambda_i > 0, i = 1, 2\}. \tag{3.40}$$

Thus, time delays can also be represented by stacked IQCs.

3.8 Summary

This chapter presented the theoretical background required for the theoretical advances proposed in this thesis. First, linear time-varying systems and their derivation from nonlinear models were introduced. Afterwards, signal and system norms for LTI and LTV systems were introduced. Thirdly, the bounded real lemma for linear time-varying systems was explicitly defined for the induced $L_2[0,T]$ and $L_2[0,T]$ to Euclidean gain. The latter was applied in the worst-case performance analysis of a space launcher to demonstrate the suitability of LTV analyses for highly time-varying systems. A Monte Carlo simulation of the corresponding nonlinear model was used for (successful) validation. In summary, IQCs were introduced as an approach to upper bound the input-output behavior of uncertainties.

Chapter 4

Worst-Case Analysis of Uncertain Finite Horizon Linear Time-Varying Systems

4.1 Introduction

This chapter presents an approach to compute the worst-case gain of the interconnection of a finite time horizon linear time-variant system and a perturbation. The input/output behavior of the uncertainty is described by integral quadratic constraints (IQC). A condition for the worst-case gain of such an interconnection can be formulated using dissipation theory as a parameterized Riccati differential equation, which depends on the chosen IQC multiplier. A nonlinear optimization problem is formulated to minimize the upper bound of the worst-case gain over a set of admissible IQC multipliers. Two tailored meta-heuristic optimization algorithms are developed to exploit the optimization problem's structure and solve it efficiently. The advantages over applying existing meta-heuristics are demonstrated on an extensive benchmark example.

4.2 Uncertain Linear Time-Varying Systems

An uncertain LTV system is described by the interconnection of a nominal LTV system G_t and the perturbation Δ as shown in Fig. 4.1 This interconnection represents a linear fractional transformation (LFT) denoted as $F_u(G_t, \Delta)$. See e.g. [21] for details. It can be generally written as:

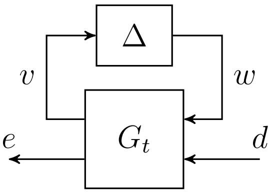


Figure 4.1: *Feedback interconnection LTV system G and uncertainty* Δ

$$\begin{aligned} \dot{x}_{G_t}(t) &= A_{G_t}(t)\, x_{G_t}(t) + B_{G_t}(t)\, d(t) \\ e(t) &= C_{G_t}(t)\, x_{G_t}(t) + D_{G_t}(t)\, d(t) \\ w(t) &= \Delta(v), \end{aligned} \tag{4.1}$$

where $v \in \mathbb{R}^{n_v}$ and $w \in \mathbb{R}^{n_w}$ are the perturbation input and output vectors, respectively. The operator $\Delta : L_2[0,T] \to L_2[0,T]$ defines the perturbation. The interconnection in Fig. 4.2 is said to be well-posed if, for all initial conditions $x_{G_t}(0)$ and $d(t) \in L_2[0,T]$ unique solutions $x_{G_t} \in L_2[0,T]$, $v \in L_2[0,T]$, and $w \in L_2[0,T]$ satisfying (3.7) and causally dependent on $d(t)$ exist.

4.3 Finite Horizon Linear Time-Varying Robustness Framework

A robust performance analysis is proposed utilizing the time-domain IQC representation of a perturbation Δ described in Section 3.7.2 and the worst-case analysis conditions for nominal LTV systems in Section 3.5. It allows for the worst-case analysis of the interconnection $F_u(G_t, \Delta)$.

This requires introducing the IQC filter Ψ into the interconnection shown in Fig. 4.1. Therefore, the input v and output w of the perturbation Δ are connected with Ψ, as shown in Fig. 4.2. Thus, the explicit input/output relation $w = \Delta(v)$ of the perturbation can be replaced by the time domain inequality (3.34) enforced on the filter output z. As Δ can be excluded from the interconnection as emphasized in Fig. 4.2, w is now treated as an external signal. The dynamics of the interconnection in Fig. 4.2 depend on an extended LTV system G defined by:

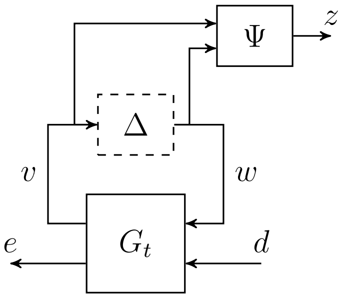


Figure 4.2: *Feedback interconnection LTV system G_t and uncertainty Δ extended with IQC filter Ψ*

$$\begin{aligned} \dot{x}(t) &= A(t)\,x(t) + \begin{bmatrix} B_1(t) & B_2(t) \end{bmatrix} \begin{bmatrix} w(t) \\ d(t) \end{bmatrix} \\ \begin{bmatrix} z(t) \\ e(t) \end{bmatrix} &= \begin{bmatrix} C_1(t) \\ C_2(t) \end{bmatrix} x(t) + \begin{bmatrix} D_{11}(t) & D_{12}(t) \\ D_{21}(t) & D_{22}(t) \end{bmatrix} \begin{bmatrix} w(t) \\ d(t) \end{bmatrix}, \end{aligned} \tag{4.2}$$

where $x(t) = [x_{G_t}(t)^T, x_\Psi(t)^T]^T \in \mathbb{R}^{n_{xG_t}+n_{x\Psi}}$ represents the state vector containing the states of G_t and Ψ, $d(t) \in \mathbb{R}^{n_d}$ the external disturbance input vector, and $e(t) \in \mathbb{R}^{n_e}$ the performance output vector.

The robust performance of an uncertain LTV system in the IQC framework can then be quantified by worst-case finite horizon input/output gains. Specifically, two metrics are used in this paper. Firstly, the finite horizon worst-case $L_2[0,T]$ to $\|e(T)\|_2$ gain for $D_{22} = 0$ is defined as follows:

$$\|F_u(G_t,\Delta)\|_2 := \sup_{\Delta \in \mathrm{IQC}(\Psi,M)} \sup_{\substack{d \in L_2[0,T] \\ d \neq 0, x(0)=0}} \frac{\|e(T)\|_2}{\|d(t)\|_{2[0,T]}}. \tag{4.3}$$

Geometrically interpreted, it describes the ball upper bounding the worst-case output $e(T)$ over all $\Delta \in IQC(\Psi, M)$ for $\|d(t)\|_{2[0,T]} = 1$ and the considered finite time horizon $[0, T]$ with $T \in [0, \infty)$. It is most applicable for evaluating physical performance values, such as quantitative design constraints, e.g gust loads or deviations from a given trajectory. The second performance measure is the finite horizon worst-case induced $L_2[0, T]$ gain:

$$\|F_u(G_t, \Delta)\|_{2[0,T]} := \sup_{\Delta \in \mathrm{IQC}(\Psi, M)} \sup_{\substack{d \in L_2[0,T] \\ d \neq 0, x(0)=0}} \frac{\|e(t)\|_{2[0,T]}}{\|d(t)\|_{2[0,T]}}. \tag{4.4}$$

It defines an upper bound on the worst-case amplification of the system's finite horizon 2-norm over all $\Delta \in IQC(\Psi, M)$ for inputs $d(t) \in L_2[0, T]$ and the time horizon $[0, T]$ with $T \in [0, \infty)$. It is most suitable to evaluate the maximum (energy) amplification from a disturbance input to performance output, e.g. required to evaluate worst tracking performance under uncertainty.

4.4 IQC Bounded Real Lemma for Linear Time-Varying Systems

A dissipation inequality using the extended system G (4.2) and the finite time horizon IQC (3.34) is formulated to upper bound either the worst-case gain in (4.3) or (4.4) of the interconnection $F_u(G_t, \Delta)$. An LMI condition for the upper bound γ is given in [151] following the approach in [6]. It is based on the definition of a time-dependent, continuously differentiable matrix function $P : \mathbb{R}_0^+ \to \mathbb{S}^{n_x}$. By $P(t)$, a quadratic storage function for the extended LTV system is defined. The LMI condition can be rewritten as an equivalent RDE applying Schur's complement, leading to the following Theorem 3 stating an analysis condition for the robust finite horizon worst-case $L_2[0, T]$ to Euclidean gain:

Theorem 3. *Let* $F_u(G_t, \Delta)$ *be well-posed* $\forall\, \Delta \in IQC(\Psi, M)$*, then* $\|F_u(G_t, \Delta)\|_2 < \gamma$ *if there exist a continuously differentiable* $P : \mathbb{R}_0^+ \to \mathbb{S}^{n_x}$

such that

$$P(T) = \frac{1}{\gamma} C_2(T)^T C_2(T), \tag{4.5}$$

$$\dot{P} = Q + P\tilde{A} + \tilde{A}^T P - PSP \qquad \forall t \in [0, T] \tag{4.6}$$

and

$$R = \begin{bmatrix} D_{11}^T M D_{11} & D_{11}^T M D_{12} \\ D_{12}^T M D_{11} & D_{12}^T M D_{12} - \gamma I_{n_d} \end{bmatrix} < 0, \tag{4.7}$$

with

$$\tilde{A} = \begin{bmatrix} B_1 & B_2 \end{bmatrix} R^{-1} \begin{bmatrix} (C_1^T M D_{11})^T \\ (C_1^T M D_{12})^T \end{bmatrix} - A, \tag{4.8}$$

$$S = -\begin{bmatrix} B_1 & B_2 \end{bmatrix} R^{-1} \begin{bmatrix} B_1^T \\ B_2^T \end{bmatrix}, \tag{4.9}$$

$$Q = -C_1^T M C_1 + \begin{bmatrix} (C_1^T M D_{11} \\ (C_1^T M D_{12} \end{bmatrix}^T R^{-1} \begin{bmatrix} (C_1^T M D_{11})^T \\ (C_1^T M D_{12})^T \end{bmatrix}. \tag{4.10}$$

Proof. The proof is based on the definition of a time-dependent quadratic storage function $V(t) = x(t)^T P(t) x(t)$ defined as $V : \mathbb{R}^{n_x} \times \mathbb{R}_0^+ \to \mathbb{R}$. As $F_u(G_t, \Delta)$ is well-posed, a unique solution for x_G, v, w, and e exists. Define the state vector of the extended system $x = \left[\begin{smallmatrix} x_G \\ x_\Psi \end{smallmatrix}\right]$. For $d \in L_2[0, T]$ and given initial conditions $x(0) = \left[\begin{smallmatrix} x_G(0) \\ 0 \end{smallmatrix}\right]$, the extended system (4.2) with inputs w, and d has a solution x, z and e. The filter output z also satisfies the time domain constraint (3.34).

After perturbing (4.6) with an infinitesimally small strictly positive scalar ϵ, the resulting Riccati differential inequality (RDI) can be transformed into an equivalent LMI applying Schur's complement. The equivalence is guaranteed by condition (4.7), which also ensures the invertibility of R. The resulting and reformulated LMI:

$$\begin{aligned}&\begin{bmatrix} P(t)A(t)+A(t)^TP(t)+\dot{P}(t) & P(t)B_1(t) & P(t)B_2(t) \\ B_1(t)^TP(t) & 0 & 0 \\ B_2(t)^TP(t) & 0 & -\gamma(1-\epsilon)I_{n_d} \end{bmatrix} \\ &+\begin{bmatrix} C_1(t)^T \\ D_{11}(t)^T \\ D_{12}(t)^T \end{bmatrix} M \begin{bmatrix} C_1(t) & D_{11}(t) & D_{12}(t) \end{bmatrix} \leq 0\end{aligned} \tag{4.11}$$

is left and right multiplied with $[x^T, w^T, d^T]$ and $[x^T, w^T, d^T]^T$, respectively. Applying the relations in (4.2) results in the dissipation inequality:

$$\begin{aligned}&\overbrace{\dot{x}(t)^TP(t)x(t)+x(t)^TP(t)\dot{x}(t)+x(t)^T\dot{P}x(t)}^{\dot{V}(t)}-\gamma(1-\epsilon)d(t)^Td(t) \\ &+z(t)^TMz(t)\leq 0\end{aligned} \tag{4.12}$$

Integrating (4.12) from 0 to T and applying the initial conditions gives

$$\begin{aligned}&x(T)^TP(T)x(T)-x_G(0)^TP_{11}(0)x_G(0)-\gamma(1-\epsilon)\overbrace{\int_0^T d(t)^Td(t)dt}^{\|d(t)\|^2_{2[0,T]}} \\ &+\int_0^T z(t)^TMz(t)dt\leq 0,\end{aligned} \tag{4.13}$$

with P_{11} being the upper left diagonal block matrix of P associated with x_G. As $\Delta \in IQC(\Delta, M)$, the last term in (4.13) can be neglected according to (3.34). Equality (4.5) is left and right multiplied with $x(T)^T$ and $x(T)$, respectively, resulting in

$$x(T)^TP(T)x(T)=\frac{1}{\gamma}x(T)^TC_2(T)^TC_2(T)x(T)=\frac{1}{\gamma}\underbrace{e(T)^Te(T)}_{\|e(T)\|_2^2}. \tag{4.14}$$

Substituting (4.14) in (4.13) and applying the Euclidean vector and finite horizon $L_2[0,T]$ norm accordingly results in:

$$\frac{1}{\gamma}\left\|e(T)\right\|_2^2 - x_G(0)^T P_{11}(0) x_G(0) - \gamma(1-\epsilon)\left\|d(t)\right\|_{2[0,T]}^2 \leq 0 \tag{4.15}$$

From (4.3), it follows that $x_G(0) = 0$. Thus, it can be concluded that $\|F_u(G_t,\Delta)\|$ is upper bounded by γ. □

Theorem 3 can be easily adjusted to formulate an upper bound on the robust worst-case induced $L_2[0,T]$ gain:

Theorem 4. *Let $F_u(G_t,\Delta)$ be well posed $\forall\,\Delta \in IQC(\Psi,M)$, then $\|F_u(G_t,\Delta)\|_{2[0,T]} < \gamma$ if there exist a continuously differentiable $P : \mathbb{R}_0^+ \to \mathbb{S}^{n_x}$ such that*

$$P(T) = 0, \tag{4.16}$$

$$\dot{P} = \hat{Q} + P\hat{A} + \hat{A}^T P - P\hat{S}P \qquad \forall t \in [0,T] \tag{4.17}$$

and

$$\hat{R} = \begin{bmatrix} D_{11}^T M D_{11} + D_{21}^T D_{21} & D_{11}^T M D_{12} + D_{21}^T D_{22} \\ D_{12}^T M D_{11} + D_{22}^T D_{21} & D_{12}^T M D_{12} + D_{22}^T D_{22} - \gamma^2 I_{n_d} \end{bmatrix} < 0, \tag{4.18}$$

with

$$\hat{A} = \begin{bmatrix} B_1 & B_2 \end{bmatrix} \hat{R}^{-1} \begin{bmatrix} (C_1^T M D_{11} + C_2^T D_{21})^T \\ (C_1^T M D_{12} + C_2^T D_{22})^T \end{bmatrix} - A, \tag{4.19}$$

$$\hat{S} = -\begin{bmatrix} B_1 & B_2 \end{bmatrix} \hat{R}^{-1} \begin{bmatrix} B_1^T \\ B_2^T \end{bmatrix}, \tag{4.20}$$

$$\begin{aligned} \hat{Q} = &- C_1^T M C_1 - C_2^T C_2 \\ &+ \begin{bmatrix} (C_1^T M D_{11} + C_2^T D_{21})^T \\ (C_1^T M D_{12} + C_2^T D_{22})^T \end{bmatrix}^T \hat{R}^{-1} \begin{bmatrix} (C_1^T M D_{11} + C_2^T D_{21})^T \\ (C_1^T M D_{12} + C_2^T D_{22})^T \end{bmatrix}. \end{aligned} \tag{4.21}$$

Proof. The proof is similar to Theorem 1. Based on the definition of a time-dependent quadratic storage function $V(t) = x(t)^T P(t) x(t)$. After perturbing (4.17), the resulting Riccati inequality can be rewritten as an LMI applying the Schur complement. Multiplying $[x^T, w^T, d^T]$ and $[x^T, w^T, d^T]^T$ on the left/right side, respectively, of the LMI results in a dissipation inequality. The integration provides the upper bound γ on $\|F_u(G_t, \Delta)\|_{2[0,T]}$ implied by $\|e(t)\|^2_{2[0,T]} \leq (1 - \epsilon)\gamma^2 \|d(t)\|^2_{2[0,T]}$ for the final condition $P(T)$, zero initial conditions, and $\Delta \in IQC(\Psi, M)$. □

4.5 Computational Approach

Some considerations are necessary to convert the results of Section 4.4 into a computationally feasible problem. In general, an infinite set of IQCs representing a given perturbation Δ exists. A common approach found in literature, see e.g. [6, 227], is selecting a fixed set of IQC filters and a free parameterization of M. Consequently, M lies within a feasibility set $\mathcal{M}$ such that $\Delta \in IQC(\Psi, M)$ for all $M \in \mathcal{M}$.

4.5.1 Worst-Case Gain Optimization Problem

The effects of a changing parameterization $M \in \mathcal{M}$, given a fixed choice of Ψ are best shown by example. Therefore, Theorem 4 is applied to calculate the induced finite horizon worst-case $L_2[0, T]$ gain of the Euler equation, a typical linear time-varying ordinary differential equation, in interconnection with an uncertainty.

Example 4.5.1. *The Euler equation is defined for $t \neq 0$ by the nominal finite horizon LTV system G_e:*

$$\begin{aligned} \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} &= \begin{bmatrix} 0 & 1 \\ -\frac{q}{t^2} & -\frac{p}{t} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{1}{t^2} \end{bmatrix} d \\ e &= \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}. \end{aligned} \tag{4.22}$$

G_e is extended by a multiplicative dynamic norm-bounded input uncertainty Δ ($\|\Delta\|_\infty \leq 1$) as shown in Fig. 4.3, to create the uncertain LTV system

$F_u(G_e, \Delta)$. Δ is represented by the IQC introduced in Example 3.7.1 built

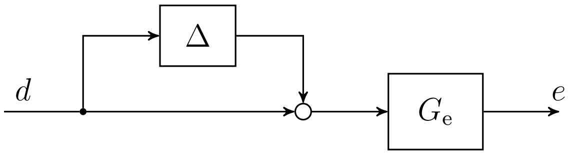
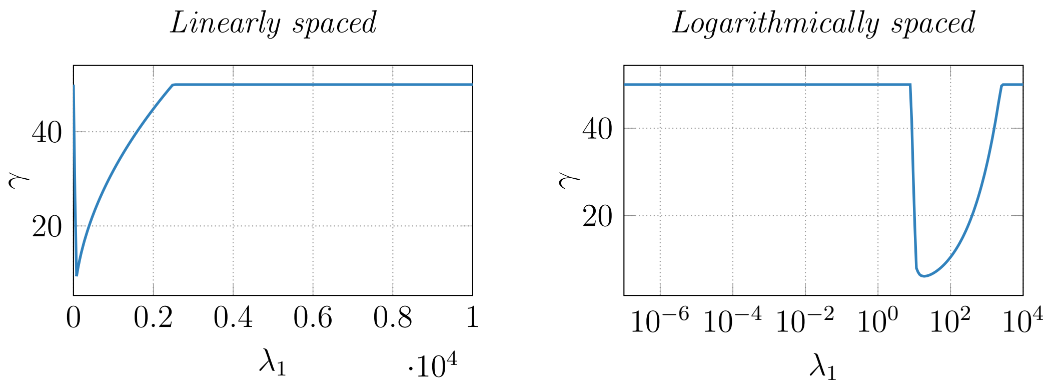


Figure 4.3: *Euler equation extended with a multiplicative input uncertainty*

by the static filter $\Psi_1 = I_2$ and the parameterization $M_1 = \lambda_1 \left[\begin{smallmatrix} 1 & 0 \\ 0 & -1 \end{smallmatrix}\right]$, with $\lambda_1 > 0$ denoted by $IQC_1(\Psi_1, M_1)$. Applying the IQC allows to convert the interconnection in Fig. 4.3 into the robust LTV framework, as described in Section 4.3. Hence, Theorem 4 can be applied to define an upper bound on the uncertain Euler equation's induced finite horizon worst-case $L_2[0,T]$ gain.

Given Theorem 4, the minimal achievable upper bound on the worst-case gain for a given Ψ_1 and feasible λ_1 value is defined by the smallest value of γ for which the RDE in (4.17) defined by $F_u(G_e, \Delta)$, Ψ_1, $M(\lambda_1)$, and γ is fully solvable. This minimal value of γ can be easily calculated via bisection over it constraint by the solvability of (4.17). The influence of λ_1 on the value of γ is evaluated by executing the bisection over a grid of 150 logarithmically and linearly space values λ_{1_i} in the interval $[10^{-7}, 10^4]$. A lower and upper bound of γ of 0 and 50, respectively is chosen. The bisection's absolute tolerance is $5 \cdot 10^{-7}$. For the integration of the RDE, the built-in Matlab function `ode15s` is applied, with an absolute and relative tolerance of 10^{-2} and 10^{-6}, respectively. This is a solver specialized on stiff ODEs [214], RDEs generally fall into this category [228]. The analysis horizon spans from 5s to 10s seconds. In Fig. 4.4, the resulting γ_i over λ_{1_i} are displayed with a linearly scaled x-axis for the linearly spaced grid on the left and a logarithmically scaled x-axis for the logarithmically spaced grid on the right. The minimal value calculated for γ is 6.0967 given $\lambda_1 = 18.5548$. In total, 8 seconds were required to calculate all γ_i for the logarithmic case with the bisection parallelized on eight processors of an Intel i7 in a standard desktop computer with 32GB memory. The linear case required 5s in the same setup and provided a minimal γ of 8.8261 for $\lambda_1 = 26.2213$.



Linearly spaced

Logarithmically spaced

Figure 4.4: *Minimal realizable value of γ for a given value of λ_1 calculated via bisection*

Using a second IQC representation for Δ, the influence of the selected IQC filter and the resulting richer parameterization is demonstrated. Following example 3.7.1, Δ is represented by the conic combination of two IQCs, namely $IQC_1(\Psi_1, M_1)$ and $IQC_2(\Psi_1, M_2)$. The factorizations are selected as $\Psi_1 = I_2$ and $\Psi_2 = \frac{1}{s+1} I_2$ with the respective parameterizations $M_1 = \lambda_1 \left[\begin{smallmatrix}1 & 0\\ 0 & -1\end{smallmatrix}\right]$ and $M_2 = \lambda_2 \left[\begin{smallmatrix}1 & 0\\ 0 & -1\end{smallmatrix}\right]$, with $\lambda_i > 0$. For the evaluation of the minimal possible γ given a fixed λ_1 and λ_2, both are gridded on the interval $[10^{-7}, 10^{4}]$ by 150 logarithmically spaced points. Hence, this results in an analysis grid of 22500 points. Subsequently, the bisection is executed over all possible combinations of λ_{1_i} and λ_{2_j} to identify the corresponding minimal γ_{ij}. The rest of the analysis setup remains unchanged. In Fig. 4.5 the behavior of this minimal γ with respect to λ_1 and λ_2 is shown, where the x and y-axis are logarithmically scaled.

A minimal γ value of 6.0940 for the parameterization $\lambda_1 = 18.5543$ and $\lambda_2 = 1.4945 \cdot 10^{-4}$ was calculated. The analysis was completed in 22min and 15s. Hence, the richer IQC parameterization only provides an negligible lower minimal γ with a 167 times higher computational cost.

As emphasized by Example 4.5.1, the upper bound γ on the worst-case gain in Theorem 3 and Theorem 4 directly depends on the choice of the IQC parameterization, for a chosen Ψ. Evaluating a grid of feasible parameterizations is

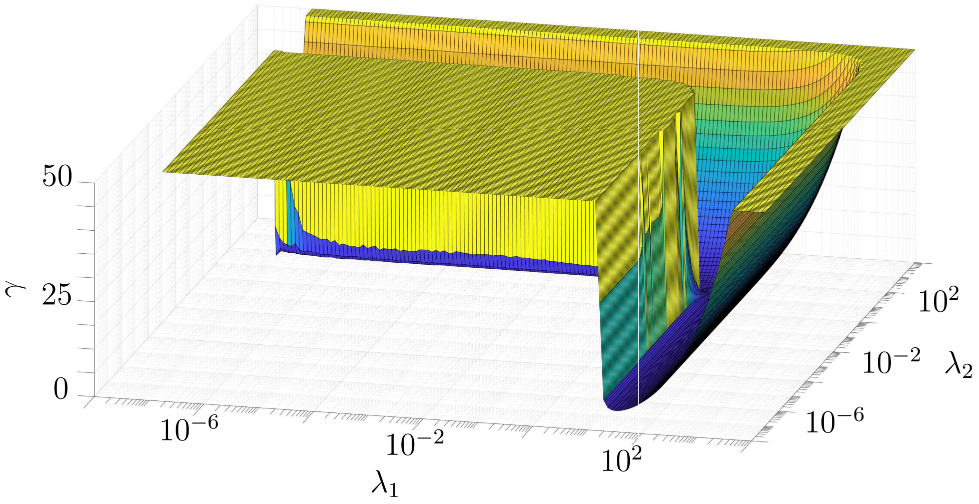


Figure 4.5: *Minimal realizable value of γ for a given value of λ_1 and λ_2 calculated via bisection*

not feasible in an industrial application, as it is unlikely that it contains the exact value of M connected to the global minimum of γ. Furthermore, it is computationally inefficient for larger parameterizations. Hence, to obtain the lowest upper bound γ of the worst-case input/output gain of the uncertain interconnection, an optimization over the IQC parameterization $M(\lambda)$ given a fixed Ψ must be performed. The RDE in (4.6) and (4.17), respectively, is parameterized by the IQC matrix M. However, due to the application of Schur's complement, M enters the RDEs nonlinearly. Thus, a nonlinear optimization problem directly over the parameterization $M \in \mathcal{M}$ minimizing γ constrained by the integrability of the RDE can be derived. This optimization approach is also motivated by the shapes of the graphs in Fig. 4.4 and Fig. 4.5. In the case of the finite horizon worst case $L_2[0, T]$ to $\|e(T)\|_2$ gain, it is written as:

$$\min_{M \in \mathcal{M}} \gamma$$

such that $\forall t \in [0, T]$

$$\begin{aligned} &P(T) = 0 \\ &\dot{P} = \hat{Q} + P\hat{A} + \hat{A}^T P - P\hat{S}P \\ &\hat{R} < 0. \end{aligned} \tag{4.23}$$

The nonlinear optimization problem for the finite horizon worst-case $L_2[0,T]$ gain can easily be derived from (4.23) by replacing $\hat{A}$, $\hat{S}$, $\hat{Q}$, and $\hat{R}$ with $\tilde{A}$, S, Q, and R respectively, and changing the final condition to $P(T) = \frac{1}{\gamma}C_2(T)^T C_2(T)$.

4.5.2 Requirement Specification for the Nonlinear Program

Note that in general, M enters (4.23) in a non-convex way, as clearly visible in Fig. 4.5. However, given the shape of the results of γ over the IQC parameterization in Fig. 4.4 and Fig. 4.5, the problem appears to be locally convex for large areas of the search space. Thus, a direct optimization over $M \in \mathcal{M}$ appears promising. However, a global optimization algorithm must be applied, to avoid to get stuck in local minimums during the search.

Furthermore, a derivative-free search procedure is required. As the calculation of analytical derivatives for the problem (4.23) is infeasible, only numerical derivatives could be used. Two main reason render this method impractical. Firstly, integrating the RDE (4.6) or (4.17) becomes computationally more expensive for longer analysis horizons and especially richer parameterizations. Note that the RDE is a matrix function, with $\dot{P} \in \mathbb{R}^{n_x \times n_x}$ and thus, n_x^2 scalar RDEs must be solved. Hence, the total evaluations of the RDE (4.6) or (4.17) should be reduced to a minimum. Secondly and even more importantly, the optimization problem must be assumed non-smooth, because of its non-convexity.

Evaluating Example 4.5.1 for the conic combination of $IQC_1(\Psi_1, M_1)$ and $IQC_2(\Psi_2, M_2)$ for a wider and denser logarithmically spaced grid of 200×200 values of λ_1 and λ_2, each in $[10^{-10}, 10^4]$, show that the optimization problem is not strictly convex around the identified minimal γ, as the gradient there is zero. Also, local minimums of γ for low values of λ_1 at the boundary of the search space can be found. These are attributed to numerical issues occurring during the integration process, due to very high condition numbers of the matrix function $\hat{R}$. This observation is emphasized with Fig. 4.6, where besides the overall result, the γ value over λ_2 for a fixed λ_1 value of 18.0419 and equivalent plot for fixed value of $\lambda_2 = 24.9451$ is plotted. Thus, gradient information cannot be used and the non-smoothness must be robustly handled. This can be problematic, especially for multiple uncertainties and, thus, more complicated parameterizations. Thus, an optimizer that does not require initial solutions is favorable for this optimization problem. Recall that although the worst-case finite horizon gains are theoretically always bounded, the underlying RDE has a finite escape time, i.e. it is not necessarily solvable over the whole analysis horizon as it blows-up to infinity [125, 228]. Also, identifying an adequate initial search space is critical as a too narrow search space could result in not identifying the global minimum, comparing Fig. 4.5 and 4.6. Consequently, adaptive search bounds would help avoid an extensive initial search space, especially for small parameterizations.

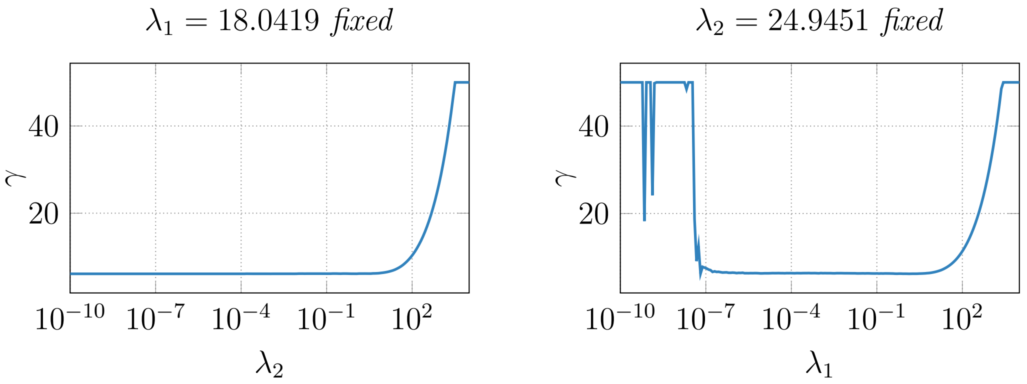


Figure 4.6: *Non-Convexity of γ over λ_1 and λ_2 for the Euler equation*

4.6 Meta-Heuristics for the Worst-Case Optimization Problem

Two novel custom-tailored optimization algorithms are proposed based on the requirements in Section 4.5.2 to deal with the nonlinear problem (4.23) efficiently. Both algorithms follow a similar concept, consisting of a simple bisection nested within a meta-heuristic optimizer. The bisection is used to obtain a minimal γ for a given M, i.e. bisect (4.23) with a fixed $M \in \mathcal{M}$. This specific M is provided by the meta-heuristic, performing the optimization over $M \in \mathcal{M}$. Thus, the main difference between both algorithms is the applied MH search strategy, a logarithmically scaled self-adaptive differential evolutionary optimizer with linear population size reduction (Log-L-SHADE), or an adaptive boundary sine cosine algorithm with population size reduction (Ab-SCA-PR). Although, the MHs' principle search procedures differ, the LTV IQC optimization problem-specific adaptations in them are similar and described in the following sections.

4.6.1 Necessary Modifications

MHs present a direct and global optimization method. The latter is achieved through a degree of randomization in the search process, which allows the solution to "escape" from local minimums. In general, they do not require a user guess for an initial solution but utilize a random one inside the search space. Due to the randomization, this initial solution does not necessarily need to be valid. A large enough population size, i.e. set of solution vectors and iteration amount, will eventually lead to a feasible solution vector. The randomization in the search procedure is also beneficial for handling the non-smoothness of the problem. Also, MHs are parallelizable. This supports larger parameterization, exploiting the fact that multi-core processors are standard in industry and academia. Hence, MHs check most of the requirements in Section 4.5.2 to allow for robustly solving the LTV IQC optimization problem (4.23).

However, most if not all MHs have been developed to deal with bound-constrained optimization, which can be generally expressed as:

$$\min : f(\lambda)\,;\ \lambda_L \leq \lambda \leq \lambda_U, \tag{4.24}$$

where λ is a vector of n_λ design variables λ_k, f is an objective function, and λ_L and λ_U are respectively the lower and upper bounds of λ. MHs are directly applicable to simple IQCs or conic combinations of IQCs as shown in Example 3.7.1 using a large enough upper bound. More general IQC representations impose additional constraints.

Nonlinear Constraints on the Design Variables

The optimization problem in (4.23) is more complicated with respect to the design variables. In general, the elements of the design vector $\lambda \in \mathbb{R}^{n_\lambda}$ building the IQC parameterization $M(\lambda)$ are not bounded. They are only constrained by the structure imposed by $M(\lambda) \in \mathcal{M}$, which in fact renders certain value combinations infeasible. More general IQC representation for structured and full-block dynamic, as well as parametric uncertainties are given in [227] and presented in the next three examples.

Example 4.6.1. *Let $\Delta = \delta I_{n_v}$ be a LTI real diagonally n_v repeated parametric uncertainty δ, with $\delta \in \mathbb{R}$ and $|\delta| \leq b$, with $b \in \mathbb{R}$. A valid time domain IQC for Δ is defined by $\Psi = \left[\begin{smallmatrix} \psi_\nu \otimes I_{n_v} & 0 \\ 0 & \psi_\nu \otimes I_{n_v} \end{smallmatrix}\right]$ and $\mathcal{M} := \{M = \left[\begin{smallmatrix} b^2 X & Y \\ Y^T & -X \end{smallmatrix}\right] : X = X^T > 0 \in \mathbb{S}^{n_v(\nu+1)}, Y = -Y^T \in \mathbb{R}^{n_v(\nu+1)\times n_v(\nu+1)}\}$. A typical choice for $\psi_\nu \in \mathbb{RH}_\infty^{(\nu+1)\times 1}$ is:*

$$\psi_\nu = \begin{bmatrix} 1 & \frac{s+\rho}{s-\rho} & \dots & \frac{(s+\rho)^\nu}{(s-\rho)^\nu} \end{bmatrix}^T, \rho < 0\,, \nu \in \mathbb{N}_0. \tag{4.25}$$

Example 4.6.2. *Let Δ be a LTI dynamic uncertainty, with $\Delta \in \mathbb{RH}_\infty$ and $\|\Delta\|_\infty \leq b$. A valid time domain IQC for Δ is defined by $\Psi = \left[\begin{smallmatrix} \psi_\nu \otimes I_{n_v} & 0 \\ 0 & \psi_\nu \otimes I_{n_v} \end{smallmatrix}\right]$ and $\mathcal{M} := \{M = \left[\begin{smallmatrix} b^2 X & 0 \\ 0 & -X \end{smallmatrix}\right] : X = X^T > 0 \in \mathbb{S}^{n_v(\nu+1)}\}$. A typical choice for $\psi_\nu \in \mathbb{RH}_\infty^{(\nu+1)\times 1}$ is:*

$$\psi_\nu = \begin{bmatrix} 1 & \frac{s+\rho}{s-\rho} & \dots & \frac{(s+\rho)^\nu}{(s-\rho)^\nu} \end{bmatrix}^T, \rho < 0\,, \nu \in \mathbb{N}_0. \tag{4.26}$$

Example 4.6.3. *Let Δ be a full-block dynamic LTI uncertainty, with $\Delta \in \mathbb{RH}^{n_w \times n_v}$ and $0 < \|\Delta\|_\infty \leq b$. A valid time domain IQC for Δ is defined by*

$\Psi = \left[\begin{smallmatrix} b\psi_\nu \otimes I_{n_v} & 0 \\ 0 & \psi_\nu \otimes I_{n_w} \end{smallmatrix}\right]$ *and* $\mathcal{M} := \{M = \left[\begin{smallmatrix} X \otimes I_{n_v} & 0 \\ 0 & -X \otimes I_{n_w} \end{smallmatrix}\right] : X = X^T \geq 0 \in \mathbb{S}^{(\nu+1)}\}$.

A typical choice for $\psi_\nu \in \mathbb{RH}_\infty^{(\nu+1)\times 1}$ *is:*

$$\psi_\nu = \begin{bmatrix} 1 & \frac{1}{(s-\rho)} & \dots & \frac{1}{(s-\rho)^\nu} \end{bmatrix}^T, \, \rho < 0 \, , \, \nu \in \mathbb{N}_0. \tag{4.27}$$

In all three cases, the matrices X and Y are the optimization variables built by λ in (4.24). Y is a skew-symmetric matrix whose off-diagonal elements are unbounded. X must be positive definite and symmetric, which adds a nonlinear constraint to the optimization problem. Thus, the diagonal entries must be strictly positive, whereas the upper diagonal values are in principle unbounded. However, the positive definiteness renders certain parameter combinations infeasible. Assuming that $\nu = 1$ and $n_v = n_w = 1$, $X = \left[\begin{smallmatrix} x_1 & x_2 \\ x_2 & x_3 \end{smallmatrix}\right] > 0$, iff $x_1 > 0$, $x_2 > 0$, and $x_1 x_3 - x_2^2 > 0$. For larger matrices of block structure, a similar condition is provided by Schur's complement. See [225] for more details. The positive definiteness of matrices like X is covered in the developed algorithms exploiting the fact that any indefinite symmetric matrix X can be made positive definite by adding the absolute value of a real number smaller than its smallest eigenvalue $\lambda_{\min}$ to every diagonal entry. Bad conditioning of the matrix X is avoided by increasing this correction value by a positive random number of the same magnitude as $\lambda_{\min}$. Here, a random number is chosen to follow the general randomization solution updates in MHs.

Nonlinear Constraint Imposed by the RDE's Solvability

The solvability of the RDE (4.6) and (4.17), respectively, which is directly related to the existence of the optimized upper bound γ, imposes another nonlinear constraint. It is handled by a bisection, which is nested in the MHs algorithm providing the respective $M \in \mathcal{M}$. The general implementation follows Example 4.5.1, with some practical changes for more flexible and efficient implementation for the worst-case optimization. In Algorithm 1, pseudo-code illustrates the general implementation used in the thesis.

Algorithm 1 Bisection to calculate local minimal γ

```
Input: $G$, $M$, $\gamma_{\text{LB}}$, $\gamma_{\text{UB}}$, $T$
Output: $\gamma = \gamma_{\text{UB}}$
Initialize: $\epsilon_{\text{BS}}$, $P(T)$, $\epsilon_{\text{rel,ODE}}$, $\epsilon_{\text{abs,ODE}}$
Solve Bisection: Calculate the $\gamma$ for a fixed $M$ provided by the MH
while $\frac{\gamma_{\text{UB}}-\gamma_{\text{LB}}}{\gamma_{\text{UB}}} \geq \epsilon_{\text{BS}}$ do
    if First execution of respective $M$ then
        $\gamma_{\text{Try}} = \gamma_{\text{UB}}$
    else
        $\gamma_{\text{Try}} = \frac{\gamma_{\text{LB}}+\gamma_{\text{UB}}}{2}$
    end if
    Solve RDE: The RDE (4.6)/(4.17) from $G$, $M$ and $\gamma_{\text{Try}}$ is solved
    backwards from $T$ to 0 with initial condition $P(T)$ and tolerances $\epsilon_{\text{rel,ODE}}$
    and $\epsilon_{\text{abs,ODE}}$. Terminate integration if eigenvalues of $\dot{P}(t)$ "blow-up".
    Output: Solution $P(t)$ and time vector $t_{\text{RDE}}$ of the integration.
    if RDE is fully solvable, i.e. $\min(t_{\text{RDE}}) = 0$ then
     $\gamma_{\text{UB}} = \gamma_{\text{Try}}$
    else
     $\gamma_{\text{LB}} = \gamma_{\text{Try}}$
    end if
end while
```

The bisection is evaluated between a user-defined upper and lower bound γ_{UB} and γ_{LB}, respectively until a user-specified relative tolerance ϵ_{BS} is reached. Note that the initial run is evaluated for γ_{Try} equaling the upper bound to identify if the bounds are suitable and avoid unnecessary evaluations. If the upper bound is infeasible, the bisection for the given bounds is not executed. Instead, the upper bound is upscaled by a factor of 1000. This procedure is repeated until either an upper bound is found feasible or a maximum upper bound of 10^{20} is reached. The upscaling allows for narrower initial bisection bounds. In combination with the pre-check, it increases computational efficiency. Note that the value fix value of 1000 was chosen based on experience. However, smaller and larger values can be chosen founded on the expected value range of γ

The RDE corresponding to the applied theorem is integrated backwards in time from the respective terminal condition $P(T)$ with the user-specified absolute and relative tolerance $\epsilon_{\text{ODE, abs}}$ and $\epsilon_{\text{ODE, rel}}$, respectively. In this thesis, exclusively the built-in Matlab ODE solver `ODE15s` is applied for solving the RDE. This solver is especially designed for stiff ODEs. For all application problems evaluated in this thesis, it outperformed Matlab's other ODE solvers (`ODE45` (non-stiff), `ODE23` (non-stiff), `ODE113` (non-stiff), `ODE23s` (stiff), `ODE23t` (stiff), and `ODE23tb` (stiff)) with respect to the required integration time. An event function recognizes if the RDE "blows-up" up, i.e. the solution $P(t)$ becomes unstable and approaches infinity, due to an escape time shorter than the analysis horizon. Therefore, it calculates the maximum eigenvalue of $\dot{P}(t)$ and terminates the integration if necessary. This avoids computationally expensive integration attempts and reduces the overall execution time of the bisection. If no analytical description of $G(t)$ is provided, piecewise cubic Hermite interpolating polynomials (PCHIPs) of the system matrices are used to describe $G(t_\mathrm{s})$ at the integration time step t_s. This guarantees continuously differentiable representations of the system matrices, which are generally assumed for the solvability of non-autonomous RDEs [124, 125, 228]. It has to be mentioned that, in general, integration can not be avoided. This is because the $Q/\hat{Q}$ matrix is indefinite due to the applied positive negative IQC multipliers. The strict definiteness of $Q/\hat{Q}$ is a necessary condition of common existence theorems, e.g. Theorem 4.1.6 in [228], which avoid solving the actual RDE. Neither did the solution of the equivalent linear system derived via Radons Lemma, see e.g. [229], show any benefit over solving the original RDE due to the resulting Hamiltonian system's instability.

Nonlinear constraint imposed by $R < 0$

The last nonlinear constraint on the optimization problem (4.23) is imposed by the condition $R < 0/\hat{R} < 0$. Note that for most valid IQC parameterizations $M \in \mathcal{M}$ the condition is automatically fulfilled. Nevertheless, $R < 0$ must be checked in the algorithm. The condition $R < 0$ is checked inside the bisection as R is a function of the bisected γ and M. If $R \geq 0$, the integration is skipped and this γ_Try is treated equivalently to an incomplete integration.

Note that another critical issue regarding the constraint $R < 0$ is that the condition number of R can become very large for valid IQC parameterizations, which has adverse numerical effects on the inverting of R and integration of the RDE. The latter is visible by a significant increase in the required integration time. Extensive test scenarios showed that these solutions do not relate to global minimums. Therefore, a user-defined upper bound on R's condition number can be proposed. It can be checked before the bisection for M and γ_{UB}, and if violated, the bisection is skipped, or if the search procedure allows it a new M resulting in a valid R can be proposed.

4.6.2 Efficient Implementation and Application

By exploiting the LTV IQC optimization problem's characteristics in the search and the general worst-case analysis procedure, the MH's computational efficiency can be significantly increased. The main goal must be the reduction of the overall amount of RDE evaluations and narrowing the technically infinite search space. Note that all the subsequently discussed adaptations and analysis strategies not only hold true for MHs developed in this thesis but also in general. Therefore, this section can be viewed as a general guideline for applying MHs efficiently to the robust LTV analysis framework.

Exploiting Symmetry

Although the integration of the RDE (4.6)/(4.17) cannot be avoided, the computational effort solving it can be reduced by exploiting the symmetry of P. As the RDE is hermitian, it is sufficient to solve the upper triangular portion and diagonal of this matrix differential equation. Hence, only $0.5(n_x(n_x + 1))$ rather than the original n_x^2 equations have to be solved per RDE evaluation, i.e. for large systems, the computational effort is nearly halved.

Narrowing the Bisection Bounds and Avoiding Unnecessary Bisections

Narrowing the initial bisection bounds reduces the number of bisection steps, and consequently, RDE evaluations. In general, the robust worst-case gain is lower bounded by the nominal worst-case gain, and its value can be taken as initial guess for the lower bound γ_{LB}. In some cases, a theoretical lower bound larger than the nominal γ exists. As $R = \left[\begin{smallmatrix} R_{11} & R_{12} \\ R_{21} & R_{22} \end{smallmatrix}\right] < 0 \Leftrightarrow R_{22} < 0$ and $R/R_{22} = R_{11} - R_{12}R_{22}^{-1}R_{21} < 0$ (Theorem 1.12 in [225]), it follows from Theorem 1 and Theorem 2 that $\gamma^2 > D_{12}^T M D_{12} + D_{22}^T D_{22}$ and $\gamma > D_{12}^T M D_{12}$, respectively. This step further reduces the computational cost of the bisection by narrowing the maximum bisection interval.

The selection of an upper bound γ_{UB} for the initial population is difficult and usually handled by simply selecting a "large enough" value, i.e. several magnitudes larger than γ_{LB}. However, for subsequent populations, a characteristic shared by all MHs can be exploited, only updated solution vectors (children) whose γ value (fitness) is smaller than the origin (parent) solution's γ will be used to update the solution vector (reproduction). Consequently, the initial γ_{UB} to evaluate children can be set to the parent's γ, and before the bisection starts the RDE can be evaluated for said γ_{UB}. If the RDE cannot be solved, the children are no improvement over its parent, the children must not be used for reproduction, and no bisection is necessary as the actual degradation information is not required to proceed in the search. In the case that a solution exists, γ_{UB} can be used as the initial upper bound. Altogether, this significantly reduces the number of RDE integrations. How well MHs handle the reduction of search information is shown in the benchmark example in Section 4.7.

Simplifying the Identification of Initial Solution Sets

For more extensive IQC representations, identifying valid initial solutions showed to be difficult for the tested MHs. Albeit, the search could commence without any valid initial solution, providing such increases the search performance. If the IQC represents dynamic or parametric uncertainties, down-scaling the individual norm bounds b enlarges the feasible search space, i.e. the search space for which a solution can be found. This effect is investigated on the example of the Euler equation from Example 4.5.1 given $IQC_1(\Psi_1, M_1)$ by reducing b from 1 to 0.2 in 0.2 increments and 150 λ_1 values logarithmically spaced in $[10^{-2}, 10^{6}]$. The resulting γ over λ_1 values are shown in Fig. 4.7. It can be seen that, the feasible search space given $\gamma_{\mathrm{UB}} = 50$ increases significantly, but the identified minimum remains inside the same magnitude. Moreover, the identified minimums for smaller b are in the feasible search space of the larger b, as the norm bound insignificantly changes the optimal solution. This effect is exploited for the rescaling of the norm bounds necessary later in the optimization. Note that the computational overhead introduced by the scaling/rescaling procedure is marginal as the rescaling only requires a recalculation of the current solution set's fitness values.

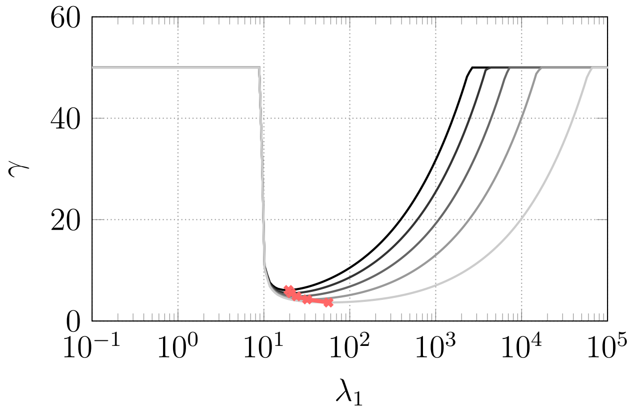


Figure 4.7: *Comparison of the valid search space for reducing uncertainty norm-bounds b: $b = 1$ (——), $b = 0.8$ (——), $b = 0.6$ (——), $b = 0.4$ (——), $b = 0.2$ (——), identified minimums (—✱—)*

Utilizing Previous Results

The observed correlation between b and the optimal solution seen in Fig. 4.7 can be exploited in analyses evaluating a control system's performance degradation for an increasing amount of uncertainty, i.e. increasing norm bounds.

Here, the optimal solution of a preceding b can be used as initial guess and narrow the search space, improving the overall analysis performance and efficiency. Furthermore, the respective minimal γ can be used as γ_{LB} for the subsequent b value. This strategy is applied in the tracking analysis conducted in Section 4.7.

A similar correlation exists between the analysis horizon and the position of the optimal solution, as emphasized in Fig. 4.8. Here, the analysis' final time T is increased in 5s steps from 10s to 30s, and the respective γ values are plotted over λ_1. It can be seen that adjacent analysis horizon's minimums are inside the same magnitude. Consequently, the optimal solution of preceding final times can be used as initial guess and narrowing the search space proceeding in the analysis. Exploiting this correlation is particularly helpful for trajectory analyses, i.e. analyses where the worst-case values of performance criteria are evaluated along the whole trajectory. This policy is applied in Chapter 6, where a space launcher's atmospheric flight phase is analyzed.

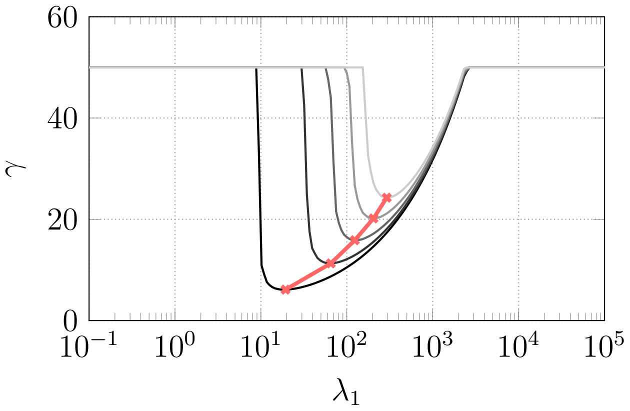


Figure 4.8: *Comparison of the optimum locations for increasing final times T: $T = 10$ (——), $T = 15$ (——), $T = 20$ (——), $T = 25$ (——), $T = 30$ (——), identified minimums (—✖—)*

Furthermore, changes in the nominal system's parameters/coefficients impact the location of the identified minimum, as long as the overall structure remains the same. This is shown by increasing p and q in (4.22) and evaluating γ over λ_1. The results are displayed in Fig. 4.9. Starting from a p and q values of 3 and 2, respectively, the values are doubled for each subsequent evaluation until $p = 48$ and $q = 32$ are reached. As the minimums' position between subsequent points does not noticeably change, in iterative gain tuning processes in the control design process, information from previous designs can be used as initial guess for the initial solution and search space.

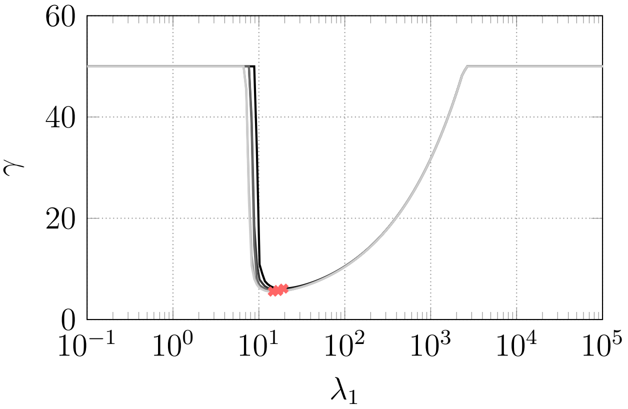


Figure 4.9: *Comparison of the optimum locations for different combinations* (p, q) *in (4.22):* $(3, 2)$ *(—),* $(6, 4)$ *(—),* $(12, 8)$ *(—),* $(24, 16)$ *(—),* $(48, 32)$ *(—), identified minimums (—✱—)*

Identifying Suitable Terminal Conditions

Meta-heuristics and global optimizations generally only utilize terminal conditions in the form of a maximum number of cost function evaluations or solution set iterations. As no convergence guarantees exist and arbitrary local minimums can exist, using relative and absolute tolerances on the function value as terminal conditions are infeasible. A common approach is to use a global optimizer to find the "rough" region of the global minimum, which is then calculated using local optimization. However, most of the time, the precise worst-case gain is not required, and it is sufficient to know if a preset limit value is not violated. Hence, the MHs in this thesis are extended with an additional terminal condition based on a minimal achievable function value to reduce the computational effort in such a case.

Parallelizing the Search Procedure

As the main computational cost driver is the number function evaluations, due to the integration of the RDE, all proposed meta-heuristics have fully parallelized cost function evaluations. This means, the bisections for a given set of solutions $M \in \mathcal{M}$ are run in parallel on the available processor cores.

4.6.3 Ab-SCA-PR

The first MHs algorithm proposed is a novel adaptive boundary sine cosine algorithm with population size reduction (Ab-SCA-PR). It is dedicated to solve for LTV worst-case gain optimization problems applying conic combinations of k IQCs defined by $IQC_k(\Psi_k, M_k)$ resulting in stacked IQCs:

$$\Psi = \begin{bmatrix} \Psi_1 \\ \vdots \\ \Psi_k \end{bmatrix} \text{ and } M(\lambda) = \begin{bmatrix} \lambda_1 M & & \\ & \ddots & \\ & & \lambda_k M \end{bmatrix} ; \ \lambda_1, ..., \lambda_k > 0 \tag{4.28}$$

as introduced in Example 3.7.1. Consequently, the design variables are $\lambda \in \mathbb{R}^k_+$, i.e. a vector consisting k strict positive real numbers arranged in form of $M(\lambda) \in \mathcal{M}$ describing the IQC parameterization. Conic combinations of IQCs cover analysis of a single dynamic uncertainties, time delays (see, e.g. [230]) or sector nonlinearities (see, e.g. [231]). The search procedure is specifically designed to work well with the arbitrarily large space, but also to work robustly under the nonlinear constraints introduced in Section 4.6.1 and exploit attained information in Section 4.6.2 on the LTV IQC analysis problem wherever possible.

Its basic search procedure is based on the original sine cosine algorithm (SCA) introduced in [232]. Similarly to most MHs, the Ab-SCA-PR contains three main steps, namely initialization, reproduction (based on sine and cosine functions), and selection phase. It is extended with an adaptive bound technique to deal with the large search space. Additionally, a population reduction is included to avoid extensive cost function evaluations late in the search. These are generally computationally expensive due to the bisection. In Algorithm 2, the Ab-SCA-PR's implementation is presented via pseudo-code.

Before the algorithm is executed, the user needs to provide the maximum

Algorithm 2 Ab-SCA-PR

```
Input: $N_{\text{p,max}}$, $N_{\text{i,max}}$, $n_\lambda$, $U_{\max}$, $\gamma_{\text{LB}}$, $\gamma_{\text{UB}}$, $\epsilon_{\text{BS}}$, $\lambda_{\max}$, $\lambda_u$, $G$, $\mathcal{M}$
Output: $\lambda_{\text{best}}$, $\gamma_{\text{best}}$
Generate random initial population $P$ and build respective $M(\lambda^l) \in \mathcal{M}$
Calculate $\gamma(M(\lambda^l))$ via bisection ($\gamma_{\text{LB}}/\gamma_{\text{UB}}$ fixed) constrained by solvability
  of the RDE over $[T, 0]$, for $P(T)$, treat $R \geq 0$ as failed integration
Find the best solution $\lambda_{\text{best}}$ and initialize $U = 0$
for $N_{\text{i}} = 1$ to $N_{\text{i,max}}$ do
    Calculate parameter $r_1$ based on (4.31)
    for $l = 1$ to $N_{\text{p}}$ do
        for $k = 1$ to $N_\lambda$ do
            Randomly generate the parameter $r_2$,$r_3$ and $r_4$ in the ranges of
              $[0, 2\pi]$, $[0, 2]$ and $[0, 1]$, respectively
            Update the $k^{th}$ element of the $l^{th}$ solution ($\lambda^l$) based on (4.30)
        end for
        Build $M(\lambda^l) \in \mathcal{M}$
        if RDE solvable for $\gamma^l_{N_{\text{i}}-1}$ and $M(\lambda^l)$ then Execute bisection with
          $\gamma_{\text{UB}} = \gamma^l_{N_{\text{i}}-1}$ calculating $\gamma(M(\lambda^l))$, handle $R \geq 0$ as failed
          integration
        else Skip bisection, treat $\lambda^l$ as failure
        end if
    end for
    Find $\lambda_{\text{best,new}}$
    if $\gamma(M(\lambda_{\text{best,new}})) < M(\gamma(\lambda_{\text{best}}))$ then $\lambda_{\text{best}} = \lambda_{\text{best,new}}$ and set $U = 0$
    else $U = U + 1$
    end if
    if $U > U_{\max}$ then Update search bounds via (4.32), reset $U$ to 0
        Generate $N_{\text{add}}$ solutions in $\mathcal{M}$ using LHS for new bounds
        Remove all solutions located in the old bounds
        Apply the $k$-mean clustering technique to group the remaining
          solutions into $N_{\text{add}}$ groups and find the centroid solutions
        Calculate $\gamma$ values of the centroid solutions of each group via
          bisection ($\gamma_{\text{LB}}/\gamma_{\text{UB}}$ fixed) and save to current population if they
          are better than the worse solution in the population
    end if
    Update population size via (4.33) and remove worst solutions from $P$
end for
```

population size $N_{\text{p,max}}$, the maximum number of population iterations $N_{\text{i,max}}$, the number of decision variables n_λ, maximum number of unsuccessful reproductions $U_{\max}$, lower and upper bound of the bisection γ_{LB} and γ_{UB}, respectively, the bisections absolute tolerance ϵ_{BS}, the vector λ_{u} with the initial upper bounds of the search space, the vector $\lambda_{\max}$ containing the maximum upper bounds of the search space, the extended LTV system G containing the fixed

IQC filter Ψ, and last the nonlinear constraint of the search space $\mathcal{M}$ providing the structure and properties of $M(\lambda)$. The estimates for γ_{LB} and γ_{UB} are chosen based on the recommendations in section 4.6.2.

The main algorithm starts with generating a random initial population P. It describes a set of l solution vectors $\lambda^l \in \mathbb{R}_+^{n_\lambda}$ written as:

$$P = \{\lambda^1, \lambda^2, ..., \lambda^{N_{\text{p,max}}}\}. \tag{4.29}$$

The elements of λ^l build the respective IQC paramterization $M(\lambda)$. Hence, it must be assured that $M(\lambda^l) \in \mathcal{M}$. In case of the IQC parameterization given in Example 3.7.1 the elements of λ need to be strict positive scalars. Hence, it is sufficient to define the search space for each element of λ^l as $\lambda_k^l \in (0, \lambda_{\text{max},k}]$, where $\lambda_{\text{max},k}$ is a sufficiently high upper limit. While an initial λ_{max} has to be specified, it will be adapted during the search if necessary. Hence, the optimization is not confined to the initial search space. This allows for a narrowed initial search space, e.g. exploiting information from previous optimizations as described in Section 4.6.2. Furthermore, it increases the algorithm's applicability as the poorly chosen initial bounds are compensated. This bound adaptation is motivated by Example 4.5.1. The comparison of Fig. 4.5 and 4.6 indicates that the search space's lower bound is too high for the first analysis, whereas the upper bound is too high for both analyses.

After it is guaranteed that $M(\lambda) \in \mathcal{M}$, the minimal value of γ for each λ^l in the initial population $\gamma(M(\lambda^l))$ is calculated using the bisection constrained by the solvability of the RDE (4.6) or (4.17), as described by Algorithm 1. The bisections for the $M(\lambda)$ are fully parallelized, i.e. if executed the on a multicore processor the bisections are distributed to all available cores. Due to the RDE's finite escape time it is possible that for a given $M(\lambda^l)$ no γ value can be calculated, as the RDE is never fully solvable. In this case, $\gamma(M(\lambda))$ is set to 10^{20}. The $R < 0$ constraint is included as described in Section 4.6.1. This means if $R(M(\lambda^l), \gamma)) \geq 0$, the respective γ bisection step is treated similarly to a not fully integrable RDE.

Then, the present best solution λ_{best} is identified. Now, the iteration starts with the reproduction process updating each design solution in the population via

$$\lambda^{l}_{\text{new},k} = \begin{cases} \lambda^{l}_{\text{old},k} + r_1 \sin(r_2) \left| r_3 \lambda_{\text{best},k} - \lambda^{l}_{\text{old},k} \right|, & \text{if } r_4 < 0.5 \\ \lambda^{l}_{\text{old},k} + r_1 \cos(r_2) \left| r_3 \lambda_{\text{best},k} - \lambda^{l}_{\text{old},k} \right|, & \text{otherwise} \end{cases}, \tag{4.30}$$

where $\lambda^{l}_{\text{new},k}$, $\lambda^{l}_{\text{old},k}$, and $\lambda_{\text{best},k}$ are the k^{th} vector element of a newly formed solution for λ^{l}, a present solution and the present best solution of the population, respectively. The parameters r_2, r_3, and r_4 are uniformly randomized for each iteration in the intervals of $[0, 2\pi]$, $[0, 2]$ and $[0, 1]$, respectively. The parameter r_1 is an iterative adaption applying

$$r_1 = a - N_{\text{i}} \frac{a}{N_{\text{i,max}}}, \tag{4.31}$$

where N_{i} is the present iteration and a is a predefined constant. Again, it has to be guaranteed that the updated $M(\lambda^{l}_{\text{new}}) \in \mathcal{M}$. As $M(\lambda^{l}_{\text{new}}) \notin \mathcal{M}$ is a simple boundary infraction for parameterizations described by (4.28), the respective $\lambda^{l}_{\text{new},k}$ are set to their nearest boundary value.

Before the bisection for the updated $M(\lambda^{l}_{\text{new}})$ is conducted, it is checked if the RDE (4.6)/(4.17) is solvable for $M(\lambda^{l}_{\text{new}})$ and the minimal $\gamma^{l}_{N_{\text{i}}-1}$ value calculated for its respective parent. If the RDE is not solvable, the bisection is omitted by the reasoning in Section 4.6.2 and a fitness of 10^{20} is assigned to the respective λ^{l}_{new}.

If the RDE (4.6)/(4.17) is fully solvable for $M(\lambda^{l})$ and $\gamma^{l}_{N_{\text{i}}-1}$, the bisection is executed using $\gamma^{l}_{N_{\text{i}}-1}$ as the upper bound γ_{UB}. Using this adaptive upper bound, significantly narrows the bisection interval compared to the initial population's or the one used after a boundary extension, which utilize user defined bounds.

After concluding all bisections, the best solution obtained from the newly created population is compared with its complement from the preceding iteration. If the new population's best solution is an improvement, the current best solution is updated, while U is reset to zero. Here, U is a variable that counts the number of unsuccessful iterations. Otherwise, the current best solution remains unchanged, and U is increased by one. If the U value is higher than a predefined limit, the search space's bounds are extended. Given the IQC parameterization in (4.28), each design variable's upper bound is extended by

$$\lambda_{\mathrm{u,new}_k} = \begin{cases} 10\lambda_{\mathrm{u,old}_k} & \text{if } \lambda_{\mathrm{u,old}_k} < \lambda_{\mathrm{max}_k} \\ \lambda_{\mathrm{u,old}_k} & \text{otherwise} \end{cases} , \tag{4.32}$$

whereas the lower bound remains zero. In (4.32), $\lambda_{u,\mathrm{old}_k}$ and $\lambda_{u,\mathrm{new}_k}$ are the upper bound of the k^{th} element of all design variables λ^l before and after updating, respectively. The maximum admissible upper bounds of the corresponding elements are given by λ_{max_k}. Note that this boundary adaption can also be adjusted to cover other search spaces.

Since the bounds of the design variables have been extended, a set of additional $N_{\mathrm{add}} = 10$ solutions located on/inside the extended boundary must be generated to enhance the optimizer's search performance. In order to have the solutions well distributed throughout the extended boundary, a Latin Hypercube Sampling (LHS) technique is first used to create $50 \cdot N_{\mathrm{add}}$ solutions throughout the whole boundary of the design variables [233]. Then, all solutions inside the old boundary are removed, while N_{add} solutions are created based on the distribution of the remaining solutions. Here, a k-mean clustering technique is used to group the remaining solutions into N_{add} groups, whereas each group's centroid is assigned as one of those N_{add} solutions [234, 235]. After calculating their fitness values, the solutions are added to the present population. Then, the worst solutions in the population are deleted to recover the required population size.

After obtaining the current best solution of γ and updating the search space for each iteration, the population size is reduced based on the following equation

$$N_{\mathrm{p}_{N_\mathrm{i}+1}} = N_{\mathrm{p,max}} - \mathrm{round}\, \frac{N_\mathrm{i}\left(N_{\mathrm{p,max}} - N_{\mathrm{p,min}}\right)}{N_{\mathrm{i,max}}}, \tag{4.33}$$

where $N_{\mathrm{p}_{N_\mathrm{i}+1}}$ is the population size at iteration $N_\mathrm{i} + 1$. $N_{\mathrm{p,max}}$ and $N_{\mathrm{p,min}}$ are the user-defined maximum and minimum population sizes, respectively.

In order words, the presented algorithm starts with the maximum population size at the first function evaluation and reduces the population size with progressing function evaluations. If $N_{\mathrm{p}_{N_\mathrm{i}+1}}$ is lower than the current population size, the current population's worst solutions are removed to match the new population size. The population reduction reduces the amount of necessary bisection evaluation towards the end of the search when the global minimum's neighborhood is likely identified. Simultaneously, it further biases the search continuously towards this neighborhood, possibly improving the convergence. Subsequently, the reproduction starts again. The search process ends as soon as the maximum number of iterations $N_{\mathrm{i,max}}$ is reached, providing the minimal calculated upper bound γ_{best} on the worst gain and the corresponding λ_{best}.

4.6.4 Log-L-SHADE

The second algorithm developed in this thesis is a novel logarithmically scaled self-adaptive differential evolutionary optimizer with linear population size reduction (Log-L-SHADE). It is custom-tailored to efficiently deal with the optimization problem (4.23) for more complex IQC parameterizations such as parametric uncertainty given in Example 4.6.1 and defined by $IQC(\Psi, M)$:

$$\Psi = \begin{bmatrix} b\psi_\nu \otimes I_{n_v} & 0 \\ 0 & \psi_\nu \otimes I_{n_v} \end{bmatrix} \text{ and } M(X,Y) = \begin{bmatrix} X & Y \\ Y^T & -X \end{bmatrix} := (m_{i,j}), \tag{4.34}$$

with $\psi_\nu = \begin{bmatrix} 1 & \frac{s+\rho}{s-\rho} & \cdots & \frac{(s+\rho)^\nu}{(s-\rho)^\nu} \end{bmatrix}^T$. Consequently, the design variables are the elements m_{ij} of the IQC parameterization M, on which $X = X^T > 0$ and

$Y = -Y^T$ imposes nonlinear constraints, as described in Section 4.6.1. This optimization problem is also significantly larger than the conic combinations, as the matrix variable X contributes $\frac{n_v(\nu+1)(n_v(\nu+1)+1)}{2}$ decision variables and Y $\frac{n_v(\nu+1)(n_v(\nu+1)-1)}{2}$ decision variables. In the case that the uncertainty is repeated three times, already, the selection of a relatively low MacMillan degree $\nu = 1$ for ψ_ν cumulates in thirty-six decision variables. Here, decision variables related to diagonal entries of X are defined over $\mathbb{R}^+$, whereas the ones related to off-diagonal entries in either X or Y are defined over $\mathbb{R}$. Thus, the resulting (individual) search spaces are arbitrarily large, i.e. cover several orders in magnitude. Recalling Example 4.5.1 and especially Fig. 4.4 and 4.5, the change in γ for a variation of the IQC parametrization $M(\lambda)$ over several magnitudes is significantly better covered by a logarithmically spaced grid and represented using a semi-logarithmic plot, respectively. Given a logarithmic scaling of the x-axis, the problem also appears (locally) convex. Consequently, searching a logarithmic rather than a decimal scale exploits this observation and allows the meta-heuristic to converge easier, especially over a search space spanning several magnitudes. A single design variable m_{ij} is represented using a logarithmic search space by:

$$\begin{aligned} &m_{ij} = (-1)^{\text{round}(m_{\text{sign},ij})} 10^{m_{\text{exp},ij}}, \text{with} \\ &m_{\text{sign},ij} \in \mathbb{R}_0^+ \leq 1, m_{\text{exp},ij} \in \mathbb{R}, \end{aligned} \tag{4.35}$$

where $m_{\text{sign},ij}$ and $m_{\text{exp},ij}$ are the elements of the matrix $M_{\text{sign}} \in \mathbb{S}^{n_m}$ and $M_{\text{exp}} \in \mathbb{S}^{n_m}$, respectively. In the proceeding, they will be arranged in the form of a block-diagonal matrix $M_{\text{Log}} = \text{diag}(M_{\text{exp}}, M_{\text{sign}}) \in \mathbb{S}^{2n_m}$ with elements $m_{\text{Log},ij}$. Therefore, the meta-heuristic searches over the new decision matrix M_{Log}.

The Log-L-SHADE's underlying search procedure is based on the original L-SHADE proposed in [236]. As the Ab-SCA-PR, it contains three main steps, namely initialization, reproduction, and selection phase. It is extended with the introduced logarithmic search space to deal with the complicated and extensive search space. These are generally computationally expensive due to the bisection. In Algorithm 3, the Log-L-SHADE's implementation is illustrated using pseudo-code.

Algorithm 3 Log-L-SHADE

```
Input: $N_{\mathrm{p,max}}$, $N_{\mathrm{p,min}}$, $N_{\mathrm{i,max}}$, $k_{CR}$, $k_F$, $G$, $\mathcal{M}$, $m$, $n$, $\gamma_{\mathrm{lim}}$, $M_{\mathrm{Log,init}}$, $k_{\mathrm{IQC}}$,
    $\gamma_{\mathrm{LB}}$, $\gamma_{\mathrm{UB}}$, $\epsilon_{\mathrm{BS}}$
Output: $\gamma_{\mathrm{best}}$, $M_{\mathrm{Log,best}}$
Initialize: $S_F$, $S_{CR}$
Scale original IQC norm bound by $k_{\mathrm{IQC}}$
while Amount of $M^l$ with $\gamma(M^l) < 10^{20}$ less than $0.2N_{\mathrm{p,max}}$ do
    Generate random initial population $P_{\mathrm{Log}}$, convert to decimal domain
        via (4.35), and guarantee $M^l \in \mathcal{M}$, assign $\gamma(M^l) = 10^{20}$ in case
        $\mathrm{cond}(R(M^l, \gamma_{\mathrm{UB}}, t)) > 10^{12}$
    Calculate remaining $\gamma(M^l)$ via bisection ($\gamma_{\mathrm{LB}}/\gamma_{\mathrm{UB}}$ fixed) constrained
        by the solvability of the RDE over $[T, 0]$ given $P(T)$, treat $R \geq 0$
        as failed integration
end while
Find current best solution $M_{\mathrm{Log,best}}$ and fitness $\gamma_{\mathrm{best}}$
Set IQC norm bound upscaling threshold $N_{\mathrm{p,IQC}} = N_{\mathrm{p,max}}$ and $N_{\mathrm{i}} = 0$
while ($N_{\mathrm{i}} \leq N_{\mathrm{i,max}}$ OR $\gamma_{\mathrm{best}} > \gamma_{\mathrm{lim}}$) AND $k_{\mathrm{IQC}} < 1$ do
    $N_{\mathrm{i}} = N_{\mathrm{i}} + 1$
    if $k_{\mathrm{IQC}} < 1$ AND ($\gamma_{\mathrm{best}} \leq \gamma_{\mathrm{lim}}$ OR $N_p < 0.8N_{\mathrm{p,IQC}}$) then
        Set $N_{\mathrm{p,IQC}} = N_{\mathrm{p}}$ and $k_{\mathrm{IQC}} = \min(3k_{IQC}, 1)$
        Upscale norm bound, recalculate $\gamma^l$ with original $\gamma_{\mathrm{UB}}$, and update
            $M_{\mathrm{Log,best}}$ and $\gamma_{\mathrm{best}}$
    end if
    for $l = 1$ to $N_{\mathrm{p}}$ do
        Compute $\bar{M}^l_{\mathrm{Log}}$ (4.37), mutate $\bar{m}^l_{\mathrm{Log},ij}$ (4.38), and enforce boundaries
        Calculate $\bar{M}^l$ via (4.35) and guarantee $\bar{M}^l \in \mathcal{M}$
        if RDE is solvable for $\bar{M}^l$ and $\gamma(M^l)$ AND $\mathrm{cond}(R) < 10^{12}$ then
            Execute bisection with $\gamma_{\mathrm{UB}} = \gamma(M^l)$ calculating $\gamma(\bar{M}^l)$,
                handle $R \geq 0$ as failed integration
        else
            Skip bisection, treat corresponding $M^l_{\mathrm{Log}}$ as failure
        end if
        if $\gamma(\bar{M}^l) < \gamma(M^l)$ then $M^l_{\mathrm{Log}} = \bar{M}^l_{\mathrm{Log}}$
        end if
    end for
    Update $S_F$ and $S_{CR}$ with $\mu_F$ and $\mu_{CR}$ calculated via
        (4.39)-(4.41) using successful $F$ and $CR$
    Identify current best solution $M_{\mathrm{Log,best}}$ and fitness $\gamma_{\mathrm{best}}$
    Update population size via (4.42) and remove worst solutions from $P_{\mathrm{Log}}$
end while
```

Before the algorithm is executed, a total of sixteen inputs must be provided by the user. The first three are similar to the Ab-SCA-PR, namely the maximum and minimum population size $N_{\text{p,max}}$ and $N_{\text{p,min}}$, respectively, and the maximum number of population iterations $N_{\text{i,max}}$. These are followed by the number of successful crossover rates and scaling factors k_{CR} and k_F, respectively. G is the extended system in (4.2), which includes the user-selected IQC filter Ψ. It is followed by $\mathcal{M}$ describing the set of feasible IQC parameterizations. The inputs m and n, with $m \in \mathbb{N}^{n_m(n_m+1)}$ and $n \in \mathbb{N}^{n_m(n_m+1)}$ define the minimum and maximum value of the elements in M_{Log}. Furthermore, the user can provide a scaling factor k_{IQC} for the uncertainty norm bound b covered by the IQC to accelerate the search of an initial population, see Section 4.6.2. The input γ_{lim} is used as rescaling and terminal condition related to the the worst-case gain's present best optimization value γ_{best}. A guess for the initial population can be provided by $M_{\text{Log,init}}$. Remaining are three inputs that are required to run the bisection, its lower and upper bound γ_{LB} and γ_{UB}, respectively, and its relative tolerance ϵ_{BS}.

The algorithm is initialized with the vectors $S_F \in \mathbb{R}^{k_F}$ and $S_{CR} \in \mathbb{R}^{k_{\text{CR}}}$ containing k_F and k_{CR} elements, respectively, with a value of 0.5. These vectors are later used to store successful weighted contra harmonic mean values μ_F and μ_{CR} of successful scaling factors F and crossover rates CR, respectively.

The main algorithm starts with generating a random initial population P_{Log}. It describes a set of $N_{\text{p,max}}$ solution matrices M^l_{Log}, written as

$$P_{\text{Log}} = \{M^1_{\text{Log}}, M^2_{\text{Log}}, ..., M^l_{\text{Log}}\}, \tag{4.36}$$

whose coefficients $m^l_{\text{Log},ij}$ are confined to their respective bound constraints. The condition $X > 0$ requires the respective m^l_{ii} related to the diagonal entries/elements of X to be strictly positive. Therefore, the respective elements $m^l_{\text{sign},ii}$ in the logarithmic search domain can be fixed to zero, reducing it by $n_v(\nu + 1)$ design variables. In case an initial guess $M_{\text{Log,init}}$ is provided, the set

P_{Log} is extended with it.

Then, the initial population P_{Log} is converted into its decimal domain equivalent P_{dec} using (4.35), and it is checked if $M^l \in \mathcal{M}$. $X > 0$ violations are handled according to the remarks in Section 4.6.1 by increasing the corresponding diagonal elements m^l_{ii}. Accordingly, the corresponding elements $m^l_{\text{Log},ii}$ assigned to the element's magnitude in P_{Log} is updated. The poor conditioning of R discussed in Section 4.6.1 becomes prominent for larger IQC parameterizations, especially given a large initial search space. Therefore, a default upper bound of 10^{12} on $R(M^l, \gamma_{\text{UB}}, t)$'s condition number over the analysis horizon $[0, T]$ is proposed, which if violated is handled as proposed in Section 4.6.1, i.e. for the respective M^l the bisection is skipped and $\gamma(M^l) = 10^{20}$.

Now, the minimal $\gamma(M^l)$ related to M^l are calculated via bisection as described in Algorithm 1 and identically to the Ab-SCA-PR's implementation. The computation is fully parallelized, i.e. the number of accessible workers/processor cores is directly inverse to the computation time. For a better convergence of the optimization, a minimum of 20% successful elements in P_{dec} and, thus, P_{Log} are required, i.e. elements with $\gamma(M^l) < 10^{20}$. If the initial population does not contain enough valid members, invalid members are replaced with new random members until the condition is fulfilled. If the search of a valid initial set P_{Log} appears problematic, the perturbation's norm bound can be downscaled via k_{IQC}. By downscaling, the co-domain/feasibility set of M^l possessing a valid γ is extended. The approach is described and reasoned in Section 4.6.2. The the more generous feasibility set allowed Log-L-SHADE to converge faster in numerous test scenarios.

After identifying the current best solution $M_{\text{Log,best}}$ with its respective γ_{best}, the meta-heuristic iteration starts. Firstly, it is checked if the norm bound covered by the IQC can be upscaled. This is the case, if either $\gamma_{\text{best}} < \gamma_{\text{lim}}$ or $N_{\text{p}} < 0.8 N_{\text{p,IQC}}$, with $N_{\text{p,IQC}} = N_{\text{p,max}}$ for the first iteration. Both cases require that the current scaling k_{IQC} is smaller than 1. If the norm bound is upscaled, all $\gamma(M^l)$ in the population are recalculated, and the new best solution $M_{\text{Log,best}}$ and fitness γ_{best} are identified.

Now, the reproduction starts, and a new generation of the population is created, using the current-to-pbest/1 strategy from [236]. Therefore, potential new individuals $\bar{M}^l_{\text{Log}}$ of the population are calculated from current individuals M^l_{Log} via

$$\bar{M}^l_{\text{Log}} = M^l_{\text{Log}} + F_l(M^l_{\text{Log,pbestl}} - M^l_{\text{Log}} + M^l_{r1} - M^l_{r2}). \tag{4.37}$$

In (4.37), M^l_{r1} and M^l_{r2} are two randomly selected individuals from the current population, whereas $M^l_{\text{Log,pbest}}$ is a individual, randomly selected from the best 10% of the current population P_{Log}. In (4.37), the scaling factor $F_l \in \mathbb{R}$ is a Cauchy distributed random number with variance 0.1 and a mean value μ_{F,k_F}. The latter is a randomly selected element from the vector S_F. After performing the mutation, each of the elements $\bar{m}^l_{\text{Log},ij}$ in $\bar{M}^l_{\text{Log}}$ has a chance to be replaced with the respective element $m^l_{\text{Log},ij}$ of its parent M^l_{Log} by the means of binomial crossover:

$$\bar{m}^l_{\text{Log},ij} = \begin{cases} \bar{m}^l_{\text{Log},ij} & \text{if } randn[0,1) \leq CR_l \,\text{or}\, ij = ij_{\text{rand}} \\ m^l_{\text{Log},ij} & \text{otherwise} \end{cases}. \tag{4.38}$$

In (4.38), the crossover rate $CR_l \in \mathbb{R}$, with $l = 1, 2, ..., N_{\text{p}}$ is a normal distributed random number with variance 0.1 and a mean value $\mu_{CR,k_{CR}}$. The later is a randomly selected element from the vector S_{CR}. The index combination ij_{rand} is a random index combination, which prevents some elements to be updated besides the crossover rate being to low. After finishing the crossover, the bound constraints are checked. In case of a violation, the respective elements $\bar{m}^l_{\text{Log},ij}$ are set to the mean value of the corresponding parental element $\bar{m}^l_{\text{Log},ij}$ and the respective violated boundary. Then, the $\bar{M}^l_{\text{Log}}$ are transformed into their respective decimal representation $\bar{M}^l$ using (4.35). Due to the mutation and crossover, it is necessary to check again if the $\bar{M}^l$ are in $\mathcal{M}$. In case that, e.g. a constraint $X > 0$ is violated, a correction as described for the initial population is conducted. Subsequently, the inner loop of the bisection is executed to calculate the minimal $\gamma(\bar{M}^l)$ for each $\bar{M}^l$. The lower and upper bounds are chosen equivalently to the Ab-SCA-PR as recommended in Section 4.6.2. Here, the same procedure of skipping bisections as described in Section 4.6.2 and utilized in the Ab-SCA-PR is applied.

If $\gamma(\bar{M}^l) < \gamma(M^l)$, then the respective $\bar{M}^l_{\text{Log}}$ replaces M^l_{Log} in P_{Log}. Otherwise, M^l_{Log} remains unchanged. The F_o and CR_o with $o = 1, 2, ..., n_{\text{opt}}$ used to create the n_{opt} improved $\bar{M}^o$ are used to update the first element in S_F and S_{CR}, respectively. Therefore, a new μ_{CR} and μ_F is calculated as:

$$\mu_{CR} = \frac{\sum_{o=1}^{n_{\text{opt}}} \eta_o CR_o^2}{\eta_o CR_o}, \tag{4.39}$$

$$\mu_F = \frac{\sum_{o=1}^{n_{\text{opt}}} \eta_o F_o^2}{\sum_{o=1}^{n_{\text{opt}}} \eta_o F_o}, \tag{4.40}$$

with

$$\eta_o = \frac{\left|\gamma(\bar{M}_o) - \gamma(M_o)\right|}{\sum_{o=1}^{n_{\text{opt}}} \left|\gamma(\bar{M}_o) - \gamma(M_o)\right|}. \tag{4.41}$$

These are weighted contra-harmonic mean values, a special form of the weighted Lehmer mean, see e.g. [237]. Note that in the next population iteration, the successful sets' subsequent elements are updated until k_{CR}/k_F updates were executed. Afterwards, the updates start again with the first elements in the sets.

Before the next iteration starts, the population size $N_{\text{p}_{N_\text{i}+1}}$ of the next iteration $N_\text{i} + 1$ is updated by

$$N_{\text{p}_{N_\text{i}+1}} = \text{round}\left(N_{\text{p,max}} - \frac{N_\text{i}(N_{\text{p,max}} - N_{\text{p,min}})}{N_{\text{i,max}}}\right). \tag{4.42}$$

If $N_{\text{p}_{N_\text{i}+1}}$ is smaller than the present population size, the $N_{\text{p}_{N_\text{i}}} - N_{\text{p}_{N_\text{i}+1}}$ worst excessive solutions in P_{Log} are removed to match the updated population size. In case the scaling factor k_{IQC} is less than 1, before the next population update

it is evaluated if $\gamma_{\text{best}} \leq \gamma_{\text{lim}}$ or $N_{\text{p}} < 0.8N_{\text{p,IQC}}$. If so the uncertainty norm-bounds are upscaled by a factor three up to a maximum value of one and the helper variable $N_{\text{p,IQC}}$ is set to the value of the present population size. Subsequently, the population is fully re-evaluated, i.e. using the initial γ_{UB} and the without skipping any bisections and the new γ_{best} and corresponding $M_{\text{log,best}}$ are identified.
The optimization concludes, as soon as the maximum number of population iterations $N_{\text{i,max}}$ or $\gamma_{\text{best}} < \gamma_{\text{lim}}$, if $k_{\text{IQC}} = 1$. It returns γ_{best} and M_{best} corresponding to the optimal solution $M_{\text{Log,best}}$ of the latest iteration.
Log-L-SHADE is also suitable for problems with mixed perturbations, which require the stacking of multiple different IQC ”types” and, thus, present a mix of the parameterizations introduced so far. Hence, it presents the algorithm of choice for the elaborate industry examples in Chapters 5 and 6.

4.7 Benchmark Example of Ab-SCA-PR

In this section, the optimization problem (4.23) arising in an industry-relevant benchmark example is solved using the Ab-SCA-PR algorithm proposed in Section 4.6.3 and thirteen existing MHs. This benchmark example is taken from [238], where a small space launcher's robust tracking performance under wind disturbance is analyzed.

4.7.1 Model of the Vanguard Space Launcher

The analyzed dynamics represent the Vanguard space launcher's first stage during the atmospheric flight phase. In the vertical plane, the launcher follows a pitch program, i.e. the launcher tracks a time-scheduled pitch angle signal θ_d calculated preflight. Equivalently to the motivational example in Section 2, the trajectory describes a so-called gravity turn maneuver with a nominal angle of attack α of zero. A linear time-varying representation G of the space launcher's nominal dynamics is given in [108]. It results from the nonlinear

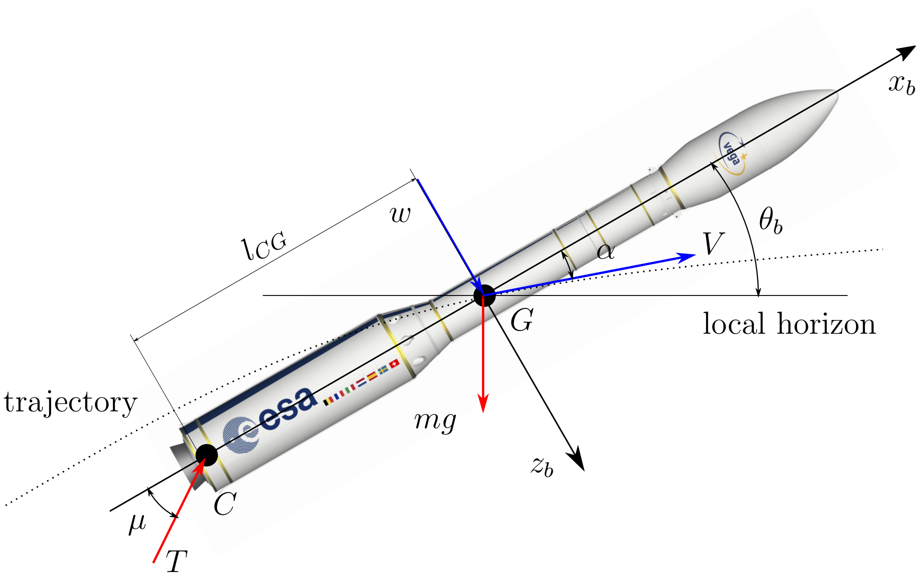


Figure 4.10: *Space launcher schematic*

dynamics' linearization along the gravity turn trajectory, with respect to the body-fixed coordinate system and is defined by:

$$\begin{bmatrix} \dot{\alpha}(t) \\ \dot{\theta}_b(t) \\ \ddot{\theta}_b(t) \end{bmatrix} = \begin{bmatrix} Z_\alpha(t) & -\frac{g_0(t)}{V(t)} & 1 \\ 0 & 0 & 1 \\ \frac{M_\alpha(t)}{Jy(t)} & 0 & \frac{M_q(t)}{J_y(t)} \end{bmatrix} \begin{bmatrix} \alpha(t) \\ \theta_b(t) \\ \dot{\theta}_b(t) \end{bmatrix} + \begin{bmatrix} \frac{T(t)}{m(t)V(t)} & \frac{Z_\alpha(t)}{m(t)V(t)} \\ 0 & 0 \\ \frac{T(t)l_{CG}(t)}{J_y(t)} & \frac{M_\alpha(t)}{J_y(t)} \end{bmatrix} \begin{bmatrix} \mu(t) \\ \delta_\alpha(t) \end{bmatrix} \tag{4.43}$$

$$\begin{bmatrix} \alpha(t) \\ \theta_b(t) \\ \dot{\theta}_b(t) \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \alpha(t) \\ \theta_b(t) \\ \dot{\theta}_b(t) \end{bmatrix} + \begin{bmatrix} 0 & 1 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \mu(t) \\ \delta_\alpha(t) \end{bmatrix}$$

The nominal LTV model's states are the angle of attack α, the pitch angle θ_b, and the pitch rate $\dot{\theta}_b$. It has two inputs, the TVC deflection μ utilized for pitch control and a wind disturbance in the form of an additional exogenous angle of attack signal δ_α. The latter is described by $\delta_\alpha \approx \frac{-w}{V}$ and defined parallel to the launcher-fixed frame's z_b-axis, as depicted in Fig. 4.10. V is the velocity of the space launcher, and w is the wind speed. Denoted by Z_α, M_α, and M_q are the aerodynamic stability derivatives. The variables m and J_y denote the launcher's total mass and overall mass moment of inertia. The latter is stated with respect to the launcher's center of gravity G. By T, the thrust is denoted, which acts at the nozzle reference point C. The geometric variable l_{CG} describes the absolute distances between C and G. Assuming an equatorial launch site, the gravitational acceleration g_0 is modeled according to the world geodetic system 84 (WGS84 [239]). The numerical values of all introduced variables are available in [108] in the form of tables, plots over time, or functions of time.

4.7.2 Analysis Interconnection

In Fig. 4.11, the corresponding analysis interconnection is displayed. Here, the block G_{LV} represents the space launcher's LTV dynamics described by (4.43). Thrust vector control (TVC) is applied to control the launcher's pitch attitude. Its dynamics are described by

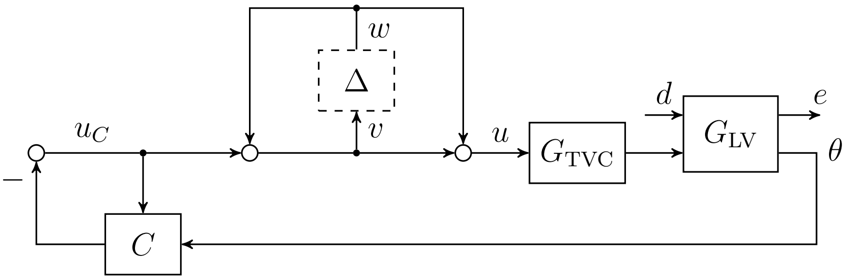


Figure 4.11: *Analysis interconnection used for disk-margin analysis*

$$G_{\text{TVC}} = \frac{50}{s+50} \tag{4.44}$$

and represented by the block G_{TVC}. A linear quadratic regulator (LQR) including an observer based on pitch angle feedback calculates the respective control signals. Their constant gains are calculated in accordance to [108] for the dynamic at 48s after lift-off. The controller is represented by C in Fig. 4.11. A norm bounded dynamic LTI uncertainty Δ is introduced into the interconnection, to evaluate the system's robustness regarding to simultaneous gain and phase variations/perturbations. The uncertainty's specific implementation as shown in Fig. 4.11 mimics input LTI disk margins. These are a common tool in the robustness analysis of flight control systems, e.g. [240]. In this case, the norm bound b directly relates to the applied simultaneous phase and gain disturbances' maximum value. Thus, by analyzing the interconnection in Fig. 4.11 for increasing b values, the maximal sustainable perturbation can be identified. In total, eight values for b ranging from 0.01 to 0.085 are evaluated. The worst case finite time induced $L_2[0,T]$ gain γ from the wind disturbance $d = \delta_\alpha$ to the angle of attack $e = \alpha$ is applied to quantify the launcher's performance.

Therefore, the analysis interconnection in Fig. 4.11 must be converted into the LTV IQC framework as detailed in Section 4.3. The dynamic LTI uncertainty Δ's input/output behavior is covered by the conic combination of two IQCs following Example 3.7.1. By $IQC_1(\Psi_1, M_1)$ the first IQC is denoted, which is factorized with $\Psi_1 = I_2$ and parameterized by $M_1(\lambda_1) = \lambda_1 M$, where $M = \left[\begin{smallmatrix} b^2 & 0 \\ 0 & -1 \end{smallmatrix}\right]$. The second IQC, namely $IQC_2(\Psi_2, M_2)$, applies $\Psi_2 = \frac{1}{s+1} I_2$ as factorization and $M_2(\lambda_2) = \lambda_2 M$ as parameterization. Both scalings, λ_1 and λ_2, are defined as strict positive. By stacking $IQC_1(\Psi_1, M_1)$ and $IQC_2(\Psi_2, M_2)$ as in (4.28), a single IQC is created equivalently to Example 3.7.1, with factorization $\Psi = [\Psi_1^T, \Psi_2^T]^T$ and parameterization $M(\lambda) = \left[\begin{smallmatrix} \lambda_1 M & \\ & \lambda_2 M \end{smallmatrix}\right]$.

Consequently, the LTV worst case gain optimization problem (4.23) identifying γ must be solved over the two decision variables λ_1 and λ_2. The optimization is repeated for increasing b values. Recall, that due to the finite escape time of the underlying RDE, it is possible that some solvers cannot find a valid γ for all the test cases executed in the benchmark example.

4.7.3 Solver and Benchmark Setup

The proposed Ab-SCA-PR algorithm is evaluated against a total of thirteen MHs. Each MHs solves the space launcher's worst-case gain optimization in five independent runs. All solvers start with an initial population size of 50. Solvers with fixed population size are terminated after 50 generations, whereas algorithms with adaptable population size terminate after 2500 (50×50) bisection runs. Except for the Ab-SCA-PR, all optimizers apply a fixed lower and upper bound of $1 \cdot 10^{-6}$ and $1 \cdot 10^{8}$, respectively, for both λ_1 and λ_2. However, the Ab-SCA-PR algorithm's self-adaptive upper bounds allow for a significantly narrower initial search space. Therefore, the initial upper bound for both decision variables is reduced to 100.

All the evaluated meta-heuristics with their specific optimization parameter settings are listed below.

1. Differential evolution (DE) [175]: DE/best/2/bin strategy was used, with a scaling factor, crossover rate and probability of choosing elements of mutant vectors of 0.5, 0.7, and 0.8, respectively.

2. Adaptive differential evolution (JADE) [241]: All optimization parameters are self-adapted during an optimization run.

3. Success-history based adaptive differential evolution (SHADE) [242]: All optimization parameters are self-adapted during an optimization run.

4. SHADE with Linear Population Size Reduction (L-SHADE) [236] : All optimization parameters are self-adapted during an optimization run.

5. Neuro-dynamic Differential Evolution Algorithm (L-SHADE-ND) [243]: All optimization parameters are self-adapted during an optimization run.

6. L-Shade with Eigenvector-Based Crossover and Successful-Parent-Selecting Framework (SPS-L-SHADE-EIG) [244]: All optimization parameters are self-adapted during an optimization run.

7. Whale optimization algorithm (WOA) [245]: The algorithm's authors provide both, code and parameters.

8. Moth-flame optimization algorithm (MFO) [166]: The algorithm's authors provide both, code and parameters.

9. Dragonfly Algorithm (DA) [167]: The default parameter setting from the original code by [167] are used in this benchmark.

10. Grey Wolf Optimizer (GWO) [169]: The default parameter setting from the original code by [169] are used in this benchmark.

11. Sine Cosine algorithm (SCA) [232] (Algorithm 1): The constant parameter a is set to 2.

12. Improved sine cosine algorithm with crossover scheme (ISCA) [246]: The constant parameter a is set to 2, while the crossover rate is set to 0.3.

13. Modified Sine Cosine Algorithm (m-SCA) [247]: The constant parameter a, crossover rate and jumping rate are set to 2, 0.3 and 0.1, respectively.

14. Adaptive boundary sine cosine optimizer with population reduction (Ab-SCA-PR) (Algorithm 1): Used the same parameter settings as SCA.

Note, all optimizations apply two important recommendations from Section 4.6.2, namely the parent fitness based bisection upper bound and to skip the bisection completely, if the offspring promises no improvement. Only the initial population is evaluated fully using a fixed specific upper bound γ_{UB}. The lower bound γ_{LB} for a given b is always fixed. The respective b-specific values are taken from [238] and presented in Tab. 4.1. Note that for $b = 0.01$, γ_1 equals the nominal worst case gain and γ_{UB} is a factor of ten higher. The subsequent norm bounds use a γ_{LB} and γ_{UB} of 0.8 and 10 times the worst case γ of the previously evaluated norm bound, respectively.

Table 4.1: Lower bound γ_{LB} and upper bound γ_{UB} used for the bisection for a given b based on [238]

Norm bound b	Lower bound γ_l	Upper bound γ_u
0.01	1.1527	11.527
0.03	1.8837	23.5460
0.05	2.2529	28.1610
0.06	2.8194	35.2420
0.07	3.2229	40.2860
0.075	3.7438	46.7970
0.08	4.2086	52.6070
0.085	5.3578	66.9730

4.7.4 Results and Discussion

Four metrics are applied to evaluate the MHs' search performance:

1. The lowest cost function value γ_{best}.

2. The worst-case gain's mean value μ_γ over the five optimization runs.

3. The number of successful runs n_{feas}.

4. The worst-case gain's standard deviation σ_γ.

Firstly, based on the lowest cost function value γ_{best}, the absolute search performance. Secondly, the worst-case gain's mean value μ_γ over the five optimization runs. It is used to measure the convergence rate and consistency of the algorithms. Both indicate the algorithm's reliability, which essential for its industrial applicability. The third criterion, the number of successful runs n_{feas}, further emphasizes this. In the case that two algorithms deliver the same number of successful runs, the worst-case gain's standard deviation σ_γ is used to measure the search consistency instead. It should be noted that only algorithms that can find feasible solutions in at least two optimization runs are considered for the μ_γ and σ_γ value comparison. In Fig. 4.12, the worst-case gain's mean value and variance over b achieved by the Ab-SCA-PR are shown. These values are compared to the two existing optimizers with the most successful runs, namely the GWO and the SCA. The proposed algorithm achieved

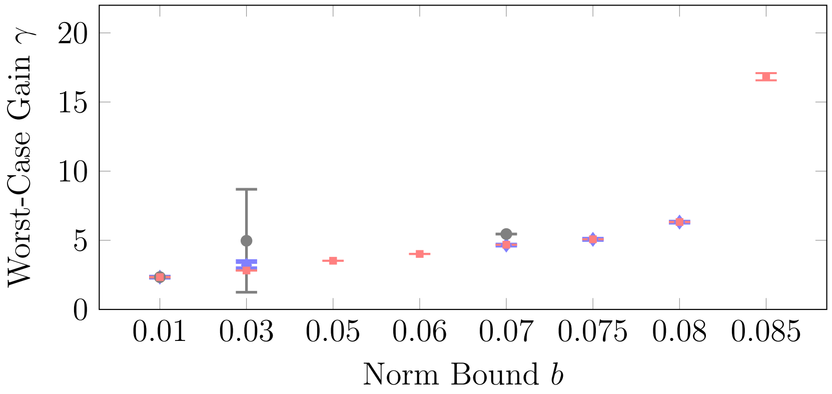


Figure 4.12: *Mean values and variance of the top three optimizers: Ab-SCA-PR (▪), GWO (♦), SCA (●)*

the lowest variance σ_γ in the worst-case gain for all b values. Furthermore, the two existing MHs did not provide multiple solutions, i.e. successful optimization runs, for all values of b. They also show a higher variance as well as higher mean values μ_γ. In particular, the SCA showed significantly worse results for $b = 0.03$ and $b = 0.07$. Hence, the proposed Ab-SCA-PR is the most reliable and consistent optimizer for the LTV robustness analysis problem.

By evaluating each algorithm's detailed results summarized in Tab. 4.2, the Ab-SCA-PR superiority can be further underlined. Note, the results for the remaining b are given in Table B.2 in appendix B. Regarding absolute search performance, the proposed Ab-SCA-PR is the best optimizer for norm bounds of 0.03, 0.05, and 0.085. However, for the lowest norm bound $b = 0.01$, it achieves only fifth place, with the LSND calculating the lowest γ_{best}. The proposed algorithm achieves the third-lowest γ analyzing $b = 0.07$. Nevertheless, the Ab-SCA-PR's γ_{best} is always equivalent to at least the second decimal achieved by the respective best algorithm.

Concerning search convergence, the proposed Ab-SCA-PR is the best performer for the cases of $b = 0.03$, $b = 0.05$, and $b = 0.085$ and is also the runner-up for $b = 0.07$. Evaluating $b = 0.01$, the best MHs in this category are the GWO and the WOA. Analyzing a norm bound of 0.07, the GWO is the best optimizer. The runner-ups for $b = 0.03$ and $b = 0.05$ are the GWO and the LSHADE-ND, respectively. The third best algorithm given $b = 0.01$ and

$b = 0.03$ is the SCA, while the third-best method for $b = 0.07$ is the WOA. Regarding the search consistency, the best performer across all norm bounds is the proposed Ab-SCA-PR with a 100% success rate. All remaining algorithms' search consistency deteriorates for increasing norm bound values b. For the lowest norm bound of 0.01, the overall results are still good, and five algorithms, namely SCA, DA, GWO, WOA, and mSCA, reach a 100% success rate. However, none of these algorithms achieved a standard deviation as low as the Ab-SCA-PR ($\sigma_\gamma = 0.0004$). The SCA achieved the second lowest standard deviation$\sigma_\gamma = 0.007$, and DA the highest with $\sigma_\gamma = 0.6704$. Two other algorithms, the LSND and the ISCA, achieved four successful runs, with the ISCA performing worse overall, achieving a ten times higher standard deviation. The only other algorithm concluding multiple successful runs was the DE, totaling 2. A total of five algorithms fail to identify a valid γ in any run. Raising b to 0.03, other than the proposed algorithm, only the GWO, and the SCA have more than one successful run, with four and three, respectively. However, their standard deviations of 0.2581 and 3.7230, respectively, are considerably worse than the Ab-SCA-PR's 0.0013. Other than that, just the WOA can identify a solution at all for this norm bound. By further increasing the norm bound to 0.05, besides the Ab-SCA-PR only the LSND ran successfully multiple times (twice). Both successful runs resulted in a γ of 3.5668. Thus, its achieved minimal γ value is slightly higher than the Ab-SCA-PR's accomplished $\mu_\gamma = 3.5213$. Besides that, just two other algorithms (GWO and WOA) finished successfully once. Thus, altogether eleven algorithms are not producing results. Evaluating $b = 0.07$, besides the proposed algorithm, the SCA, WOA, and GWO had multiple successful runs, with the first two finishing successfully three and the last two times. These algorithms achieved a standard deviation of zero. The Ab-SCA-PR still provides a very low $\sigma_\gamma = 0.0040$. However, only the GWO calculated a γ better than the Ab-SCA-PR's mean value. For the maximum norm bound of $b = 0.085$, only the proposed Ab-SCA-PR identified a valid solution $M(\lambda) \in \mathcal{M}$. Furthermore, considering the test cases in Table B.2, only the novel Ab-SCA-PR algorithm is applicable for all b values. For all other algorithms, the user would be required to change the solvers depending on the analyzed b. Compounding, there exists no a-priori-information on which algorithm will work for the problem. These limitation renders the existing algorithms infeasible for industrial application. In general, the proposed algorithm is superior analyzing high values of b, which is especially important for worst-case performance evaluations. Concluding, it can be stated that the proposed Ab-SCA-PR attained the best overall search performance. Hence, extending the original SCA search procedure with boundary adaptation and linear population reduction schemes significantly increased its suitability for the LTV worst-case analysis.

Table 4.2: Optimization results applying adaptive bisection bounds (n.f. = not feasible, and N/A = not available, S = SHADE, LS = L-SHADE, LSND = L-SHADE-ND, SPS = SPS-L-SHADE-EIG, New = Ab-SCA-PR)

b		DE	JADE	S	LS	LSND	SPS	SCA	DA	GWO	MFO	MVO	WOA	ISCA	mSCA	New
0.01	γ_{best}	2.3383	n.f	6.6080	n.f.	2.3346	n.f.	2.3349	2.3436	2.3350	n.f.	n.f.	2.3350	2.3887	2.3370	2.3365
	μ_γ	2.3383	N/A	N/A	N/A	2.3406	N/A	2.3362	3.5430	2.3361	N/A	N/A	2.3361	2.3764	2.3377	2.3372
	σ_γ	0	N/A	N/A	N/A	0.0020	N/A	0.0007	0.6704	0.0016	N/A	N/A	0.0010	0.0246	0.0010	0.0004
	n_{feas}	2	0	1	0	4	0	5	5	5	0	0	5	4	5	5
0.03	γ_{best}	n.f.	n.f	n.f.	n.f.	n.f.	n.f.	2.8172	n/A	2.8181	n.f.	n.f.	2.8177	n.f.	n.f.	2.8140
	μ_γ	N/A	N/A	N/A	N/A	N/A	N/A	4.9667	N/A	3.2052	N/A	N/A	N/A	N/A	N/A	2.8156
	σ_γ	N/A	N/A	N/A	N/A	N/A	N/A	3.7230	N/A	0.2581	N/A	N/A	N/A	N/A	N/A	0.0013
	n_{feas}	0	0	0	0	0	0	3	0	4	0	0	1	0	0	5
0.05	γ_{best}	n.f.	n.f	n.f.	n.f.	3.5668	n.f.	n.f.	n.f.	8.2731	n.f.	n.f.	3.5212	n.f.	n.f.	3.5204
	μ_γ	N/A	N/A	N/A	N/A	3.5668	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	3.5213
	σ_γ	N/A	N/A	N/A	N/A	0	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	0.0005
	n_{feas}	0	0	0	0	2	0	0	0	1	0	0	1	0	0	5
0.07	γ_{best}	4.6686	n.f	n.f.	n.f.	n.f.	n.f.	5.4528	n.f	4.6552	n.f.	n.f.	4.6818	n.f.	n.f.	4.6737
	μ_γ	N/A	N/A	N/A	N/A	N/A	N/A	5.4528	N/A	4.6552	N/A	N/A	4.6818	N/A	N/A	4.6796
	σ_γ	N/A	N/A	N/A	N/A	N/A	N/A	0	N/A	0	N/A	N/A	0	N/A	N/A	0.0040
	n_{feas}	1	0	0	0	0	0	3	0	2	0	0	3	0	0	5
0.085	γ_{best}	n.f.	n.f	n.f.	n.f.	n.f.	n.f.	n.f.	n.f.	n.f.	n.f.	n.f.	n.f.	n.f.	n.f.	16.6789
	μ_γ	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	16.8310
	σ_γ	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	0.2606
	n_{feas}	0	0	0	0	0	0	0	0	0	0	0	0	0	0	5

4.7.5 Effects of the Bisection Adaptations

The bisection modifications, namely an adaptive upper bound and skipped evaluations, reduce the search space exploration. Hence, their impact on the optimization performance must be investigated. Accordingly, the previous analyses are repeated using fixed upper and lower bounds in the bisection as provided in Tab. 4.1. Furthermore, all offspring fitnesses are fully calculated in the bisection. Thus, the optimizer is given substantially more search information but requires significantly more computational effort.

In Fig. 4.13, the three most consistent optimizers', namely the Ab-SCA, the SCA, and GWO, mean value and variance of γ over b are compared. The increase in search information shows only a minor improvement in the Ab-SCA-PR's search performance than the initial evaluation. Only for higher b values, the σ_γ reduces slightly but without visible effect on the achieved μ_γ. However, this marginally improved search performance was significantly more computationally expensive.On average, compared to the initial evaluations, twice as much time was required for the same number of function evaluations. The original SCA identifies more successful solutions over a broader range of b, with mean worst-case values closer to the Ab-SCA-PR. Furthermore, γ's variance reduces. An adverse effect on the search performance of the GWO concerning μ_γ and σ_γ is apparent. Compared to the initial analysis, it only executes successfully multiply for four b values and, hence, once less. Concluding, the existing optimizers are highly sensitivity regarding alterations in the search information, whereas the novel Ab-SCA-PR is significantly robuster.

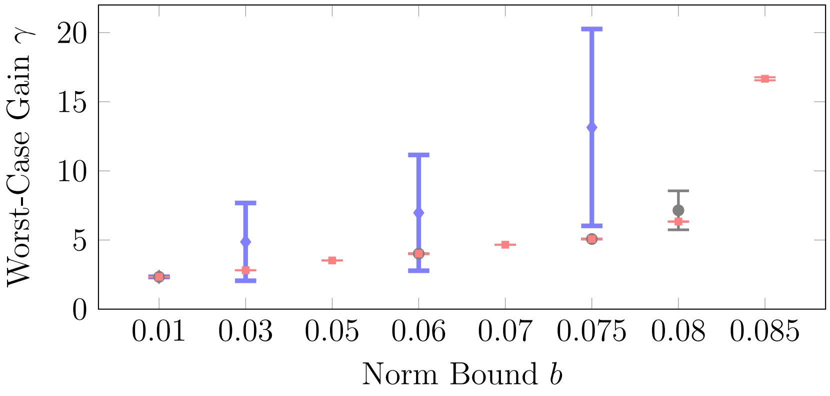


Figure 4.13: *Mean values and variance of the top three optimizers without bisection adaptations: Ab-SCA-PR (▪), GWO (♦), SCA (●)*

Comparing the detailed performance criteria for all solvers in Tab. 4.3 with the initial results in Tab. 4.2 emphasizes this observation. Note, the results for the remaining norm bounds can be found in B.1 in Appendix B. The Ab-SCA-PR's γ_{best} for a given b on average only reduces by -0.25%. Most of the existing solver show more improvement in the absolute search performance than the Ab-SCA-PR, especially for higher values of b. The most significant improvements are visible for the GWO with -58% for $b = 0.05$ and the MFO with -53% for $b = 0.08$. Nevertheless, some solvers performed significantly worse, e.g. the DE's γ_{best} increased by 262% evaluating $b = 0.06$. Consequently, extending search information and increasing computational effort does not guarantee γ_{best} improvements for the existing solvers.

Evaluating the search convergence (μ_γ), exclusively the proposed algorithm showed an improvement for all norm bounds. This improvement is insignificant, averaging -0.2%, with a maximum of 1% for $b = 0.085$. As for the absolute search performance, the existing algorithms show indifferent behavior. The DA improved the most, with -34.03% for $b = 0.01$, whereas the GWO worsened the most, with 159% for $b = 0.075$.

Concerning the search consistency, the best algorithm is still the Ab-SCA-PR reaching a 100% success rate for all evaluated norm bounds. Although the provided search information was significantly increased, none of the off-the-shelf solvers' search performance became more consistent for all b. The total number of calculated valid γ dropped from 116 to 114. However, the total number of n_{feas} increased for $b > 0.07$. On average, the analyses took nearly twice as long as in the initial, for the same number of function evaluations.

Concluding, only the Ab-SCA-PR allows to fully exploit the bisection modification with nearly no degradation in search performance. The existing solvers perform significantly better in some cases without the recommended modifications from Section 4.6.2. Nonetheless, they remain significantly less consistent than the proposed algorithm with the modifications, which in this case is also much faster.

Table 4.3: Optimization results for fixed bisection bounds (n.f. = not feasible, and N/A = not available, S = SHADE, LS = L-SHADE, LSND = L-SHADE-ND, SPS = SPS-L-SHADE-EIG, New = Ab-SCA-PR)

b		DE	JADE	S	LS	LSND	SPS	SCA	DA	GWO	MFO	MVO	WOA	ISCA	mSCA	New
0.01	γ_{best}	2.3351	n.f	6.6080	n.f.	2.3307	n.f.	2.3348	2.3361	2.3345	2.3375	n.f.	2.3343	2.3376	2.3374	2.3360
	μ_γ	2.3357	N/A	N/A	N/A	2.3463	N/A	2.3365	2.3373	2.3355	2.3376	N/A	2.3350	2.3958	2.4481	2.3365
	σ_γ	0.0005	N/A	N/A	N/A	0.0037	N/A	0.0014	0.0011	0.0008	0.0001	N/A	0.0010	0.1297	0.1565	0.0004
	n_{feas}	4	0	1	0	4	0	4	3	4	2	0	2	5	2	5
0.03	γ_{best}	2.8152	n.f	n.f.	n.f.	n.f.	n.f.	n.f.	2.8156	2.8129	n.f.	n.f.	n.f.	2.8161	3.4976	2.8130
	μ_γ	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	4.8670	N/A	N/A	N/A	N/A	N/A	2.8146
	σ_γ	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	2.8127	N/A	N/A	N/A	N/A	N/A	0.0011
	n_{feas}	1	0	0	0	0	0	0	1	5	0	0	0	1	1	5
0.05	γ_{best}	n.f.	n.f	n.f.	n.f.	n.f.	n.f.	3.5216	2.5236	3.5199	n.f.	n.f.	n.f.	n.f.	n.f.	3.5199
	μ_γ	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	3.5213
	σ_γ	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	0.0013
	n_{feas}	0	0	0	0	0	0	1	1	1	0	0	0	0	0	5
0.07	γ_{best}	n.f.	n.f	n.f.	n.f.	n.f.	n.f.	4.6743	n.f	12.599	n.f.	n.f.	4.6484	n.f.	n.f.	4.6570
	μ_γ	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	4.6593
	σ_γ	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	0.0036
	n_{feas}	0	0	0	0	0	0	1	0	1	0	0	1	0	0	5
0.085	γ_{best}	16.210	n.f	n.f.	n.f.	n.f.	n.f.	n.f.	n.f.	n.f.	n.f.	n.f.	n.f.	n.f.	n.f.	16.4784
	μ_γ	17.526	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	16.6686
	σ_γ	0.9668	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	0.1111
	n_{feas}	2	0	0	0	0	0	0	0	0	0	0	0	0	0	5

4.7.6 Comparison to Original Benchmark

The proposed Ab-SCA-PR's evaluation is concluded by comparing the best results obtained in the thesis with the best results from the algorithm used in [238], shown in Tab. 4.4. When comparing the results with the previous work [238], the proposed MH (Ab-SCA-PR) returns better results than the optimizer in [238] for both the adapted and non-adapted bisection procedure. The improvement in the Ab-SCA-PR's search performance compared to the original becomes more significant for increasing b. Given $b = 0.085$, the nonlinear program applied in [238] fails to identify a feasible solution. This nonlinear program was based on a local gradient-free search, whose performance was significantly influenced by the guessed initial solution. Thus, significant background research estimating good initial values for λ_1 and λ_2 was necessary. Searching for initial guesses is highly undesirable for the algorithm's industrial utilization, as it shall be deployable robustly with almost no a priori information. Furthermore, the applied solver in [238] can neither exploit adaptive bisection bounds nor avoid bisections at all by the nature of its search strategy.

Table 4.4: Comparison of the best results obtained with Ab-SCA-PR in this study with the best results from the algorithm used in the previous work

b	Results [238]	Ab-SCA-PR (fixed bounds)	Ab-SCA-PR (adaptive bounds)
0.01	2.3546	2.3360	2.3365
0.03	2.8161	2.8130	2.8140
0.05	3.5242	3.5199	3.5204
0.06	4.0286	4.0071	4.0127
0.07	4.6797	4.6570	4.6737
0.075	5.2607	5.0733	5.0718
0.08	6.6973	6.2975	6.3140
0.085	n.f.	16.6624	16.6789

4.8 Summary

A robust LTV analysis framework allowing for the worst-case gain calculation of finite-horizon LTV systems in interconnection with perturbations was introduced. By representing the perturbations via time-domain IQCs, recent advances in the robustness analysis of LPV systems were extended to the LTV case. Built upon the solvability of an RDE, an analysis condition upper bounding the interconnection's worst-case gain was proposed. The selection of a fixed IQC factorization Ψ and a free parameterization $M \in \mathcal{M}$ parameterizes this RDE with M and γ. By performing a bisection of γ over a fixed $M \in \mathcal{M}$, it was shown that the chosen M directly influences the achievable minimal gamma. As this relation appears locally convex, a nonlinear optimization problem directly optimizing gamma over M constrained by the RDE's solvability was proposed. This optimization problem can be readily solved by bisecting γ for a fixed M in an inner loop. A global optimization identifies M in an outer loop to find the minimal γ. The direct optimization approach avoids the explicit definition and respective gridding of the storage function necessary for LMI-based analysis conditions.

Following this, general guidelines to handle the optimization problem's nonlinear constraints were stated. These guidelines define a general requirement list to assess the suitability of a given nonlinear program for the worst-case gain optimization problem. Furthermore, general recommendations for efficient computation of the worst-case gain were proposed exploiting the optimization problem's structure. Based on these guidelines, two specifically tailored MHs, Ab-SCA-PR, and Log-L-SHADE were developed.

The novel Ab-SCA-PR algorithm was successfully applied to an industry-relevant benchmark example, where it outperformed 13 off-the-shelf metaheuristics. Moreover, it solved the problem more robustly than an algorithm initially proposed and applied by the author.

The proposed Log-L-SHADE algorithm was specifically designed for large parameterizations built from multiple IQC. Hence, it will be deployed on two elaborate worst-case analyses of industrial complexity to show that the robust LTV framework can provide a valuable asset in the certification process.

Hence, this chapter provides the necessary theoretical and computational tools for an efficient worst-case analysis of systems with time-varying dynamics.

Chapter 5

Finite Time Horizon Analysis of an Autolanded Aircraft in Landing Configuration under Crosswind

5.1 Introduction

The final approach presents the most dangerous flight segment in aircraft operations, accounting for more than 49% of all disastrous accidents, see [8]. Autoland systems (AS) were introduced to moderate the risk, primarily for poor visual conditions, at the beginning of the 1950s [9]. These generally employ a runway-based instrument landing system (ILS) to produce a localizer and glideslope signal tracked by the aircraft's autopilot. Given its operational limits, the autoland system must satisfy tight touchdown constraints for safe operation.

In general, Monte Carlo analyses [4] or worst-case optimizations on the nonlinear model [5] are state-of-the-art methods to evaluate touchdown conditions. Given the aircraft's large parameter space, various possible environmental conditions, and disturbances, these procedures are computationally costly. Hence, Monte Carlo simulations are not suitable to provide fast feedback in an iterative design/tuning process of the autopilot. Furthermore, the tools provided cannot provide guaranteed worst-cases but only a probability distribution or a lower bound of the worst case, respectively.

However, linear analysis methods can provide guaranteed worst-cases. As the aircraft tracks a particular trajectory imposed by the ILS signals, its nonlinear dynamics can be treated as solely time-dependent over a finite horizon. Hence, the linearization along this particular trajectory provides a corresponding finite horizon LTV model. Treating the aircraft as a finite horizon LTV system allows applying the LTV analysis framework in Chapter 4 to calculate worst-case touchdown conditions.

This chapter introduces a robust LTV worst-case analysis for touchdown conditions of an autolanded aircraft under crosswind. The nonlinear aircraft dynamics are directly obtained from [17], which provides a large airliner model in final approach configuration and the corresponding nonlinear simulation environment. It is freely available from http://w3.onera.fr/smac/?q=aircraftModel. The LTV representation of the aircraft dynamics is derived by numerical linearization along a reference approach trajectory. An autoland controller for this aircraft model was developed in [18] whose worst-case touchdown performance under wind disturbance will be evaluated.

A tailored wind filter is designed to cover this turbulent wind disturbance. It is specifically designed to generate realistic wind disturbance under the constraints imposed by the strict BRL. Thus, an arbitrary norm-bounded input disturbance must be shaped into a wind signal, whose PSD matching Dryden-like turbulence common for aircraft certification. The influence of constant/frozen altitude-dependent wind fields, e.g. wind shears, is directly included in the aircraft's linearized dynamics.

The LTV worst-case analysis results are evaluated against a Monte Carlo analysis conducted on the corresponding industry-sized, high-fidelity nonlinear airliner model. The evaluation concludes the chapter and demonstrates the LTV analysis's adequacy for the final approach problem.

This chapter contributes an entirely new method to analyze worst-case touchdown conditions of autolanded aircraft. In contrast to existing linear worst-case methods, the LTV framework explicitly respects the aircraft's time-varying dynamics and the approach problem's finite time horizon. Additionally, the chapter contributes a novel wind filter design procedure to cover general wind signals in the LTV framework accurately.

5.2 Nonlinear Dynamics

The nonlinear aircraft model describes a large twin-engine civil transport aircraft in final approach configuration from 1000ft above the runway until touchdown and is directly taken from [4]. It is implemented as a standard nonlinear six-degrees-of-freedom flight mechanics model concerning translational velocities u, v, and w and the angular rates p, q, and r formulated in the body-fixed frame. The aircraft's orientation in the earth-fixed reference frame is defined by the well-known Euler angles Φ, Θ, and Ψ [248]. In terms of x, y, and z, the aircraft's center of gravity's position in the earth fixed frame is specified. The flight path is defined relative to the earth's surface by the path angle γ, course angle χ, and ground speed V_g, i.e. the horizontal speed relative to earth. The aerodynamic angle of attack α and sideslip angle β are defined based on the aerodynamic velocity V_a, which results from superimposing the aircraft's translational velocity and atmospheric wind speed. Fully linear aerodynamic coefficients are implemented. However, the aerodynamics respect the ground effect. The aircraft is controlled by anti-symmetrically operating ailerons, an elevator, a rudder, and symmetrically operating twin engines during the approach. Rate and amplitude-limited first-order filters are utilized to model the control surface actuators' dynamics. Their specifications are provided in Tab. 5.1. The engine's thrust is modeled via the exhaust pressure ratio (EPR), with a minimum and maximum of 0.95 and 1.6. A first-order delay with a

Table 5.1: Actuator and engine parameter

	Time Constant [s]	Deflection Limited [°]	Rate Limit[°/s]
Aileron	0.06	±55	60
Elevator	0.07	±25	20
Rudder	0.2	±30	30

time constant of 2s is implemented to cover its dynamics. The maximum rate of change of the EPR is limited to 0.11/s. Note that the aircraft's LTV model will omit all rate and deflection limits. The model utilizes a simple atmosphere model based on the international standard atmosphere model. Hence, the influence of different airfield elevations as well as outside air temperatures on the autopilots performance can be analyzed.

5.3 Autolanding Controller

In this chapter, the autoland controller as proposed in [18] is analyzed. As the design considers the lateral and longitudinal dynamics as decoupled, the autopilot consists of two separate controllers. The longitudinal controller handles the pitch motion and velocity, whereas the lateral controller the roll and yaw motion. Their basic structure and functionality are described in the following mainly to introduce necessary adaptations for the LTV analysis. A schematic of the overall control architecture is shown in Fig. 5.1.

5.3.1 Longitudinal

The longitudinal controller utilizes a cascaded control structure, consisting of an inner loop and two nested outer loops. Given the engines' and elevator' significantly different bandwidths, throttle δ_T is used to control airspeed, and the elevator δ_e is used to control the attitude of aircraft during the approach. The longitudinal autoland controller cannot be directly implemented as in [18]

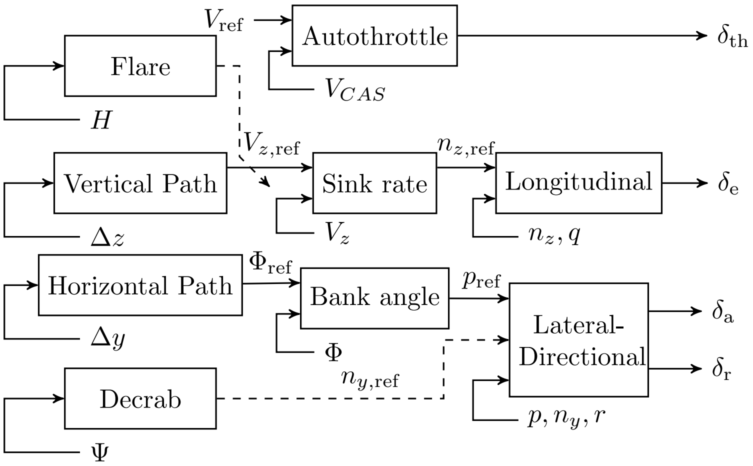


Figure 5.1: *Architecture of the autoland controller proposed in [18]*

for the LTV analysis of the touchdown constraints. Reasons are the changing control laws, e.g. activation of flare and nonlinearities in the signal modifications. The necessary alterations are highlighted in the controller's integrator chain analogy displayed Fig. 5.2 and explained in detail in the following paragraphs. Note the LTV analysis omits all rate and amplitude saturations and,

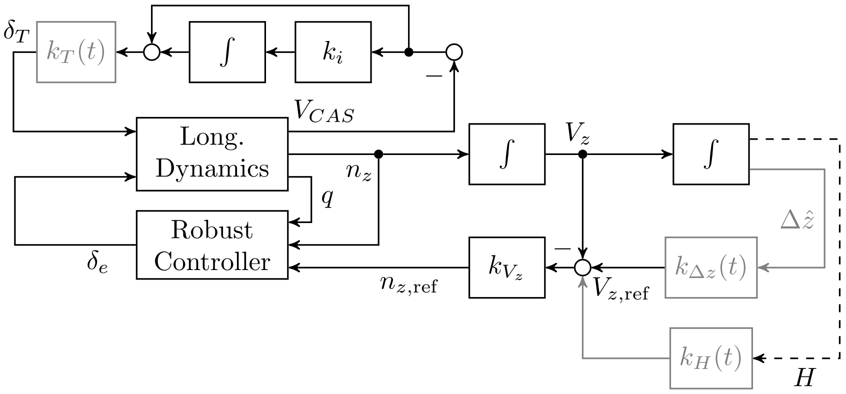


Figure 5.2: *Longitudinal part of the autoland controller as used in the LTV analysis (adaptations in gray)*

thus, they are not explicitly highlighted in Fig. 5.2.

Signal Modification

The deviation to the glide slope provided to the autopilot is calculated concerning the aircraft's sensor location and not the landing gear. Given the sensors offsets $\delta z_{\text{GLD}} = -5\text{m}$ vertically and $\delta x_{\text{GLD}} = 28\text{m}$ horizontally in the aircraft fixed frame, high pitch angles likely during the approach would provide Δz values differing significantly from the landing gear's Δz_{LG}. Therefore, it is more reasonable to track the actual landing gear deviation rather than the sensor deviation Δz. The necessary geometrical correction and filtering of the Δz signal are achieved by

$$\Delta\hat{z} = \frac{\omega_z}{s+\omega_z}\left(\Delta z - \delta x_{\mathrm{GLD}}\sin\theta + \delta z_{\mathrm{GLD}} + \frac{V_z}{\omega_z}\right), \tag{5.1}$$

with $\omega_z = 2\mathrm{rad/s}$. As this filter is nonlinear in θ, it cannot be used directly in the LTV analysis. The latter requires linearization of (5.1) along the approach path resulting in the filter's following LTV representation

$$\Delta\Delta\hat{z} = \frac{\omega_z}{s+\omega_z}\left(\Delta\Delta z(t) - \delta x_{\mathrm{GLD}}\cos\left(\theta_0(t)\right) + \frac{\Delta V_z(t)}{\omega_z}\right). \tag{5.2}$$

In (5.2), Δ denotes the deviation from the reference trajectory and $\theta_0(t)$ the respective reference pitch angle.

Furthermore, the sink rate signal provided to the controller needs to be corrected as the barometrically measured sink rate V_z does not match the sink rate of the landing gear in the case of sloped runways. Accordingly, the altitude of the landing gear above ground level H_{AGL} measured with the radio altimeter during the final approach is passed through the differentiating filter

$$\hat{V}_z = \frac{15s}{s+15}H_{\mathrm{AGL}}, \tag{5.3}$$

leading to a better sink rate estimate $\hat{V}_z$ with respect to the ground. The filtered signal is only utilized during the flare and can be directly applied in the LTV analysis.

Auto-Throttle

An auto-throttle controller maintains the approach speed V_{ref} constant under wind disturbance and attitude changes. It is implemented as a standard PI controller calculating the necessary throttle command δ_T based on the measured calibrated airspeed V_{CAS}:

$$\delta_T = k_T \left(1 + \frac{1}{15}\frac{1}{s}\right)(V_{\mathrm{ref}} - V_{\mathrm{CAS}}), \tag{5.4}$$

with $k_T = 0.045$. The auto-throttle gets deactivated, reaching a radar altitude of $H_{\mathrm{AGL}} = 20\mathrm{m}$ to initiate the flare maneuver. Below 20m, the last throttle command is fixed and fed to the engine control in the nonlinear simulation until touchdown. In contrast, in the LTV case, the gain k_T is implemented as a time-varying scalar, as highlighted in Fig. 5.2. Here, it is exploited that in the nominal case, $H_{\mathrm{AGL}} = 20\mathrm{m}$ is passed at a particular time T_{f} along the trajectory. To approximate the nonlinear model's behavior in the LTV analysis, the value of k_T equals 0.045 for times before T_{f} and zero afterwards. Thus, the auto-throttle command effectively goes back to the trim value. A corresponding implementation in the nonlinear simulation showed a close match with the original in numerous evaluations.

Inner Loop Controller

A multi-input single-output H_∞ controller of fifth-order is used in the inner-loop. Based on a calculated $n_{z,\mathrm{ref}}$ and the respectively measured signals n_z and q, it calculates the elevator deflections. The design philosophy follows a classical PI regulator with additional pitch damping implementation. However, it provides additional lead compensation and roll-off in both control channels. A discretized version using a standard Tustin transformation with a sampling rate of 20Hz is used in the nonlinear analysis, whereas it is directly applied in the LTV analysis.

Sink Rate Control

The $n_{z,\mathrm{ref}}$ signal tracked by the inner loop controller is provided by the sink rate tracking controller. It is a proportional controller represented by

$$n_{z,\mathrm{ref}} = k_{V_z}(V_{z,\mathrm{ref}} - V_z), \tag{5.5}$$

with $k_{V_z} = 0.625$. In the nonlinear analysis, the output $n_{z,\mathrm{ref}}$ fed to the inner loop is limited to $\pm 5\mathrm{m/s}^2$. Saturations are inherently challenging to cover via IQCs, see e.g. [7]. However, as the saturation limit was not reached in any test scenario, (5.5) can be reasonably applied without in the LTV analysis.

Glide Slope Tracker

The glide slope tracker provides the reference sink rate $V_{z,\mathrm{ref}}$ in (5.5), based on the glide path deviation until initiation of the flare maneuver at $H_{\mathrm{AGL}} = 20\mathrm{m}$. It is implemented as a simple proportional controller

$$V_{z,\mathrm{ref}} = k_{\Delta z}\Delta\hat{z}, \tag{5.6}$$

with $k_{V_z} = 0.1$. In the nonlinear analysis, the output is limited to differ a maximum $\pm 3\mathrm{m/s}$ from the trim value. Again, this output saturation is omitted in the respective LTV analysis. Equivalent to the auto-throttle's implementation, the gain $k_{\Delta z}$ is a time-varying scalar, with a value of 0.1 before T_{f} and 0 after.

Flare Controller

From a radar altitude of $H_{\mathrm{AGL},0} = 20$ on, the flare controller is engaged and provides the $V_{z,\mathrm{ref}}$ signal to the subsequent loop instead of the glide slope tracker. Therefore, the feedback loop from H_{AGL} to V_z is closed via

$$V_{z,\mathrm{ref}} = k_H(H_{\mathrm{AGL}} + H_{\mathrm{bias}}). \tag{5.7}$$

The gains $k_H = 1/\tau$ and H_{bias} are calculated from the sink rate at flare initialization $V_{z,\mathrm{f}}$, via $H_{\mathrm{bias}} = \tau V_{z,\mathrm{f}} - H_0$, with $\tau = H_0/(V_{z,\mathrm{f}} - V_{z,\mathrm{TD}})$. Here, the desired vertical velocity at touchdown $V_{z,\mathrm{TD}}$ is 0.3m/s. Accordingly, the constants get calculated at flare initialization T_{f} using an estimate of the current reference sink rate $V_{z,\mathrm{f}} = \frac{5}{s+5}V_{z,\mathrm{ref}}$. The result is a varying τ control law, which accounts for different approach scenarios, mainly wind disturbance and approach velocities $V_{z,\mathrm{f}}$. For the nominal trajectory under headwind τ's value is 8.39 and under tailwind 6.26, while varying in a range from 4.6 to 12.3 in extensive evaluations over the parameter space. In the LTV model, k_H is implemented as a time-varying scalar with a value of 0 before T_f and $1/\tau$ after. Also, only ΔH is fed back. The variation of τ over many approach parameters is covered in the uncertainty set applied in the LTV analysis. As time-varying gains are applied in (5.7) and (5.6), the flare controller's and glide path tracker's added output is used as input for the sink rate tracker. Thus, the change of the control law is accurately covered in the LTV analysis.

5.3.2 Lateral Control System

The lateral controller is designed under the "crabbed approach" paradigm. At the same time, ϕ shall remain zero for zero deviation from the localizer. As for the longitudinal control, a cascaded control architecture, with an inner loop and two nested outer loops, is utilized. Again, adaptations to the original implementation are necessary to apply the lateral part of the autoland controller. They are highlighted in the controller's integrator chain analogy displayed in Fig. 5.2. A detailed explanation is given in the following paragraphs. In the lateral LTV analysis, all rate and amplitude saturations are also omitted and not explicitly highlighted in Fig. 5.2.

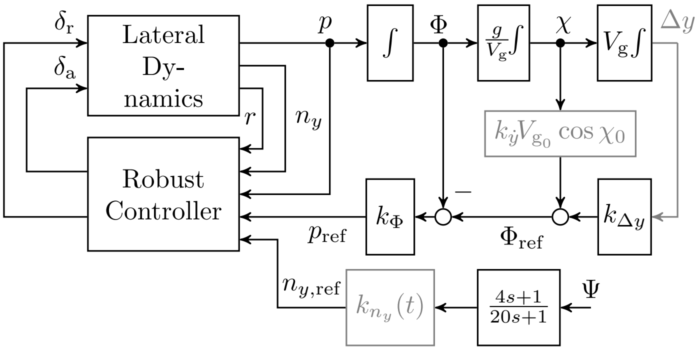


Figure 5.3: *Lateral part of the autoland controller as used in the LTV analysis (adaptations in gray)*

Signal Modification

As for the glideslope, a localizer offset signal is necessary, as it is desirable to control the deviation of the landing gear rather than the sensor. Due to the large heading angles required for a crabbed approach and the horizontal distance between the main landing gear and the ILS sensor of $\delta_{x_{\mathrm{LOC}}} = 30\mathrm{m}$, the difference between both values can grow considerably large. The necessary filtering and geometrical correction are given by

$$\Delta\hat{y} = \frac{\omega_y}{s+\omega_y}\left(\Delta y - \delta x_{\mathrm{LOC}} \sin\psi + \frac{V_g \sin\chi}{\omega_y}\right), \tag{5.8}$$

with $\omega_y = 0.3\mathrm{rad/s}$. This filter is nonlinear in ψ and χ. Hence, it must be linearized along the reference approach trajectory

$$\Delta\Delta\hat{y} = \frac{\omega_y}{s+\omega_y}\left(\Delta\Delta y - \delta x_{\mathrm{LOC}} \cos(\psi_0)\Delta\psi + \frac{V_{\mathrm{g}_0}}{\omega_y}\cos(\chi_0)\Delta\chi\right). \tag{5.9}$$

The values $\Delta\Delta\hat{y}$, $\Delta\chi$, and $\Delta\psi$ represent the deviation from the reference approach trajectory, whereas V_{g_0}, ψ_0, and χ_0 are the values on the reference approach trajectory at a specific time.

Inner Loop Controller

The inner loop applies a ninth-order multivariable H_∞ controller. It provides lateral directional control via roll rate and lateral load factor feedback, damping augmentation of the dutch-roll mode via yaw rate feedback, and adverse yaw cancellation via a pronounced roll rate to rudder forward-feed. Furthermore, it adds lead compensation and roll-off characteristics. The robust controller provides the command signals δ_{a} and δ_{r} to the respective actuators. This controller can be directly applied in the LTV analysis, whereas a discretization using standard Tustin transformation with a sampling frequency of 20Hz is used in the nonlinear analysis.

Bank Angle Tracking

The reference roll rate for the inner loop controller is provided by the bank angle tracking realized by proportional bank angle feedback. It is implemented as

$$p_{\text{ref}} = k_\phi \left(\phi_{\text{ref}} - \phi\right), \tag{5.10}$$

with $k_\phi = 0.7$ and directly applied in the LTV analysis. The bank angle ϕ is controlled to achieve zero deviation Δy from the localizer signal

Localizer Tracking

The reference bank angle in (5.10) is provided by the localizer tracker implemented as a derivative (PD) controller is implemented. Rather than directly implementing an differentiator, $\Delta\dot{y}$ is approximated by $\Delta\dot{y} \approx V_{\text{g}} \sin\chi$. Thus, the controller is implemented as

$$\phi_{\text{ref}} = k_{\Delta y}\Delta y + k_{\dot{y}} V_{\text{g}} \sin\chi, \tag{5.11}$$

with $k_{\Delta y} = 0.003$ and $k_{\dot{y}} = 0.033$. This implementation assures that ϕ remains zero for zero deviation. In the nonlinear, model the maximum commanded bank angle is limited to ±30°. This saturation is again omitted in the LTV analysis. Nominal LTV worst-case analyses and subsequent simulations showed no control signals close to the saturation limit were commanded. As (5.11) is nonlinear in χ, the controller needs to be linearized along the reference approach trajectory. The linearized controller used in the LTV analysis is given by

$$\Delta\phi_{ref} = k_{\Delta y}\Delta\Delta y + k_{\dot{y}} V_{\text{g}_0}(t)\cos\left(\chi_0(t)\right)\Delta\chi, \tag{5.12}$$

where $\Delta\phi_{\text{ref}}$ and $\Delta\chi$ are the offsets of the reference bank angle and nominal course angle, respectively, with respect to the nominal/reference approach trajectory/states. V_{g_0} and χ_0 are the nominal ground speed and course angle, respectively, at a given time.

Decrab Controller

The decrab maneuver is initiated at a fixed H_{AGL} of 5m. Following a specific approach trajectory, this altitude maps to a particular time t_{DC}. The nonlinear model uses a ψ feedback controller with lag compensator given by

$$n_{y,\mathrm{ref}} = k_{n_y} \frac{4s+1}{20s+1} \psi, \tag{5.13}$$

with $k_{n_y} = 33$. During decrab, the bank angle command in the nonlinear model is limited to ±5°, which is omitted in the LTV analysis. In the LTV analysis, k_{n_y} is implemented as a time-varying scalar with a value of zero until t_{DC} and 33 after. Thus, the initiation of the decrab maneuver is covered in the LTV analysis as only after t_{DC} a value different to zero is fed to the robust controller.

5.3.3 Linear Dynamics

The LTV representation of the aircraft is derived by linearizing the nonlinear model along a nominal approach trajectory. This trajectory is calculated based on a reference approach in the nonlinear simulation using the autoland controller from Section 5.3. For the nominal aircraft configuration, a center of gravity position of 22% and a mass of 140t is chosen. The aircraft lands on a runway at mean sea level under standard/nominal ISA conditions. The simulation starts at an altitude of 300m, 30m below the glideslope signal and 20m right of the localizer signal and ends with the aircraft's touchdown. The nonlinear aircraft dynamics are extended with the actuator dynamics and linearized along the reference trajectory using the tools provided by [17]. This results in a general finite horizon LTV presentation G_t of the aircraft's dynamics described by (3.7). For the approach scenario, the cross-coupling between the longitudinal and lateral motion is neglectable. Hence, a separate model for the longitudinal dynamics $G_{t,\mathrm{long}}$ and lateral dynamics $G_{t,\mathrm{lat}}$ can be extracted from G_t. The resulting LTV models' respective states, inputs, and outputs are summarized in Table C.2 and Table C.1 in Appendix C. In interconnection with the longitudinal autoland controller in accordance to Fig. 5.2, the analyzed longitudinal closed loop has 17 states, two disturbance inputs, namely longitudinal wind u_w and vertical wind w_w, and the two performance outputs, the vertical touchdown velocity $V_{z,\mathrm{TD}}$ and the height of the landing gear 60m behind the touchdown threshold H_{60}. The resulting lateral closed-loop, as pictured in Fig. 5.3 has a total of 19 states, and the three performance outputs, namely bank angle ϕ, lateral offset to the centerline y_{LG}, and the landing gear's sideslip angle relative to the centerline β_{LG}. The single disturbance input is the lateral turbulence v_w. Note that the influence of static wind profiles can be implicitly respected in the LTV dynamics by including them then calculating the reference trajectory, i.e. executing the reference approach with a static wind field.

During the approach, it can be noticed that the dynamics of the aircraft vary noticeably over time. This is depicted exemplarily in Fig. 5.4, showing the Bode magnitude plot of the transfer function from δ_e to α evaluated at frozen points in time along the approach trajectory.

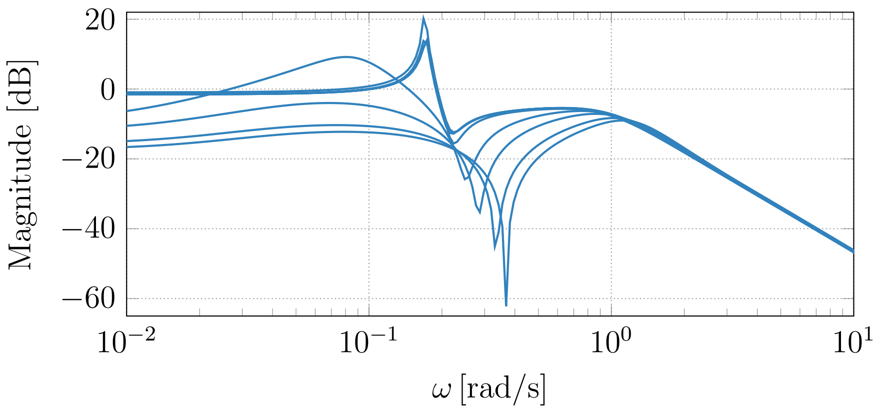


Figure 5.4: *Bode magnitude plot of δ_e to α transfer function evaluated at different points in time along approach trajectory (——)*

5.3.4 Wind Model

The wind disturbances analyzed in this paper are derived from the original landing challenge [17], which was also used for the design verification of the autoland controller in [18]. It is built by the superimposition of an altitude-dependent wind shear and a turbulent wind field. Based on the resulting wind profiles, an equivalent/corresponding LTV wind model is derived. It allows for a direct quantitative comparison between the LTV analyses and the Monte Carlo analyses. The longitudinal and lateral touchdown conditions are each analyzed for two distinct wind scenarios. In the longitudinal case, first, a tailwind with a fixed maximum amplitude of 10kts is analyzed. The second analysis covers headwind with a fixed maximum amplitude of 30kts. Simultaneously, lateral wind shear and simultaneous turbulent head-/tailwind and vertical turbulence are applied in both scenarios. In the lateral scenarios, first, a tailwind with a fixed maximum amplitude of 10kts is analyzed. The second analysis covers headwind with a fixed maximum amplitude of 30kts. In both scenarios, a lateral wind shear superimposed with turbulent crosswind is applied.

Nonlinear Analysis

The lateral wind shear has a fixed maximum amplitude of 25kts, i.e. in each Monte Carlo run, its amplitude reaches 25kts from the left. Its amplitude builds up with decreasing altitude, reaching its maximum 15m above ground as described by

$$v_{\mathrm{lat}} = 25\mathrm{kts} \cdot \min\left(\frac{(H_{\mathrm{AGL}} - H_{\mathrm{AGL},0})^2}{(H_{\mathrm{AGL},0} - 15\mathrm{m})^2}, 1\right), \tag{5.14}$$

where $H_{\mathrm{AGL},0}$ is the altitude above ground level at initialization of the analysis. The turbulence is generated by filtering a random number signal with a mean of zero, a variance of one, and a sample time of 0.05s through the first-order filter G_{lat} described by

$$G_{\mathrm{lat}} = \sigma_{\mathrm{lat}} \frac{20}{2.5s + 1}, \tag{5.15}$$

with σ_{lat} having a fixed value of 5.8kts. Note that in [17], σ_{lat} is a normally distributed random number with mean 0kts and variance 7kts. Consequently, the wind disturbance in the thesis is, on average, more conservative.

For the longitudinal wind disturbance, two scenarios exist. The first is a constant altitude-dependent headwind of 30kts and the second a constant altitude-dependent tailwind of 10kts. In both cases, the maximum value of the wind shear v_{long} is reached in each run of the Monte Carlo simulation. The altitude scheduling is identical to (5.14), with the maximum amplitude adjusted accordingly. The superimposed turbulence field is generated by filtering band-limited white noise through a shaping filter identical to (5.15). In addition to this combined head-/tailwind, there is a turbulent vertical (up/down) wind profile, which is generated by passing a random number signal with a mean of zero, a variance of one, and a sample time of 0.05s through the shaping filter G_{vert} given by

$$G_{\text{vert}} = \sigma_{\text{vert}} \frac{2.25}{0.125s + 1}, \tag{5.16}$$

where σ_{vert} is a constant with a value of 1.5kts. Note that in [17], σ_{vert} is a normally distributed random number with mean 0kts and variance 2.5kts.

Linear Analysis

Due to their altitude dependence, the wind shear profiles are unique for a specific approach trajectory. Therefore, calculating the reference trajectory under the respective wind scenarios, the derived linear model includes the wind profiles' influence on the aircraft dynamics. Thus, only the turbulent component must be covered in the LTV analysis.

Consequently, the (external) wind disturbance in the LTV analysis only needs to cover the turbulent wind field explicitly. As the turbulence filters (5.15) and (5.16) require a white noise input, they cannot be applied directly in the LTV analysis. Recalling the definition of the $L_2[0,T]$ to $\|e(T)\|_2$ gain, the LTV wind filter has to be designed in such a way that it converts any $L_2[0,T]$ bounded signal into realistic turbulence. In particular, the LTV wind filter's design goal is to match the power spectral density (PSD) of the turbulence signals utilized in the Monte Carlo analysis. As the lateral and longitudinal turbulence are identical in the nonlinear model, they also share the same turbulence filter in the LTV analyses. Consequently, only two LTV turbulence filters have to be designed, i.e. a common filter for lateral and horizontal turbulence and a separate one for the vertical turbulence.

The proposed design procedure is closely related to the work in [216] and can be applied to any (turbulent) wind profile. It consists of three steps and is exemplarily shown for vertical turbulence. In the first step, 2000 random turbulence profiles are generated along the reference approach trajectory using the nonlinear simulation's turbulence wind model with a fixed sampling time of 20Hz. The second step is the calculation of the PSDs $\Omega_{v_{w,i}}$ of the time domain wind signals $v_{w,i}(t)$ using

$$\Omega_{v_{w,i}}(\omega) = \lim_{T\to\infty} \frac{2}{\pi}\frac{1}{T}\left|\int_0^T v_{w,i}(t)e^{-j\omega t}dt\right|^2 . \tag{5.17}$$

Accordingly, the PSD of a time-domain signal is simply the average squared of the signal's Fourier transform. The Fourier transform of the wind signals can be calculated via a fast Fourier transform (FFT), e.g. using the built-in Matlab function `fft`. In the third step, a minimum phase first-order transfer function is calculated upper bounding the square roots of the calculated PSD magnitudes, $\sqrt{\left|\Omega_{v_{w,i}}(\omega)\right|}$ of all wind signals using the internal Matlab function `fitmagfrd` and safety margin of 8dB. Note that the square root is necessary to account for the general PSD input/output relation $\Omega_{\text{out}}(\omega) = |H(\omega)|^2\,\Omega_{\text{in}}(\omega)$, where $H(\omega)$ is a corresponding shaping filter. The safety accounts for the finite amount of considered wind signals, an immanent probability of exceedance for any statistically derived wind turbulence intensity, see, e.g. [216] and the missing information on the worst-case input signal's PSD. Subsequently, the transfer function is transformed into a state-space representation of the wind filter $G_{v_w,\text{LTV}}$ as the analysis is conducted in the time-domain.

In Fig. 5.5, the PSD magnitude square roots for a selection of $v_{\text{w},i}$ are compared to the magnitude of the fitted wind filter $G_{v_w,\text{LTV}}$. This approach allows to easily cover the influence of two turbulence disturbance inputs by increasing

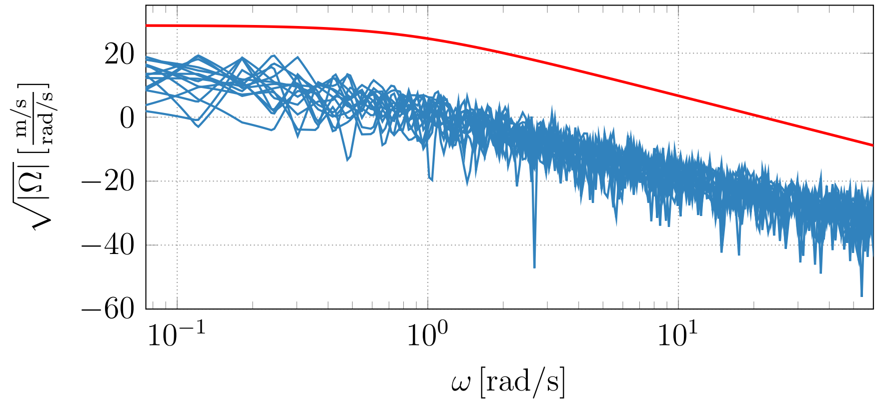


Figure 5.5: *Comparison of the power spectral density magnitudes: LTV wind filter $G_{v_w,LTV}$ (—), Monte Carlo turbulence signals $v_{w,i}$ (—)*

the lower bound of the transfer function fit. Note that for the LTV analysis, a simple LTI filter is sufficient because the underlying turbulence generator in the nonlinear model does not possess any parameter dependencies.

5.3.5 Uncertainty Model

Table 5.2: Aircraft and environmental parameters covered in Monte Carlo Analysis

Parameter	Distribution*	min	max
Mass [t]	uniform	120	180
Center of mass [%]	uniform	15	41
Temperature [°C]	uniform	-69	40
Runway slope [%]	$\mathcal{N}(0, 0.4)$	-2	2
Glide Slope [°]	$\mathcal{N}(-3, 0.075)$	-3.15	-2.85
Runway elevation [ft]	$[-1000, 250] : 50\%$	-1000	9200
	$[250, 750] : 28.33\%$		
	$[750, 1250] : 13.33\%$		
	$[1250, 1750] : 3.33\%$		
	$[1750, 2500] : 1.67\%$		
	$[2500, 3500] : 1.00\%$		
	$[3500, 4500] : 0.67\%$		
	$[4500, 9200] : 1.67\%$		

$^*\mathcal{N}(\mu, \sigma)$: normal distribution with mean μ and standard deviation σ.

The LTV analysis has to cover the aircraft configurations and environmental conditions evaluated in the Monte Carlo analysis, as summarized in Tab. 5.2. Note that the percentages for the different runway elevation intervals denote the percentage of the overall parameter set values uniformly distributed inside this interval.

Different aircraft and environmental parameters directly influence the aircraft's dynamics and the approach trajectory. This is emphasized with the Bode magnitude plots of transfer functions from δ_e to α at a fixed altitude of 10m above ground level, i.e. the time the aircraft passes 10m, in Fig. 5.6. The red graph shows the nominal configuration, whereas the blue charts resemble ten different dynamics randomly generated using Tab. 5.2. Thus, the corresponding LTV closed-loop dynamics differ from the nominal configuration in Section 5.3.3. Explicitly respecting every uncertain parameter in Tab. 5.2 results in an extensive IQC parameterization. Therefore, the following general uncertain LTV representation of the (lateral or longitudinal) closed loop is introduced, whose range of behaviors covers the dynamics of a large set of approaches:

$$G_\mathrm{CL} = G_\mathrm{CL,nom}(1 + W_\mathrm{LTV}(t)\Delta). \tag{5.18}$$

In (5.18), Δ is a norm bounded dynamic LTI uncertainty, with $\|\Delta\|_\infty \leq 1$, $G_\mathrm{CL,nom}$ represent the nominal longitudinal or lateral closed-loop dynamics, and W_LTV is a time-varying shaping filter. The weighting filter W_LTV is calculated based on the approach proposed in [249]. Firstly, LTV models of the aircraft and controller resulting from 200 approaches are generated using parameter combinations based on Tab. 5.2 and a selected static wind profile. At frozen altitudes, the LTI weight W is calculated such that all approach models are included in the uncertainty set (5.18), covering all disturbance inputs and all outputs of the respective LTV aircraft model (see Tab. C.2 and C.1). Afterwards, the altitude grid is mapped to the time grid of the corresponding nominal approach trajectory resulting in a time-dependent grid of weights. Finally, the time-varying weighting filter W_t is obtained by piecewise cubic Hermite polynomial interpolation of the obtained weights over the time grid of the nominal trajectory. Each wind scenario must be analyzed individually, hence, a separate weighting filter has to be calculated for each analyzed static wind profile introduced in Section 5.3.4 and for the lateral/longitudinal motion of the aircraft. The results are two diagonal 2 × 2 weights $W_{\text{long},1}$ and $W_{\text{long},2}$ for the longitudinal analysis due to the two disturbance inputs and two SISO weights $W_{\text{lat},3}$ and $W_{\text{lat},4}$ for the lateral analysis. Note that a set of 200 approaches was determined sufficient to cover the Monte Carlo simulation parameter set, as including more models did not further increase the calculated LTV worst-case gains.

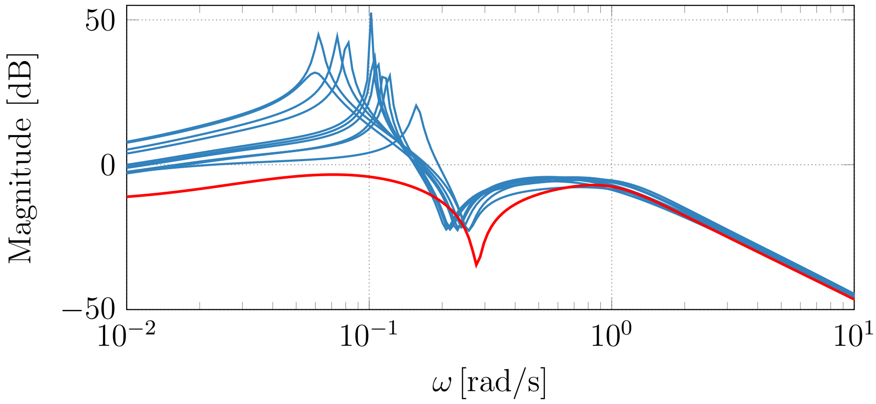


Figure 5.6: *Bode magnitude plot of δ_e to α transfer evaluated for an AGL of 10m: nominal model (—), random models in the parameter space defined by Tab. 6.1 (—)*

5.4 Analysis

In this section, the different analyzed wind disturbance scenarios and analysis interconnections are described. Afterwards, the Log-L-SHADE algorithm is applied to solve the arising LTV worst-case analysis problem. The section concludes by comparing the LTV analysis results with the nonlinear model's corresponding Monte Carlo analyses.

Table 5.3: Wind scenarios covered in the analysis

Case	Model	Static Wind [kts]		Turbulence	Criterion
		Long.*	Lat.		
1	Long.	+10	25	σ_u, σ_w	$V_{z,\mathrm{TD}}$,H_{60}
2	Long.	−30	25	σ_u, σ_w	$V_{z,\mathrm{TD}}$,H_{60}
3	Lat.	+10	25	σ_v	$y_{LG,\mathrm{TD}}$,$\beta_{\mathrm{LG,TD}}$, ϕ_{TD}
4	Lat.	−30	25	σ_v	$y_{LG,\mathrm{TD}}$,$\beta_{\mathrm{LG,TD}}$, ϕ_{TD}

* positive/negative value indicates tailwind/headwind

5.4.1 Analysis Scenarios

The analysis scenarios in this chapter are closely related to the one introduced in [17]. However, they are more tailored to give feedback in the controller's design process regarding the worst-case touchdown conditions. Hence, rather than running a single Monte Carlo analysis to evaluate the complete autoland controller design, LTV worst-case analyses of the longitudinal and lateral controller for four different wind scenarios are conducted. Afterwards, the results are compared to the results of the corresponding Monte Carlo analyses.

A summary of the different analyzed wind scenarios and resulting test cases are given in Tab. 5.3. The first two test cases solely focus on the aircraft's longitudinal dynamics evaluating the vertical velocity $V_{z,\mathrm{TD}}$ of the landing gear at touchdown and the landing gears altitude above runway 60m behind the threshold H_{60}. For these analyses, two separate LTV models, i.e. longitudinal closed-loop and uncertainty weight, must analyze either head- or tailwind. The first one is derived from a reference trajectory calculated under 25kts crosswind and 10kts tailwind. Cross- and headwind of 25kts and 30kts, respectively, are applied to generate the second test case's reference trajectory used to derive the LTV models and uncertainty weight. Both test cases apply external disturbances in the form of horizontal and vertical turbulence. Note that the final time for evaluating the H_{60} constraints is not related to the touchdown but when the aircraft is 60m behind the threshold.

The last two cases exclusively analyze the lateral dynamics of the aircraft evaluating the lateral offset of the landing gear to the centerline $y_{\mathrm{LG,TD}}$, the sideslip angle of the landing gear $\beta_{\mathrm{LG,TD}}$, and the bank angle ϕ_{TD} at touchdown. Both cases are evaluated for two separate uncertain lateral closed loops, which are derived equivalently to the first two test cases, i.e crosswind and either tail- or headwind. However, for these analyses the aircraft, is externally disturbed by a turbulent crosswind. Note that the analysis inherently assumes that the maximum static wind profiles, i.e. the vertices of the allowed head-/tailwind and crosswind, lead to a worst-cases.

5.4.2 Analysis Interconnection and Setup

The general analysis interconnection for the LTV analyses is shown in Fig. 5.7. According to the individual test case and performance measure, the single blocks, the disturbance, and the performance signal must be adjusted. For the first two test cases, G_{wind} represents the block diagonal system $\begin{bmatrix} G_{u_w,\mathrm{LTV}} & \\ & G_{w_w,\mathrm{LTV}} \end{bmatrix}$ and Δ the dynamic 2×2 full-block LTI uncertainty. The remaining blocks, C the longitudinal autoland controller, W_{LTV} the 2×2 full-block time-varying uncertainty weight $W_{\mathrm{long},1}$ or $W_{\mathrm{long},2}$, and $G_{\mathrm{A/C}}$ the longitudinal LTV aircraft dynamics have to be adjusted chosen regarding the tail- or headwind scenario (case 1 or case 2, respectively). In the last two test cases, G_{wind} represents $G_{v_w,\mathrm{LTV}}$, C the lateral autoland controller, W_{LTV} the time varying SISO uncertainty weight $W_{\mathrm{lat},3}$ or $W_{\mathrm{lat},4}$, Δ the dynamic SISO LTI uncertainty, and $G_{\mathrm{A/C}}$ the lateral LTV aircraft dynamics. The test case-dependent blocks have to be chosen in an obvious fashion. respectively.

The LTV worst-case touchdown conditions are calculated applying the Log-L-SHADE algorithm from Section 4.6.4 on the nonlinear optimization problem (4.23). Therefore, the interconnection in Fig. 5.7 must be transferred into the LTV robustness analysis framework described in Section 4.3. In the longi-

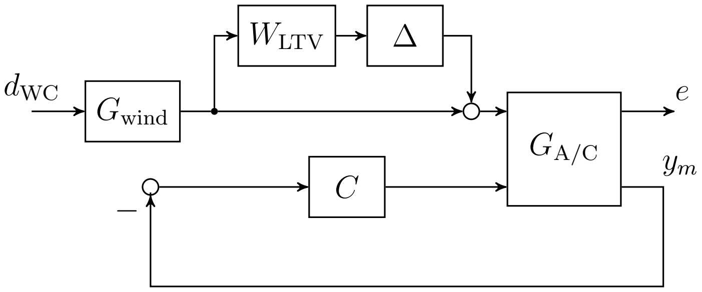


Figure 5.7: *General analysis interconnection*

tudinal analysis test cases (1 and 2), Δ is a 2×2 full block, dynamic LTI uncertainty. Its behavior is covered by the IQC described in Example 4.6.3, with $n_v = n_w = 2$. The MacMillan degree ν is chosen as one, and the value of ρ is -1.25. This means $\Delta \in IQC_1(\Psi_1, M_1)$, with M_1 restricted to the set $\mathcal{M} := \{M = \left[\begin{smallmatrix} X\otimes I_2 & 0 \\ 0 & -X\otimes I_2 \end{smallmatrix}\right] : X = X^T \geq 0 \in \mathbb{R}^{2\times 2}\}$ and $\Psi_1 = \left[\begin{smallmatrix} \psi_1\otimes I_2 & 0 \\ 0 & \psi_1\otimes I_2 \end{smallmatrix}\right]$. The same class of IQC is used to cover the behavior of the SISO, dynamic LTI uncertainty in the lateral analysis. Note in this case, the IQCs described in Examples 4.6.2 and 4.6.3 are identical. Here, $n_v = n_w = 1$, with the IQC factorization defined by $\nu = 1$ and $\rho = -0.75$. Hence, $\Delta \in IQC_2(\Psi_2, M_2)$, with M_2 restricted to the set $\mathcal{M} := \{M = \left[\begin{smallmatrix} X & 0 \\ 0 & -X \end{smallmatrix}\right] : X = X^T \geq 0 \in \mathbb{R}^{2\times 2}\}$ and $\Psi_2 = \left[\begin{smallmatrix} \psi_2 & 0 \\ 0 & \psi_2 \end{smallmatrix}\right]$.

The finite horizon worst-case $L_2[0, T]$ to Euclidean gain only bounds the Euclidian vector norm of the output performance signal over the disturbance inputs at the final time T. Therefore, for a given test case, each touchdown condition must be evaluated separately. Thus, the LTV worst-case gain optimization has to be executed ten times. In a combined analysis, i.e. one for each test case covering all performance output, the actual worst-cases of the single conditions would not be identified given the worst-case gains definition. The Log-L-SHADE is initialized with an initial population size of 40 in the longitudinal analysis and 20 in the lateral analysis. This difference is reasoned by the longitudinal analysis's larger IQC parameterization. However, all other settings are identical in both analyses. A total of 10 population iterations are conducted with a minimum population size of four. The logarithmic search

Table 5.4: Log-L-SHADE settings used in the longitudinal and lateral analyses

Case	$N_{\text{p,max}}$	$N_{\text{p,min}}$	$N_{\text{i,max}}$	m	n	k_{CR}	k_F	ϵ_{BS}	$\epsilon_{\text{ODE,rel}}$	$\epsilon_{\text{ODE,abs}}$
1&2	20	4	10	-7	1	5	5	10^{-3}	10^{-3}	10^{-6}
3&3	20	4	10	-7	1	5	5	10^{-3}	10^{-3}	10^{-6}

space's lower and upper bound are set to -7 and 1, respectively. The numbers of successful crossover rates k_{CR} and scaling factors k_F are both set to five. None of the analyses requires downscaling to facilitate the identification of a valid initial population. A relative tolerance of 10^{-3} is chosen for the bisection. The relative and absolute tolerance of `ODE15s` are set to 10^{-3} and 10^{-6}, respectively. Tab. 5.4 summarizes the settings used for both analysis.

Four separate Monte Carlo analyses, one for each test case, are necessary to evaluate all touchdown conditions of the full nonlinear closed-loop. Each test case is covered by 10000 samples defined in the parameter set in Tab. 6.1.

5.4.3 Results

The worst-case touchdown conditions calculated in the longitudinal plane for 10kts tailwind and 25kts crosswind (test case 1) are a vertical touchdown velocity $V_{z,\text{TD}}$ of 16.35ft/s and an AGL 60m behind the threshold H_{60} of 11.32m. Note that the LTV worst-case analysis only delivers the absolute value of the deviation from the design trajectory. Thus, the results have to be added/subtracted from the touchdown values of the recorded nominal trajectory used to derive the underlying nominal LTV model. In the proceeding, the total values are given so that they match the sign of the most critical value identified in the Monte Carlo simulation. The results are the absolute worst-cases $V_{z,\text{TD,WC}_1} = 20.8$ft/s and $H_{60,WC_1=1.9\text{m}}$. The most critical touchdown conditions found in the respective Monte Carlo analyses are $V_{z,\text{TD,MC}_1} = 15.7$ft/s and $H_{60,\text{MC}_1} = 2.25$m.

Afterwards, for the 30kts headwind scenario (test case 2), the same analyses are run. A total worst-case vertical touchdown velocity of 24.42ft/s, and a worst-case H_{60,WC_2} of -6.40m were calculated. Note that the value of H_{60,WC_2} indicates a short landing due to its negative sign. The corresponding Monte Carlo analysis provided a critical $V_{z,\mathrm{TD,MC}_2}$ of 22.62ft/s and a H_{60}, MC_2 of -0.48m. Hence, at least one short landing occurred, as the LTV analysis predicted. Note that the absolute LTV worst-case for the short landing ($H_{60,\mathrm{WC}_2} = -6.40$m) can only be interpreted by its sign and not its value. A (significantly) negative value indicates that the aircraft touched the runway noticeably before the threshold as it would (theoretically) below be below the surface at the threshold. All previously discussed results, including the reference touchdown conditions, are summarized in Tab. 5.5.

Table 5.5: Longitudinal analysis results

Analysis	Test Case 1		Test Case 2	
	$V_{z,\mathrm{TD}}$[ft/s]	H_{60}[m]	$V_{z,\mathrm{TD}}$[ft/s]	H_{60}[m]
Reference	4.5	13.25	4.89	4.7
LTV WC	16.35	11.32	19.53	11.13
Ref.+LTV	20.85	1.9	24.42	−6.40
Monte Carlo	15.699	2.25	22.62	−0.48

Subsequently, the lateral touchdown constraints are evaluated. For test case 3 (10kts tailwind), the LTV worst-case analysis delivered a total bank angle at touchdown $\phi_{\mathrm{TD,WC}_3}$ of 11.29°, a lateral offset to the centerline $y_{\mathrm{LG,TD,WC}_3}$ of 11.4m, and a worst-case sideslip angle of the landing gear $\beta_{\mathrm{LG,TD}_3}$ of 17.17°. Maximum values of 11.27°, 8.36m, and 11.13° for the bank angle, lateral offset, and sideslip angle, respectively, were identified in the corresponding Monte Carlo analysis. For test case 4 (30kts headwind), the LTV worst-case analyses calculated a total value of 11.01° for the bank angle, 14.46m for lateral offset, and 17.61° for the sideslip angle. The corresponding Monte Carlo analysis' results are 10.72° for the bank angle, 13.42m for the lateral offset, and 16.37° for the sideslip angle. In Tab. 5.6, the lateral analysis' results, as discussed, are summarized. It also provides the reference touchdown conditions used to calculate the LTV worst case values. In Fig. 5.10, the most critical

Table 5.6: Lateral analysis results

Analysis	Test Case 3			Test Case 4		
	ϕ_{TD} [°]	y_{TD} [m]	β_{LG_TD} [°]	ϕ_{TD} [°]	y_{TD} [m]	$\beta_{LG,TD}$ [°]
Reference	+0.02	−0.90	0.97	0.21	1.39	1.39
LTV WC	11.27	10.51	16.20	10.8	13.07	16.22
Ref.+LTV	11.29	11.40	17.17	11.01	14.46	17.61
Monte Carlo	11.27	8.36	11.13	10.72	13.42	16.37

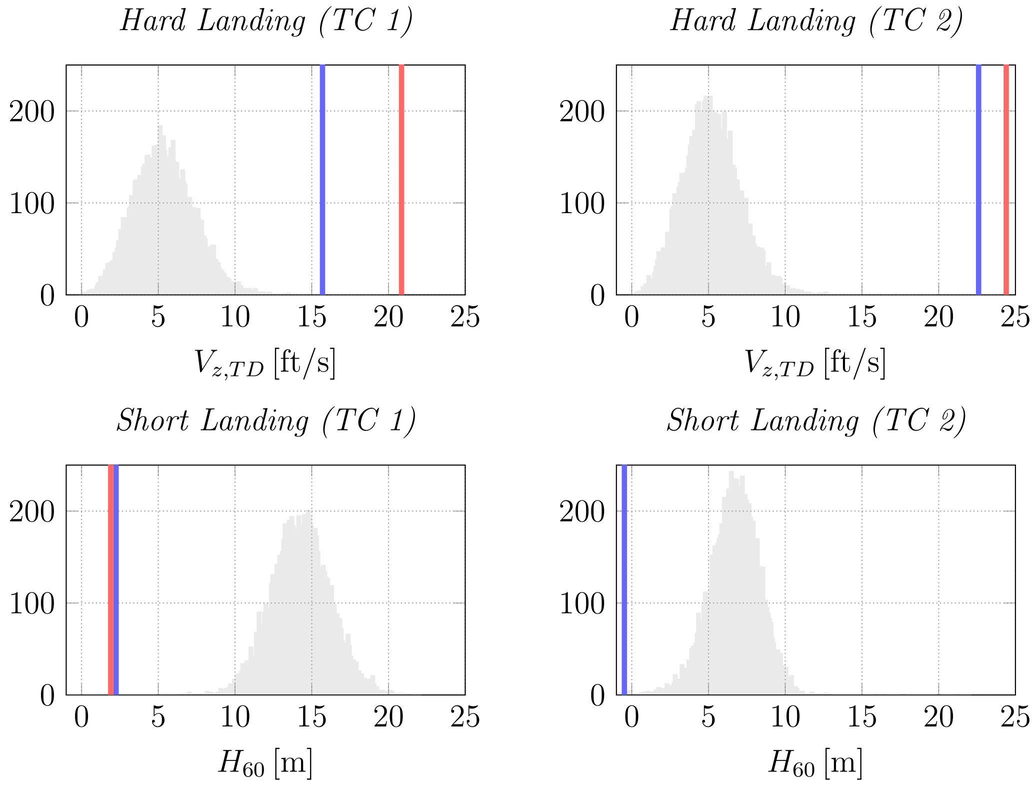


Figure 5.8: *Analysis results test case 1 (TC 2) and test case 2 (TC 2) : LTV worst case analysis (—), histogram Monte Carlo simulation (—), most critical Monte Carlo results(—)*

y_{LG} value identified in the Monte Carlo simulations for head- and tailwind are plotted against the corresponding LTV worst-cases. It can be seen that, the LTV worst case provides a not overly conservative upper bound for the

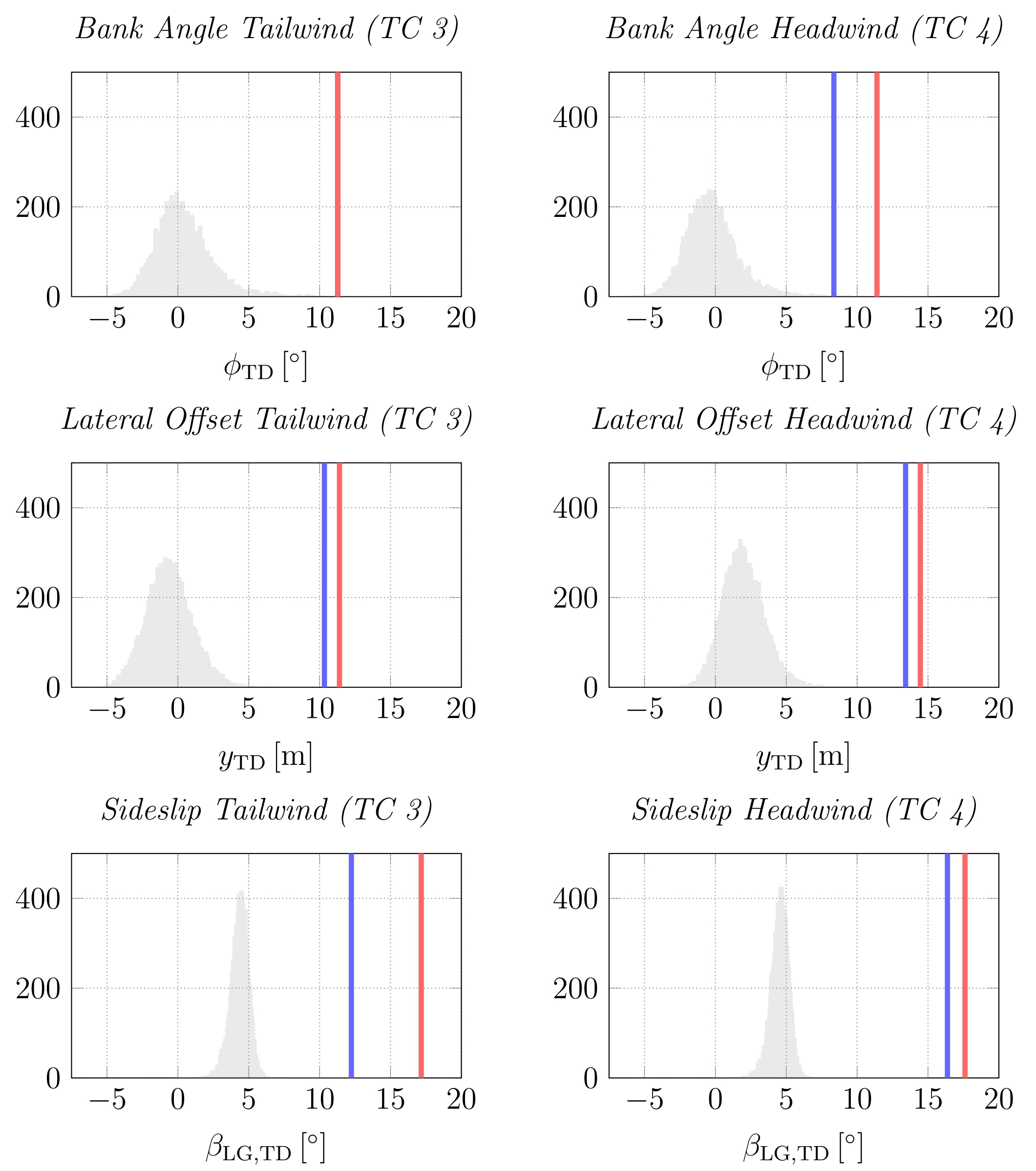


Figure 5.9: *Analysis results test case 3 (TC 3) and test case 4 (TC 4) : LTV worst-case analysis (▬), Histogram Monte Carlo simulation (░), most critical Monte Carlo results(▬)*

nonlinear simulation. Thus, each Monte Carlo analysis is upper bounded by the respective LTV worst-case. This is visualized in Fig. 5.8 and 5.9, showing the four Mont Carlo analyses' histograms and their most critical value and the individual LTV worst-cases. Notably, the ten LTV analyses were completed in 80min, which is eight times faster than the 640min required for the two Monte Carlo simulations, given the relatively small sample size of 10000. Also, there is no general rule on how large the sample size must be to draw conclusions for the design process. Therefore, the LTV analysis is more viable to assess the

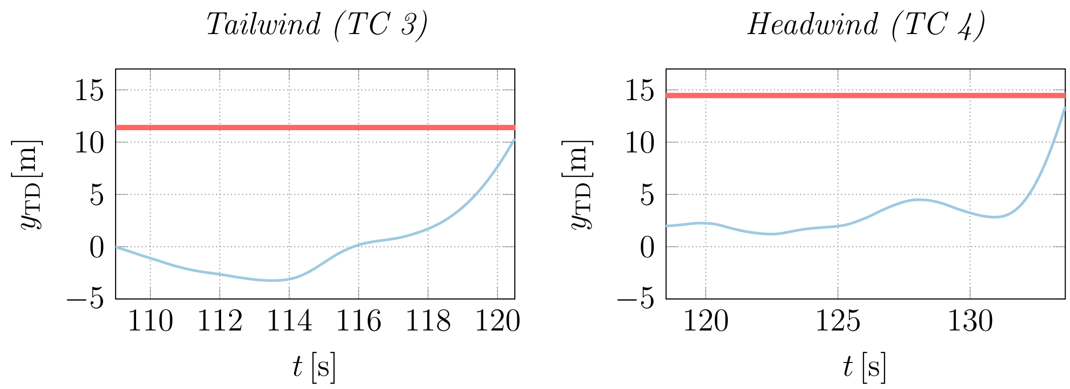


Figure 5.10: *Comparison of lateral offset from flare initiation to touchdown: Most critical nonlinear Simulation (—), LTV worst-case bound (━)*

qualitative impact of design changes in an iterative tuning process. All analyses were run on a standard desktop computer equipped with Intel i7 processor and 32GB memory. Furthermore, only the LTV worst-case analysis identifies (guaranteed) worst-cases, whereas the Monte Carlo analysis can only provide lower bounds on the touchdown conditions. Additionally, the distributions in Fig. 5.8 and 5.9 indicate that the latter requires large sample sizes to allow for meaningful conclusions on the most critical touchdown scenarios. Note that the touchdown constraints in [17] are based on probabilistic. Thus violations are allowed to a certain number of occurrences. Hence, the LTV analysis can quickly evaluate if a touchdown constraint is likely to be violated, allowing to adjust the design, avoiding extensive simulations.

5.5 Summary

The proposed robust LTV analysis of an autolanded aircraft provided fast upper bounds on worst-case touchdown conditions under crosswind. The common LPV aircraft representation was simplified to a special finite horizon LTV case, exposing the autoland scenario's characteristics. This allowed to explicitly respect the changing dynamics and control laws under the restriction of the final approach's finiteness. Feasible upper bounds for the Monte Carlo simulations conducted on corresponding the high-fidelity nonlinear model were provided by the LTV worst-case analysis in a fraction of time. Thus the proposed approach delivers a supplemental tool for the design process and evaluation of autoland controllers.

Chapter 6

Finite Time Horizon Analysis of a Launch Vehicle in Atmospheric Ascent

6.1 Introduction

A significant amount of time in the pre-launch preparation of expendable launch vehicles (ELVs) is spent optimizing the ascent trajectory and the respective tuning of the launcher's controller. One of the primary optimization objectives is reducing the maximal aerodynamic loads on the launcher due to wind disturbance. Additionally, most launch sites impose tight constraints on ascent corridors in the yaw plane to limit or avoid land overflight [10–12]. Hence, safety ranges must be included in the launch corridor design. At the same time, there are limits on the launcher's azimuth to reach the target orbit. In general, an excessive lateral drift in the atmospheric flight phase is energetically expensive to correct in the later flight phases, e.g. orbit injection, especially for polar orbits. Until hours before launch, updates are made based on wind data gathered by wind-balloons, etc. to identify a load minimizing trajectory based on gravity turns to provide enough safety range regarding land overflight [13].

Although these estimations are reasonably accurate for static wind profiles, it is hard to predict and account for turbulent wind fields' effects. Furthermore, there are no guarantees that the launcher model used for the calculations accurately matches the actual system. Focusing on the launcher's pitch motion, particularly perturbations in the launcher's thrust and mass profile, significantly influences the actual trajectory and, consequently, the occurring aerodynamic loads. Moreover, perturbations in the aerodynamic parameters and other perturbations such as time delays contribute to an overall degradation of the control performance.

Most state-of-the-art industrial approaches to evaluate the effects of turbulence and the estimated perturbation set utilize Monte Carlo analyses and worst-case optimizations conducted on nonlinear launcher models [1–3]. These methods can be directly deployed on the high-fidelity nonlinear launcher model. They require significant computational resources and time. Even more critical, they can only provide a lower bound on any worst-case performance measure. Therefore, a linear worst-case analysis for two critical launcher performance criteria, namely aerodynamic loads and lateral drift, is proposed in this chapter. It considers realistic wind disturbances and multiple perturbations, such as an accurately modeled trust and mass uncertainty. These analyses support the nonlinear Monte Carlo simulations by providing strict upper bounds.

However, existing linear worst-case analysis procedures are restricted to perturbations not primarily influencing the launch trajectory. Otherwise, the utilized linearization would lose its validity. Consequently, these analyses exclude thrust and mass uncertainties as they induce an increasing deviation from the planned trajectory in the pitch plane. Common approaches for linear analyses of launch vehicles pitch dynamics, see e.g. [204] or [3], treat thrust perturbations solely as an uncertainty in the thrust vectoring control (TVC). This procedure neglects the thrust profile's actual correlation with the launcher's weight and balance and the drift from the ascent trajectory. Thus, the most significant effects of these perturbations regarding the worst-case aerodynamic loads and controllability of the launcher are not covered.

For a more sophisticated linear worst-case analysis, in this thesis, the thrust is treated as an external input and the mass as a state in the linearization of the ELV's pitch dynamics along the pre-calculated trajectory. As a direct result, a perturbation of the nominal thrust in the LTV model describes the deviation from the design trajectory accurately, including the resulting aerodynamic load build-up. Because the exhaust mass flow directly connects the thrust and the mass, the direct coupling is maintained during the linearization. To cover the effects of the deviation from the design trajectory on the launcher's pitch dynamics, e.g. due to changing dynamic pressure, a dynamic uncertainty with time-varying weight is applied.

In contrast to the pitch plane, the thrust and mass perturbations do not directly influence the launcher's trajectory in the yaw plane. The lateral drift is only triggered by the wind disturbance. Thus, the additional thrust input and mass state are not included in the lateral analysis. However, the respective effects on the yaw dynamics of the launcher are covered using a dynamic uncertainty with time-varying weight.

Appropriate scaling and filtering of the disturbance inputs is necessary to restrict the analysis to realistic disturbance inputs. In case of the wind disturbance signal, the filter design approach from Chapter 5 is applied. The thrust disturbance is scaled using the $L_{2[0,T]}$ norm of the thrust uncertainty analyzed in the nonlinear simulation. Hence, it is guaranteed that the worst-case analysis covers the maximal allowable disturbance of the nonlinear system due to a variation in thrust norm-wise.

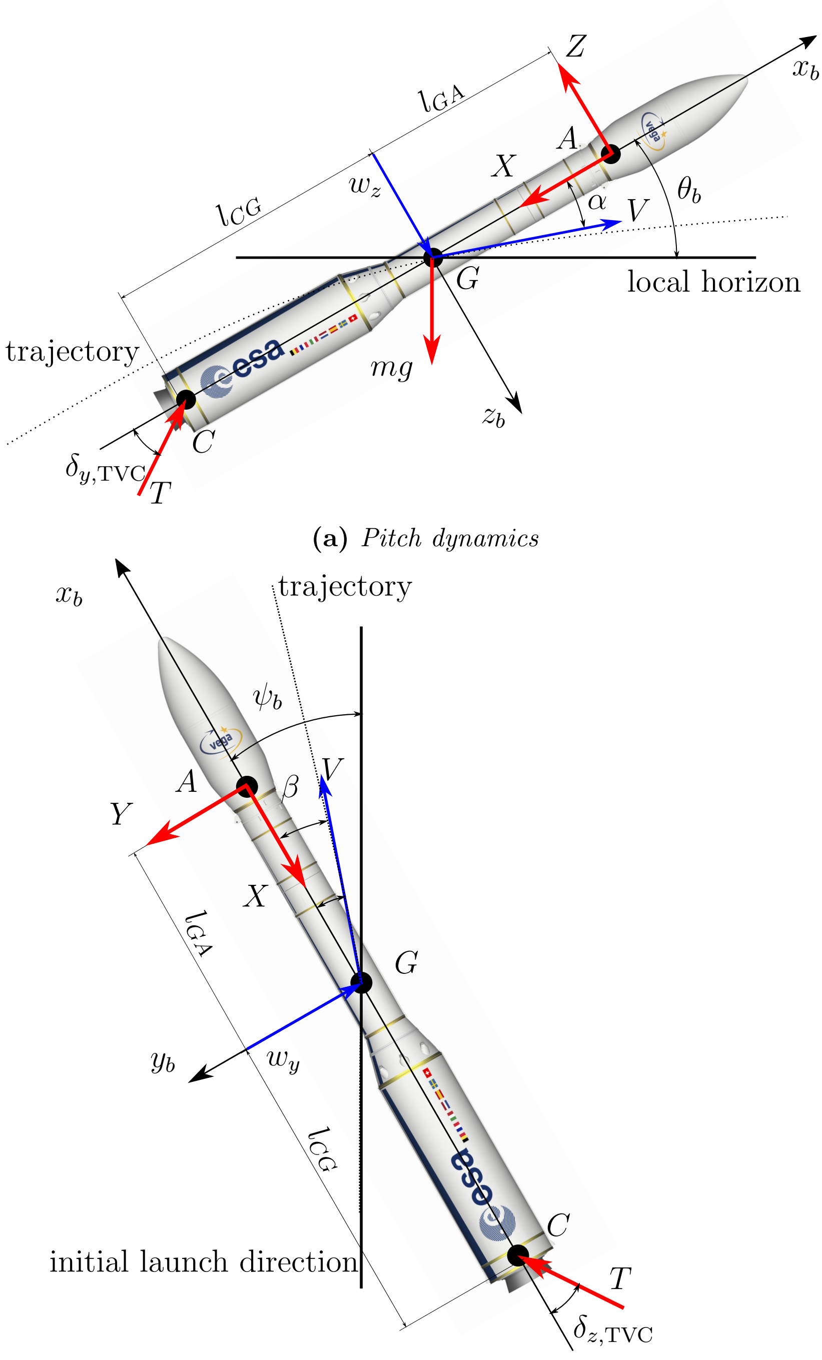
This chapter contributes a sophisticated worst-case aerodynamic loads analysis in the pitch plane and lateral drift analysis in the yaw plane of launch vehicles under wind disturbance and perturbations. Included is a novel approach to cover thrust and mass uncertainties in the pitch plane. Here, the thrust uncertainty is incorporated as an adequately scaled input disturbance, which is directly coupled with the launcher's mass. Contrary to the traditional approach using parametric uncertainties, the proposed method allows considering the perturbations' effects on the trajectory and states in the LTV analysis. This uncertainty description fits seamlessly into the overall LTV launcher analysis framework, as shown by extending the analysis with other types of perturbations. Here, all significant perturbations usually considered in the analysis of launch vehicles are covered. The applicability of the approach is demonstrated by comparison with the results of a random search conducted on the launcher's nonlinear model.

6.2 Launcher Model

Launch vehicles face stringent and complex performance requirements under a high amount of uncertainty. One of the most challenging mission segments is the atmospheric flight phase. Space launchers are aerodynamically highly unstable systems [250–252]. The high dynamic pressures aggravate this during the ascent leading to substantial aerodynamic loads accompanied by unsteady aerodynamics in the transonic region [253, 254]. The launcher is also subject to various disturbances. The most influential of these is wind [255].

6.2.1 Nonlinear Dynamics

An expandable launch vehicle built of 3 solid rocket motor (SRM) stages and a liquidly propelled upper module is investigated. The covered time segment spans from 25s to 95s after lift-off covering the launcher's atmospheric flight phase. It concludes with the burnout of the first SRM. During the atmospheric ascent, the space launcher can be treated as perfectly symmetric, with fully decoupled pitch and yaw dynamics [211, 252]. Thus, the pitch and yaw dynamics can be analyzed separately. Given this flight segment's overall duration and velocity regime, the earth can be assumed flat and non-rotating [209, 211]. Furthermore, only the launcher's rigid body motion is considered in this thesis. Based on the launcher's configuration, neither fuel sloshing nor TVC inertias are critical given their respective fractions on the overall system mass. Consequently, their influences are neglectable. These mainly affect the launcher's flexible modes, which are not a focal point in the static loads and drift analysis [210, 256]. An illustration of the launcher's dynamics is given in Fig. 6.1. It shows the pitch dynamics on the top and the yaw dynamics on the bottom. Forces are indicated by red and velocities by blue arrows.



(a) *Pitch dynamics*

(b) *Yaw dynamics*

Figure 6.1: *Expendable launch vehicle in body-fixed reference frame*

The launcher's nonlinear equations of motion (EoM) are defined with respect to its center of gravity G in a body-fixed coordinate system denoted by the subscript b. The x_b axis is aligned with the launcher's symmetry axis and is defined as positive in the direction of forward travel. The z_b axis is pointing downward, building a right-hand system with the y_b axis. The nonlinear EoM describing the launcher's pitch and yaw dynamics described by the set of equations:

$$
\begin{aligned}
\ddot{\theta}_b(t) &= \frac{\sum M_y(Ma,\alpha,h,t)}{J_{yy}(t,m)} \\
&= \frac{Z(Ma,\alpha,h,t)l_{GA}(t,Ma,m)}{J_{yy}(t,m)} - \dot{\theta}_b\frac{\dot{J}_{yy}}{J_{yy}} \\
&- \frac{T(t)l_{CG}(t,m)}{J_{yy}(t,m)}\sin\delta_{y,TVC}(t) \\
\ddot{\psi}_b(t) &= \frac{\sum M_z(Ma,\beta,h,t)}{J_{zz}(t,m)} \\
&= \frac{Y(Ma,\beta,h,t)l_{GA}(t,Ma,m)}{J_{zz}(t,m)} - \dot{\psi}_b\frac{\dot{J}_{zz}}{J_{zz}} \\
&- \frac{T(t)l_{CG}(t,m)}{J_{zz}(t)}\sin\delta_{z,TVC}(t) \\
\ddot{x}_b(t) &= \frac{\sum F_x(Ma,\alpha,h,t)}{m(t)} - \dot{\theta}_b(t)\dot{z}_b(t) \\
&= \frac{T(t)\cos\delta_{TVC}(t) - X(Ma,\alpha,h,t)}{m(t)} \\
&- g_0(h)\sin\theta_b(t) - \dot{\theta}_b(t)\dot{z}_b(t) \\
\ddot{y}_b(t) &= \frac{\sum F_y(Ma,\beta,h,t)}{m(t)} + \dot{\psi}_b(t)\dot{x}_b(t) \\
&= -\frac{Y(Ma,\beta,h,t)}{m(t)} - \frac{T(t)}{m(t)}\sin\delta_{z,TVC}(t) \\
&+ \dot{\psi}_b(t)\dot{x}_b(t), \\
\ddot{z}_b(t) &= \frac{\sum F_z(Ma,\alpha,h,t)}{m(t)} - \dot{\theta}_b(t)\dot{x}_b(t) \\
&= -\frac{Z(Ma,\alpha,h,t)}{m(t)} - \frac{T(t)}{m(t)}\sin\delta_{y,TVC}(t) \\
&+ g_0(h)\cos\theta_b(t) - \dot{\theta}_b(t)\dot{x}_b(t).
\end{aligned}
\tag{6.1}
$$

In (6.1), $\sum M_y$ and $\sum M_z$ are the sums of the angular moments around the pitch and yaw axis. Both are formulated with respect to the center of gravity G. The sum of forces in x_b, y_b, and z_b direction are denoted by $\sum F_x$, $\sum F_y$, and

$\sum F_z$, respectively. By θ_b, the launcher's pitch angle is denoted, describing the angle between the x_b axis and the local horizon. The notation for the launcher's yaw angle is ψ_b describing the angle between the x_b-axis and the initial launch direction. The axial, lateral, and vertical accelerations are denoted by $\ddot{x}_b$, $\ddot{y}_b$, and $\ddot{z}_b$, respectively. X, Y, and Z are the axial, lateral, and vertical aerodynamic forces, respectively. They are described by

$$X(Ma, \beta, h, t) = Q(h, t) S_{\text{ref}} C_X(\beta, Ma) \tag{6.2}$$

$$Y(Ma, \beta, h, t) = Q(h, t) S_{\text{ref}} C_Y(\beta, Ma) \tag{6.3}$$

and

$$Z(Ma, \alpha, h, t) = Q(h, t) S_{\text{ref}} C_Z(\alpha, Ma) \tag{6.4}$$

with

$$Q(h, t) = 0.5\rho(h, t) V(t)^2 \tag{6.5}$$

Accordingly, C_X, C_Y, and C_Z are the axial, lateral, and vertical aerodynamic force coefficients. These coefficients are nonlinear in the Mach number Ma and the angle of attack α or side-slip angle β. V is the absolute aerodynamic velocity of the ELV. The density of the air ρ is calculated according to the international standard atmosphere (ISA, [257]). In contrast to standard aerospace conventions, the aerodynamic forces are defined parallel to the respective body axis rather than the aerodynamic velocity. Here, the axial and vertical forces are defined in the negative axis direction and the lateral force in the positive axis direction. In (6.2) and (6.3), the angle of attack is approximated as

$$\alpha(t) \approx \frac{\dot{z}_b(t) - w_z(t)}{\dot{x}_b(t)}, \tag{6.6}$$

where w_z is the external wind disturbance. It is aligned with the z_b-axis and defined positive in z_b's direction. The sideslip angle in (6.3) is similarly defined as

$$\beta(t) \approx \frac{\dot{y}_b(t) - w_y(t)}{\dot{x}_b(t)}, \tag{6.7}$$

with w_y denoting the wind velocity in the positive y_b-axis direction.

The launcher's thrust T acts at the nozzle reference point C. It can be deflected by the angle $\delta_{y,\mathrm{TVC}}$ and $\delta_{z,\mathrm{TVC}}$ using the thrust vector control. Due to the characteristics of the solid rocket motor, the thrust follows a predefined profile. The thrust at particular point in time is defined by

$$T(t) = v_{\mathrm{ex}}(t)\dot{m}_{\mathrm{ex}}(t), \tag{6.8}$$

where v_{ex} is the exhaust velocity and $\dot{m}_{\mathrm{ex}}$ the exhaust mass flow of the engine. As a consequence of perturbations in the combustion process as well as tolerances in the packing process, T has a noticeable degree of uncertainty. Based on post-flight analysis of mission data, a common assumption is a constant thrust uncertainty of up to ±10%. The thrust profile (6.8) directly relates to the launcher mass m by m_{ex}. The mass of the launcher is given by

$$m(t) = m_0 - \int_0^{T_f} \dot{m}_{\mathrm{ex}} dt = m_0 - \int_0^{T_f} \frac{T(t)}{v_{\mathrm{ex}}(t)} dt. \tag{6.9}$$

It is assumed that the thrust uncertainty is purely a consequence of $\dot{m}_{\mathrm{ex}}$ and not v_{ex}. An uncertainty in T directly affects the mass (6.9) and indirectly through m the launcher's attitude and translation (6.1). Furthermore, the overall mass moment of inertias J_{yy} and J_{zz} defined with respect to G depends directly on the launcher's momentary mass. Hence, they are also indirectly affected by thrust disturbances. The same holds for the center of gravity, which is also a function of the launcher's mass. Consequently, the lever arms of the introduced thrust and aerodynamic forces l_{CG} and l_{GA}, defined as absolute distances between C and G, and G and A, respectively, are also affected by a thrust disturbance. Contrary to the mass, these perturbations mainly affect the launcher's controllability and instability rather than the trajectory. The altitude-dependent gravitational acceleration $g_0(h)$ is calculated based on the world geodetic system 84 (WGS 84, [239]), assuming a launch side at the equator.

6.2.2 Trajectory Calculation

The launcher performs a so-called gravity turn maneuver in the pitch plane, i.e. the centrifugal and the gravitational force on the launcher compensate each other. It minimizes the static aerodynamic loads and maximizes the longitudinal acceleration for the available amount of propellant. Under this assumption, the trajectory results from solving the initial value problem:

$$\begin{aligned} \dot{h} &= \dot{x}_b \sin\theta_b & \dot{N} &= \dot{x}_b \cos\theta_b \\ \ddot{x}_b &= \frac{T - X}{m} - g \sin\theta_b & \dot{\theta}_b &= -\frac{g}{\dot{x}_b} \cos\theta_b \end{aligned} \tag{6.10}$$

derived from the launcher's EoMs given by (6.1). In (6.10), h is the altitude, and N is the downrange distance. Solving (6.10) for a given h_0, θ_0, $\dot{x}_{b_0}$, and N_0 provides a pitch program for the launcher, which results in $\alpha \approx 0$ and $\delta_{TVC} \approx 0$ during the ascent. In the case of thrust perturbations, the equilibrium of gravitational and centrifugal force as assumed for (6.10) is no longer fulfilled if the pre-calculated pitch program is followed. The result are a build-up of α as well as a continuous deviation from the design trajectory. Note that no elaborate yaw program is designed. The launcher shortly after lift-off aligns with a fixed reference heading/yaw angle. Consequently, the thrust and mass deviations do not directly influence the equilibrium of forces in the yaw plane. Thus, no build-up of β is induced by said perturbations.

6.2.3 Linear Dynamics

The LTV worst-case analyses of the ELV require a linear representation of the ELV along the calculated gravity turn trajectory with a constant yaw angle of 0°. Thus, the nonlinear dynamics in (6.1) are linearized along this trajectory. The result is a finite horizon LTV system G_t as described by (3.7) in Section 3.1. Due to the decoupled pitch and yaw motion, this LTV model can be separated into two LTV models describing the pitch and yaw dynamics, respectively.

The LTV model $G_{\mathrm{LV,pitch}}$ applied in the aerodynamic loads analysis for the pitch plane is given in (6.11). In (6.11), the state vector is chosen as $x_{\mathrm{pitch}} = [\Delta\theta_b, \Delta\dot{\theta}_b, \Delta\dot{z}_b, \Delta\dot{x}_b, \Delta m]^T$, the input vector is $d_{\mathrm{pitch}} = [\Delta\delta_{y,\mathrm{TVC}}, \Delta w_z, \Delta T]^T$, and the output vector $e_{\mathrm{pitch}} = [\Delta\theta_b, \Delta Q\alpha]^T$. $Q\alpha$ is a measure for the static aerodynamic load and is defined as the product of the dynamic pressure Q and the angle of attack α. Δ refers to the deviation from the reference value on the design trajectory. It will be dropped in the equations and in the following to shorten the notation. The standard approach in the literature is to treat thrust and mass as parameters in the linearization, e.g. [204] or [3]. Thus, a thrust and mass perturbation can only be respected in the linear analysis by treating the respective reference values T_0 and m_0 in the system matrices as uncertain. Therefore, only its influence on the controllability via $\delta_{y,\mathrm{TVC}}$ can be covered.

In this thesis, the thrust is defined as an input and the mass as a state in the linearization. Therefore, the LTV model retains the inherent coupling between thrust and mass disturbance due to (6.8) and (6.9). If the thrust input is adequately scaled, it can accurately represent a thrust uncertainty in the nonlinear dynamics. Hence, the LTV description of the launcher in (6.11) presents a more accurate approximation of the ELV's nonlinear dynamics for the worst-case analysis than the standard literature approaches.

The lateral drift analysis in the yaw plane is described by the nominal LTV model $G_{\mathrm{LV,yaw}}$ in (6.12). In this case, the state vector is chosen as $x_{\mathrm{yaw}} = [\Delta\psi_b, \Delta\dot{\psi}_b, \Delta\dot{y}, \Delta y]$, the input vector as $d_{\mathrm{yaw}} = [\Delta\delta_{z,\mathrm{TVC}}, \Delta w_y]^T$, and the output vector as $e_{\mathrm{yaw}} = [\Delta\psi_b, \Delta y]^T$. Δy is the performance measure for the lateral analysis and describes the lateral deviation from the design trajectory. All coefficients in (6.11) and (6.12) are strictly time-dependent, omitted only to shorten the notation. The subscript 0 relates to the reference value on the nominal trajectory. Note that the explicit thrust input and mass state are omitted in the lateral LTV model as they do not directly influence the drift in the trajectory's yaw plane.

$$
\begin{bmatrix} \dot{\theta}_b \\ \ddot{\theta}_b \\ \ddot{z}_b \\ \ddot{x}_b \\ \dot{m} \end{bmatrix}
=
\begin{bmatrix}
0 & 1 & 0 & 0 & 0 \\
0 & -\frac{\dot{J}_{yy_0}(t)}{J_{yy_0}(t)} & \frac{Q_0(t) S l_{GA}(t)}{J_{yy}(t)\dot{x}_{b_0}(t)} \frac{\partial C_Z}{\partial \alpha}|_0(t) & 0 & -\dot{\theta}_b \frac{\partial(\frac{\dot{J}_{yy}}{J_{yy}})}{\partial m}(t) \\
-g_0(t)\sin\theta_{b_0}(t) & -\dot{x}_{b_0}(t) & -\frac{Q_0(t)}{m(t)\dot{x}_{b_0}(t)} \frac{\partial C_Z}{\partial \alpha}|_0(t) & \dot{\theta}_{b_0}(t) & 0 \\
-g_0(t)\cos\theta_{b_0}(t) & 0 & -\dot{\theta}_{b_0}(t) - \frac{Q_0(t) S}{m_0 \dot{x}_{b_0}(t)} \frac{\partial C_X}{\partial \alpha}|_0(t) & -\frac{Q_0(t) S}{m_0} \frac{\partial C_X}{\partial \dot{x}_b}|_0(t) & -\frac{T_0(t) - Q_0(t) S C_{X_0}(t)}{m_0^2(t)} \\
0 & 0 & 0 & 0 & 0
\end{bmatrix}
\begin{bmatrix} \theta_b \\ \dot{\theta}_b \\ \dot{z}_b \\ \dot{x}_b \\ m \end{bmatrix}
+
\begin{bmatrix}
0 & 0 & 0 \\
-\frac{T_0(t) l_{CG_0}(t)}{J_{yy_0}(t)} & \frac{Q_0(t) S l_{GA}(t)}{J_{yy}(t)\dot{x}_{b_0}(t)} \frac{\partial C_Z}{\partial \alpha}|_0(t) & 0 \\
-\frac{T_0(t)}{J_{yy_0}(t)} & \frac{Q_0(t) S}{m_0(t)\dot{x}_{b_0}(t)} \frac{\partial C_Z}{\partial \alpha}|_0(t) & 0 \\
0 & 0 & \frac{1}{m_0(t)} \\
0 & 0 & -\frac{1}{v_{ex,0}(t)}
\end{bmatrix}
\begin{bmatrix} \delta_{y,\mathrm{TVC}} \\ w_z \\ T \end{bmatrix}
$$

$$
\begin{bmatrix} \theta_b \\ Q\alpha \end{bmatrix}
=
\begin{bmatrix}
1 & 0 & 0 & 0 & 0 \\
0 & 0 & \frac{Q_0(t)}{\dot{x}_{b_0}(t)} & 0 & 0
\end{bmatrix}
\begin{bmatrix} \theta_b \\ \dot{\theta}_b \\ \dot{z}_b \\ \dot{x}_b \\ m \end{bmatrix}
+
\begin{bmatrix}
0 & 0 & 0 \\
0 & -\frac{Q_0(t)}{\dot{x}_{b,0}(t)} & 0
\end{bmatrix}
\begin{bmatrix} \delta_{y,\mathrm{TVC}} \\ w_z \\ T \end{bmatrix}
\tag{6.11}
$$

$$\begin{bmatrix} \dot{\psi}_b \\ \ddot{\psi}_b \\ \ddot{y}_b \\ \dot{y} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & -\frac{\dot{J}_{zz_0}(t)}{J_{zz_0}(t)} & \frac{Q_0(t)Sl_{GA}(t)}{J_{yy}(t)\dot{x}_{b_0}(t)}\frac{\partial C_Y}{\partial \beta}|_0(t) & 0 \\ 0 & \dot{x}_{b_0}(t) & -\frac{Q_0(t)S}{m(t)\dot{x}_{b_0}(t)}\frac{\partial C_Y}{\partial \beta}|_0(t) & 0 \\ \dot{x}_{b_0}(t)\cos\theta_0(t) & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} \psi_b \\ \dot{\psi}_b \\ \dot{y}_b \\ y \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ -\frac{T_0(t)l_{CG_0}(t)}{J_{zz_0}(t)} & \frac{Q_0(t)Sl_{GA}(t)}{J_{zz}(t)\dot{x}_{b_0}(t)}\frac{\partial C_Y}{\partial \beta}|_0(t) \\ -\frac{T_0(t)}{J_{zz_0}(t)} & \frac{Q_0(t)S}{m_0(t)\dot{x}_{b_0}(t)}\frac{\partial C_Y}{\partial \beta}|_0(t) \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \delta_{z,\mathrm{TVC}} \\ w_y \end{bmatrix} \tag{6.12}$$

$$\begin{bmatrix} \psi_b \\ y \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} \psi_b \\ \dot{\psi}_b \\ \dot{y}_b \\ y \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \delta_{z,\mathrm{TVC}} \\ w_y \end{bmatrix}$$

6.2.4 Modeling Mass and Thrust Uncertainty Effects via External Disturbance

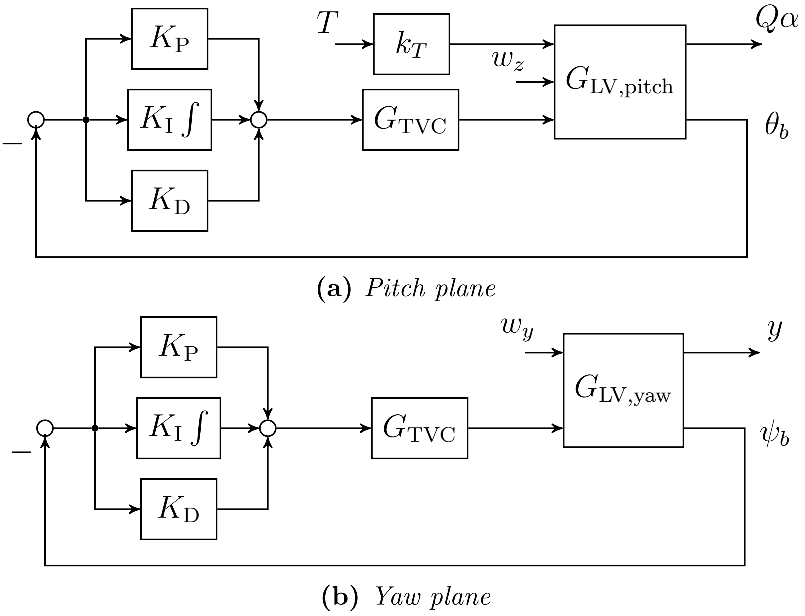
The application of the strict BRL restricts the search space of the worst-case disturbance signals only by the $L_{2[0,T]}$ norm, i.e. $\|d(t)_{\mathrm{WC}}\|_{2[0,T]} = 1$. Therefore, to calculate conclusive results, an adequate scaling of the thrust input in (6.11) is necessary to cover 10% thrust uncertainty. A practical scaling is the $L_{2[0,T]}$ norm of 10% nominal thrust for the given analysis horizon, i.e. $k_T = \|0.1T_0(t)\|_{2[0,T]}$, basically treating the thrust uncertainty as a form of energy disturbance. Hence, it is ensured that the LTV analysis covers the maximal thrust disturbance's norm/energy considered in the nonlinear analysis. Consequentially, the LTV worst-case analysis also provides an upper bound to the respective constant thrust disturbance due to the latter's norm's worst-case re-distribution. Nonetheless, this can lead to worst-case thrust disturbances, which temporarily exceed ±10%. The consequent increase in the LTV analysis' conservatism is acceptable as its primary purpose is to provide a reliable upper bound for the nonlinear analysis. This additional conservatism can even be regarded as profitable regarding the limited information about the actual thrust disturbance during the mission.

6.2.5 Augmentation

Feedback controllers are necessary to track the calculated pitch and yaw program and minimize the deviation from the pre-calculated trajectory. As both the pitch and yaw motion are aerodynamically unstable, the respective controllers further need to stabilize the ELV. Note that the pitch and yaw dynamics only differ by the gravity and centrifugal term immanent for the pitch dynamics, which offset in case of a gravity turn. Therefore, only one fixed-gain PID controller C was designed using the longitudinal dynamics at the point of maximum dynamic pressure $Q_{max} = 5.603 \cdot 10^4\mathrm{Pa}$ during the ascent. Employing loop-shaping, the proportional gain $K_{\mathrm{P}} = -4.81$, the integral gain $K_\mathrm{I} = -3.0175$, and the differential gain $K_\mathrm{D} = -1.1395$ were calculated, achieving 6dB gain and 40° phase margins for a maximum tracking bandwidth of 6rad/s. Thus, the controller satisfies recommended robustness margins for space applications [209, 256]. This controller can be directly applied to the launcher's yaw dynamic using ψ_b feedback to satisfies the same margins. The launcher model is extended with the second-order dynamics of the TVC:

$$G_\mathrm{TVC}(s) = \frac{1}{0.000374s^2 + 0.0384s + 1} \tag{6.13}$$

in the pitch and yaw channel. Fig. 6.2 shows the nominal closed loop systems for the pitch and yaw plane.



(a) *Pitch plane*

(b) *Yaw plane*

Figure 6.2: *Nominal launcher closed loops*

The nonlinear simulation for separate lateral and vertical wind disturbance ([258]) shows sufficient tracking of the pitch program ($|\Delta\theta_b \leq 0.1°|$) while maintaining the reference yaw angle. At the same time, the absolute values of the developing aerodynamic loads $|Q(t)\alpha(t)|$ and $|Q(t)\beta(t)|$ in the pitch and yaw plane never exceed the Mach dependent structural limit load under test scenarios suggested by ESA as shown in Fig. 6.3 [258].

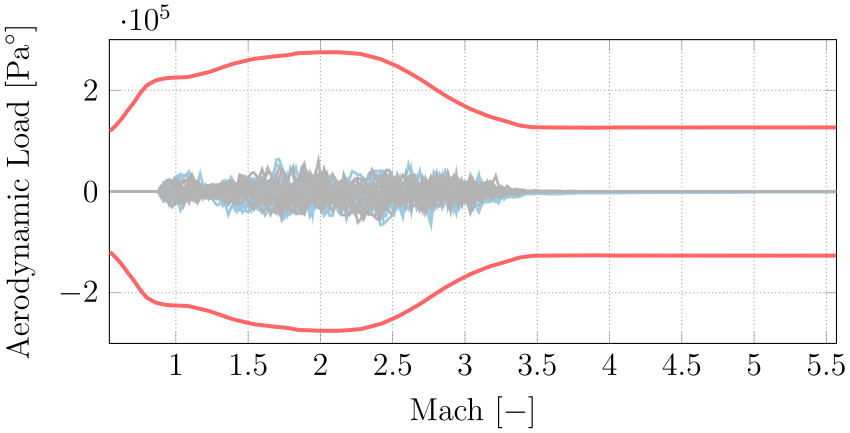


Figure 6.3: *Nominal evaluation of the yaw controller: $Q\alpha$ nominal launcher under wind disturbance (—), $Q\beta$ nominal launcher under wind disturbance (—), structural limit load (—)*

6.2.6 Uncertainty Model

The perturbations analyzed in this chapter can be categorized into two main categories. Firstly, the explicitly modeled perturbations/uncertainties in the parameters defining the launcher's and subsystems' dynamics. These include aerodynamic parameters, time delays, and higher order dynamics. Secondly, trajectory uncertainties are considered, which account for the perturbations of the launcher's dynamics due to the thrust and mass uncertainty induced deviation from the nominal trajectory. These implicitly cover the effects of uncertainties in the mass/weight and balance and controllability of the launcher. In combination, these two groups account for the majority of uncertainties recommended for launcher performance analyses, e.g. mass and balance, dynamic pressure, aerodynamics, and the thrust vectoring control system [3, 202, 204, 258, 259].

Launcher Uncertainty Model

A significant amount of uncertainty arises from the launcher's aerodynamic parameters, mainly due to the limited means of testing and the ensuing reliance on simulation. Furthermore, the launcher passes through the transonic ($0.8 \leq Ma \leq 1.2$), for which the calculation of aerodynamic parameters is complicated. One of the most challenging parameters to estimate is the center

Table 6.1: Explicit uncertainty set used for the robustness analysis

Parameter	Notation	Value	Occurrences	Type
l_{GA}	$\delta_{l_{GA}}$	20%	3	real
TVC	Δ_{TVC}	$\lvert\frac{0.18s+2.74}{s+56.01}\rvert$	1	dynamic
Delay	τ	22ms	1	-

of aerodynamic pressure. This is a consequence of the turbulent and complex airflow caused by the payload fairing. Due to its significant contribution to the launchers instability it is accounted for by an multiplicative input repeated LTI parametric uncertainty $\delta_{l_{GA}}$ for l_{GA} of 20%. It is included in both the aerodynamic loads and lateral drift worst-case analyses. Furthermore, the TVC's dynamics are treated as uncertain, primarily to account for the overall system's higher-order dynamics, which are not explicitly modeled. It is represented using dynamic LTI uncertainty Δ_{TVC} with $\|\Delta_{\text{TVC}}\|_\infty < 1$, which is implemented as

$$G_{\text{TVC}} = G_{\text{TVC,nom}}(1 + \Delta_{\text{TVC}} W_{\text{TVC}}), \tag{6.14}$$

with a weighting filter W_{TVC}. W_{TVC} is calculated based on the approach in [249]. It covers up to 5% uncertainty in the TVCs static gain $\delta_{\text{TVC},k}$, damping ratio $\delta_{\text{TVC},\zeta}$, and eigenfrequency $\delta_{\text{TVC},\omega}$. Finally, an explicit time delay of 22ms is included between the pitch/yaw controller and the TVC, which accounts for the control command's maximal computing time. All modeled perturbations are summarized in Tab. 6.1.

Trajectory Uncertainty Model

Following the introduction in Section 6.2.1, a thrust perturbation affects the nonlinear dynamics in (6.1) indirectly and directly. Accordingly, a variation from the nominal thrust profile leads to a continuous/steady deviation from the planned trajectory as the equilibrium conditions in (6.10) are violated, which by itself perturbs the nonlinear dynamics. It affects the (reference) values of mass and balance parameters like G, J_y, J_z and, m, controllability via l_{CG}, and T_0, trajectory-related parameters like $\dot{\theta}_b$, $\dot{x}_b$, and $\dot{z}_b$, and environmental parameters such as V and Q for a given point in time t. Thus, linearizing the nonlinear dynamics in (6.1) along such a perturbed trajectory results in LTV systems different to the ones in (6.11) and (6.12). Fig. 6.4 emphasizes the effects. It compares the bode magnitude plot of the nominal $\delta_{y,\text{TVC}}$ to θ_b open-loop transfer to the transfers belonging to trajectories resulting from constant thrust uncertainties in the range of ±10 at 75s after lift-off. This is emphasized in Fig. 6.5. It compares the singular values from ΔT and Δw_z to $Q\alpha$ for the nominal closed-loop dynamics to the singular values belonging to trajectories resulting from constant thrust uncertainties in the range of ±10 at 75s after lift-off. These perturbed dynamics must be respected in the LTV worst-case

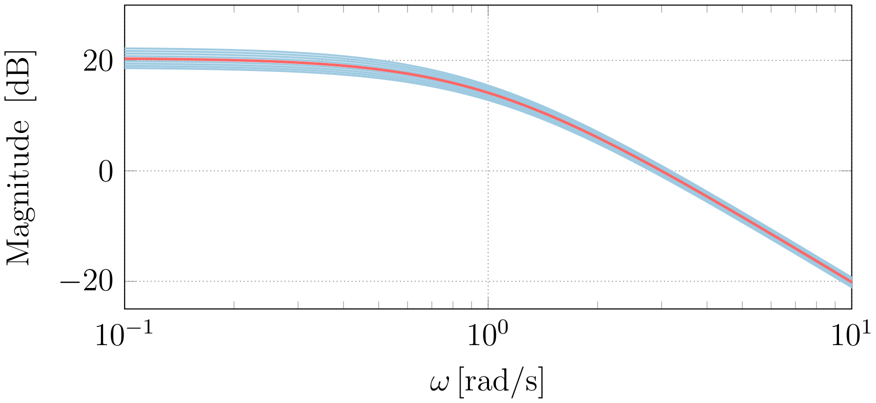


Figure 6.4: *Comparison of the transfer functions from $\delta_{y,TVC}$ to θ_b at 75s after lift-off: nominal dynamics (—), dynamics resulting from up to ±10% thrust disturbance (—)*

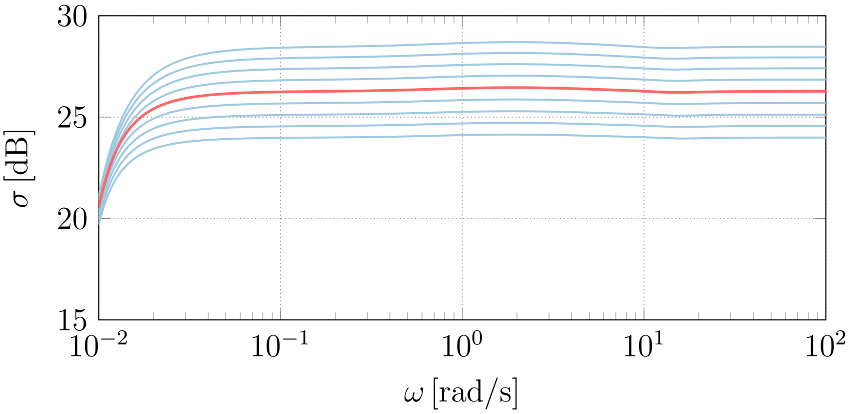


Figure 6.5: *Comparison of the singular values from ΔT and Δw_z to $Q\alpha$ at 75s after lift-off: nominal dynamics (—), dynamics resulting from up to ±10% thrust disturbance (—)*

analysis. Otherwise, comparing the linear and the nonlinear analysis would be of little significance. Comparable to Section 5.3.5, the following uncertain LTV representation of the launcher's closed-loop pitch and yaw dynamics is introduced, whose range of behaviors covers the dynamics along the perturbed trajectories.

$$G_{\text{CL}} = G_{\text{CL,nom}} + W_{\text{LTV}}(t)\Delta_{\text{traj}} \tag{6.15}$$

In (6.15), Δ_{traj} is a norm bounded dynamic LTI uncertainty, with $\|\Delta_{\text{traj}}\|_\infty \leq 1$, W_{LTV} is a time-varying shaping filter, and $G_{\text{CL,nom}}$ represents the nominal LTV closed-loop pitch or yaw dynamics. The weighting filter W_{LTV} is chosen based on the approach proposed in [249]. As it theoretically requires the underlying system to be stable, the closed-loop is chosen. Firstly, LTV closed-loop models of 8 perturbed trajectories covering a range of ±10% constant thrust uncertainty are generated. Subsequently, the weight is calculated at frozen times so that all perturbed models are included in the uncertainty set (6.15). Finally, the time varying-weighting is obtained by piecewise cubic polynomial interpolation. Note that thrust and mass disturbance are not explicitly modeled in the lateral LTV model (6.12) due to their limited direct effects on the lateral path. However, the trajectory uncertainty description is included in the lateral worst-case analysis. Thus, the effects on the launcher dynamics concerning the mass and balance, controllability, trajectory, and environmental conditions are covered.

6.2.7 Wind Disturbance Model

The analyzed wind disturbances in the aerodynamics loads and lateral drift analyses cover realistic wind profiles encountered during launcher missions. Therefore, dedicated shaping filters are designed according to Section 5.3.4 for the LTV and nonlinear analyses, shaping the respective input signals accordingly. As the most critical wind characteristics concerning the aerodynamic loads and lateral drift differ significantly, separate filters for both analyses are designed.

Wind Filter Aerodynamic Loads Analysis

The vertical component of a suitable wind profile for the considered trajectory from an equatorial launch side can be found in [204]. This (vertical) wind profile is calculated from the post-flight analysis of the Vega space launcher mission VV05 and shown in Fig. 6.6. The figure also shows the wind component which can be estimated pre-flight using data from air-balloons and launch side specific wind charts. [260] This wind profile can be respected in the reference trajectory calculation. However, this estimation will never perfectly match the actual wind-profile met by the launcher. Thus, potential differences must be accounted for in the worst-case analysis. To cover theses differences from the pre-flight wind profile, an approach based on spectral characteristics is proposed, i.e., bounding a specific power spectral density. Hence, an LTV wind filter is designed based on the representative wind profile's [204] offset to the pre-flight estimation and a specified safety margin. The procedure is almost identical to the one introduced in Section 5.3.4 In a first step, the offset wind signal (i.e., the wind velocity without the reference profile in Fig. 6.6) is divided into 5s segments from 25s to 65s. Based on the wind profile in Fig. 6.6, zero wind is assumed for times later than 65s after lift-off. Note that after 65s

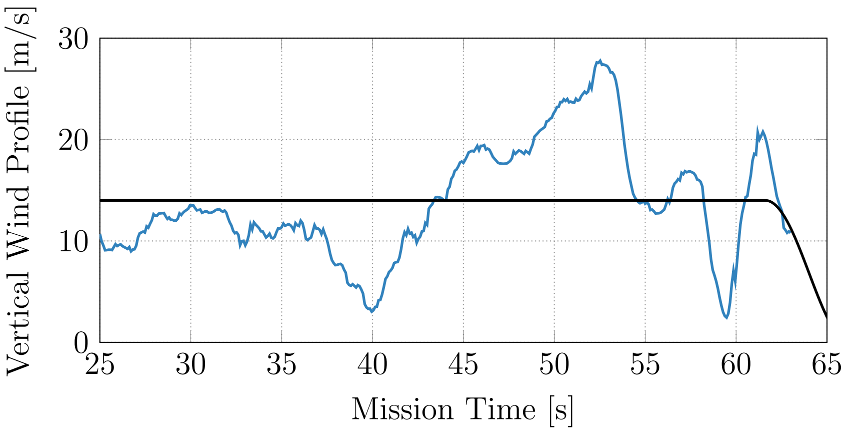


Figure 6.6: *Vertical wind profile Vega space launcher flight VV05: Pre-flight estimation used for trajectory/controller design (—), post-flight estimation (—)*

the launcher has ascended to such a high altitude that wind is no detectable which is established by the representative wind profile [204]. Based on the time history of the reference wind signal w_z, the PSD $\Omega_{w_{z,n}}$ of a segment n is defined by

$$\Omega_{w_{z,n}}(\omega) = \lim_{T_n \to \infty} \frac{2}{\pi} \frac{1}{T_n} \left| \int_0^{T_n} w_{z,n}(t) e^{-j\omega t} dt \right|^2 , \tag{6.16}$$

with T_n defining the time span of the segment, where $w_{z,n}(t)$ has been truncated to have zero value outside the range 0 to T_n. Hence, the PSD of a signal is determined by the average squared of its Fourier transform. In the present paper, the internal Matlab function `fft` is applied for this purpose using a sampling rate of 100Hz. This calculation is repeated for all segments n of the wind profile. In a second step, for each time segment, a transfer function upper bounding the respective $\sqrt{\left|\Omega_{w_{z,n}}\right|}$ and an added safety margin is calculated. For this purpose, the internal Matlab function `fitmagfrd` is applied, which determines a minimum phase transfer function using log-Chebychev magnitude design. The considered margin has a value of 12.5dB at $\omega = 1$rad/s and a value of 15dB at $\omega = 315$rad/s. This accounts for increasingly higher uncertainty with higher frequencies. The margin increases logarithmically between these boundaries. The fitted transfer functions are then transformed into consistent state-space models. In the third an final step, an LTV representation $G_{w_z,\mathrm{LTV}}$ of the wind filter is calculated by linear interpolating the system matrices' coefficients over the analysis horizon. In Fig. 6.7, the square roots of the PSD magnitudes and the fitted transfer function for the time segment from 55s to 60s is compared to the offset wind signal in this time span. In the Monte Carlo simulation a set of wind signals with PSDs comparable to the reference wind profile and upper bounded by the LTV wind filter is evaluated. PSD magnitudes of signals from this set are shown in gray (—) in Fig. 6.7.

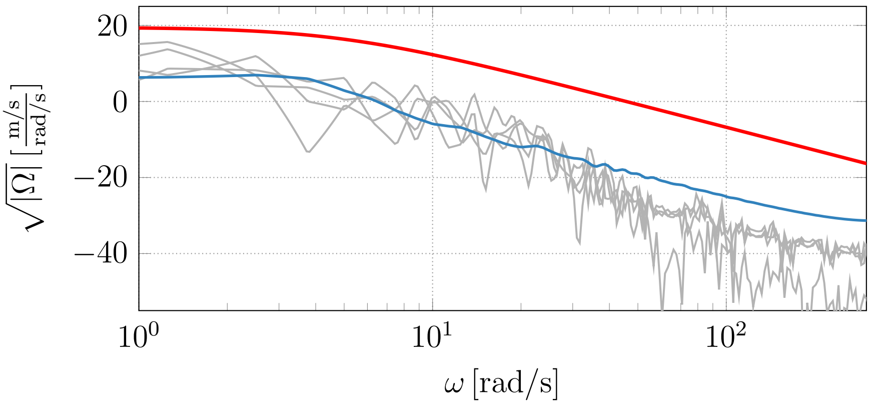


Figure 6.7: *Comparison of power spectral density magnitudes for the loads analysis segment covering 55s to 60s: LTV wind filter (—), reference wind signal (—), example signals from the set used in the nonlinear analysis (—)*

Wind Model Lateral Drift Analysis

The lateral component of the encountered wind profile must be used as the reference to derive the wind models for the yaw analysis. The approach to derive the wind model is the same as for the vertical wind profile. As the actual mission's data is restricted, a generic but applicable wind profile based on ESA practice is used as reference. It is displayed in Fig. 6.8 and is reasoned by the fact that, the launcher's lateral drift is more sensitive to low frequency, high amplitude wind disturbances. Due to the lower frequency content the safety boundaries are reduced to 5dB at 1rad/s and 10dB at 315rad/s. The yaw plane Monte Carlo wind signals follow the approach described for the pitch plane.

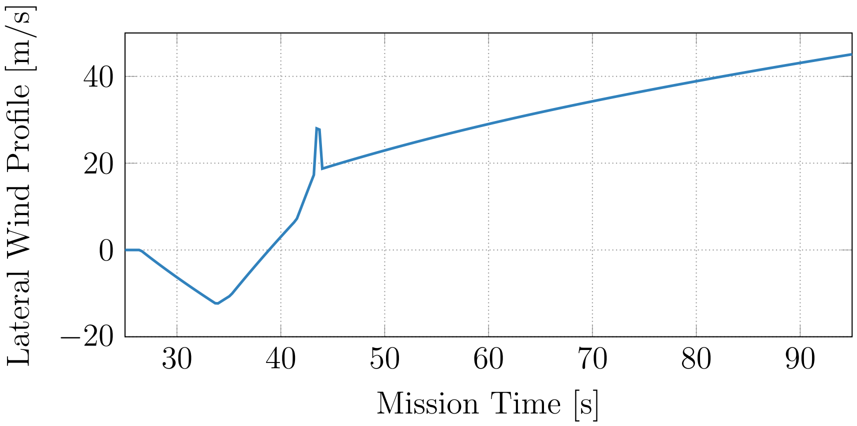


Figure 6.8: *Offset wind profile for lateral drift analysis*

6.3 Analysis

In the final section, the LTV worst-case analyses of the aerodynamic loads and lateral drift are presented. A direct evaluation against the results of the corresponding Monte Carlo simulations performed on the nonlinear is used to prove the approach's practicability. Before the analysis results are presented, the respective analysis setups and interconnections are introduced.

6.3.1 Analysis Interconnection

The general analysis interconnection used for both LTV worst-case analyses is displayed in Fig. 6.9. Note that the respective blocks need to be adjusted for the respective analysis. Hence, ELV represents the launcher's nominal LTV pitch closed loop dynamics or yaw dynamics as pictured in Fig. 6.2, extended with the respective LTV wind filter $G_{w_z,\mathrm{LTV}}$ and $G_{w_y,\mathrm{LTV}}$.

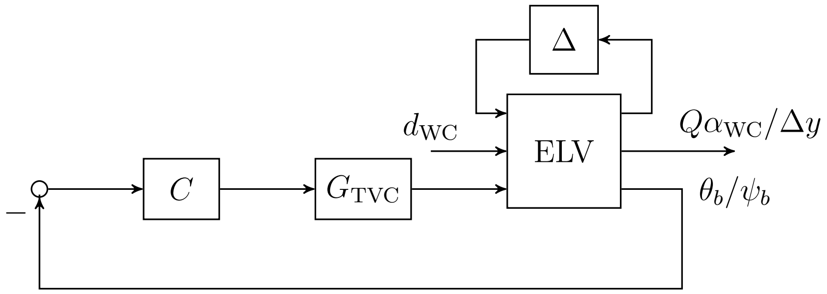


Figure 6.9: *Launcher interconnection for LTV worst-case analyses*

The underlying nominal LTV models $G_{\mathrm{LV,pitch}}$ and $G_{\mathrm{LV,yaw}}$ are computed via numerical linearization over the given analysis horizon $[25\mathrm{s}, 95\mathrm{s}]$ with a step size of 0.1s. For the loads analysis, the input signal $d_{\mathrm{WC}} = [d_{\mathrm{WC},w_z}, d_{\mathrm{WC},T}]^T$ represents the wind and thrust disturbance, respectively. In the lateral analysis, only the wind disturbance is included, and hence, $d_{\mathrm{WC}} = [d_{\mathrm{WC},w_y}]$.

The LTV worst case aerodynamic load $Q\alpha_{\mathrm{WC}}$ and lateral drift y are calculated using the Log-L-SHADE algorithm from Section 4.6.4 on the optimization problem (4.23) originating from Theorem 3. Therefore, the interconnection in Fig. 6.9 has to be transferred into the IQC framework described in Section 4.3. Here, the uncertainties introduced in Section 6.2.6 are covered by their respec-

Table 6.2: Parameters used for the IQC description

Uncertainty	b	ν	ρ
$\delta_{l_{GA}}$	0.2	1	-1
Δ_{TVC}	1	1	-1
Δ_{traj}	1	1	-1

tive IQC representations introduced in section 4.6.1. The repeated parametric uncertainty $\delta_{l_{GA}}$ is represented with the IQC in Example 4.6.1. For the dynamic uncertainty in the actuator and trajectory uncertainty description, the IQC description in Example 4.6.2. The specifically selected McMillan degrees ν, pole locations ρ, as well as norm bounds b, are summarized in Tab 6.2. In case of the time delay, the IQC from Example 3.7.2 introduced in Section 3.7.2 is applied. Note that this representation presents an upper bound on the time delay. Thus, all smaller time delays are inherently covered in the analysis.

6.3.2 Aerodynamic Loads Analysis

The first analysis calculates the aerodynamic loads in the pitch plane. It starts with LTV worst case loads using the analysis interconnection in Fig. 6.9 and IQC description following Tab. 6.2 described in Section 6.3.1. Subsequently, a corresponding Monte Carlo simulation is run on the corresponding nonlinear model of the launcher to validate the results of the LTV analysis.

LTV Worst-Case Aerodynamic Load Calculation

Recall, the finite horizon worst-case $L_{2[0,T]}$ to Euclidean gain only provides an upper bound on $Q\alpha_{\text{WC}}$ at the final time of the analysis horizon. Thus, it is inevitable to analyze a set of final times covering the trajectory. The presented analysis is conducted for final times T_i ranging from 30s to 95s with a step size of 5s. The required scaling of the thrust input $k_{T,i}$ is determined following the descriptions in Section 6.2.4 for a constant thrust uncertainty of 10%. As two disturbance inputs exists, the worst case disturbance signal's norm $\|d_{WC}\|_{2[0,T]}$ is distributed between the wind and thrust input. This distribu-

Table 6.3: Initial Log-L-SHADE settings used in the $Q\alpha$ and Δy analysis

Metric	$N_{\mathrm{p,max}}$	$N_{\mathrm{p,min}}$	$N_{\mathrm{i,max}}$	m	n	k_{IQC}	k_{CR}	k_F	ϵ_{BS}	$\epsilon_{\mathrm{ODE,rel}}$	$\epsilon_{\mathrm{ODE,abs}}$
$Q\alpha$	40	4	50	-14	8	0.4	5	5	10^{-5}	10^{-4}	10^{-6}
Δy	50	4	50	-14	8	0.1	5	5	10^{-5}	10^{-4}	10^{-6}

tion must be accounted for to reach the required disturbance levels. Hence, an additional scaling is introduced for the LTV loads analysis calculated, determined as follows. First, the $L_2[0,T]$ norms of the nominal LTV worst-case input's $d_{\mathrm{WC},i} = [d_{\mathrm{WC},w}, d_{\mathrm{WC},T}]^T$ components are calculated for final times T_i. Afterwards, these norm's inverses are used as the scalings $k_{T_{\mathrm{WC}},i}$ and $k_{W_{\mathrm{WC}},i}$ of the respective input channels for a given terminal time. This allows accounting for the distribution of the norms in the worst-case analysis.

The first run of the analysis calculating $Q\alpha_1$ for $T_1 = 30$s using the Log-L-SHADE algorithm is conducted with an initial downscaling of the uncertainty norm bounds k_{IQC} to 40%. As reasoned in Section 4.6.2, the downscaling simplifies and accelerates the identification of an initial solution set but reduces population iterations with the fully scaled uncertainty set. However, no degradation in the optimized $Q\alpha$ value was observed than in non-scaled optimization runs, but the initial population was identified five-times faster. The logarithmic search space's lower and upper bound are initialized with -13 and 8, respectively. An initial population size of 40 is selected, with a maximum number of 50 population iterations. The minimum population size is 4. For the numbers of successful crossover rates k_{CR} and scaling factors k_F, a value of five is chosen. A relative tolerance of 10^{-5} is chosen for the bisection. The bisection applies a lower bound of the nominal worst-case gain $\gamma_{\mathrm{nom},i}$ and an initial upper bound of $10^3 \cdot \gamma_{\mathrm{nom},i}$. The relative and absolute tolerance of `ODE15s` are set to 10^{-4} and 10^{-6}, respectively. Tab. 6.3 summarizes the settings used for the initial grid point's analysis.

Subsequent final times T_i points include the optimal solution of T_{i-1} in the initial solution set and narrow the search space to five magnitudes around this solution. Reasoned by Section 4.6.2, this procedure exploits that the optimal solutions of consecutive final times are relatively close. This observation also reasons the reduction of the maximal population size and number of iterations. Here, they are reduced by 30% compared to T_1. Hence, the overall computational effort for later grid points is noticeably reduced, speeding up the analysis.

The absolute values of the calculated worst-case $Q\alpha_{\text{WC}}$ are displayed in Fig. 6.10, compared to the Mach dependent limit load $Q\alpha_{\text{lim}}$ along the trajectory. Points in time in-between the analysis grid points are linear interpolated. A local peak of 140000Pa° (ca. 62% of $Q\alpha_{\text{lim}}$), at 30s, can be identified, before $Q\alpha_{\text{WC}}$ drops to 67750 (ca. 29% of $Q\alpha_{\text{lim}}$). This characteristic matches the expected wind turbulence, see Fig. 6.6. Afterwards, the aerodynamic load increases until reaching its maximum $Q\alpha_{\text{WC,max}}$ at 45s with a value of 223200Pa° (84% of the limit load). The $Q\alpha_{\text{WC}}$ values remain in this range up to 55s until they start to gradually decrease to 111200Pa° (around 66% of Q_{lim}) at 65s. A combination of three effects leads to the high $Q\alpha_{\text{WC}}$ in this trajectory phase. Firstly, the expected turbulence amplitudes are the highest in this region, as apparent from Fig. 6.6, causing high wind induced α disturbances. Secondly, the dynamic pressure increases to $Q_{\text{max}} = 56248$Pa at 51.2s due to the

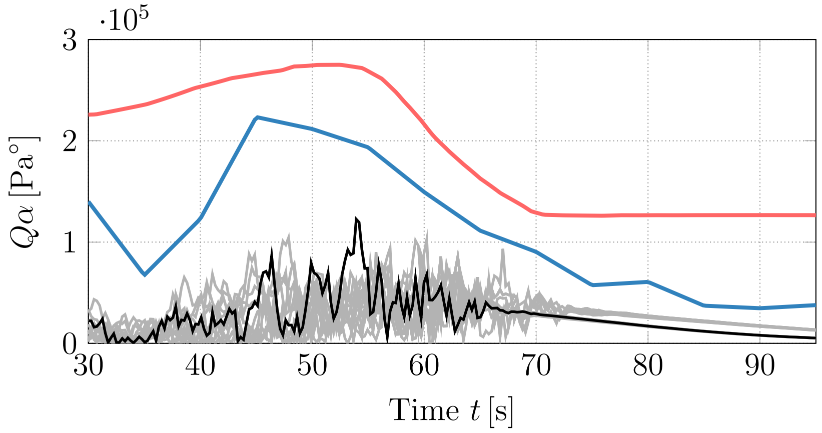


Figure 6.10: *Aerodynamic loads analysis results: $Q\alpha_{WC}$ bound LTV analysis (—), most critical Monte Carlo simulation $Q\alpha_{MC,WC}$ (—), selected critical $Q\alpha$ signals Monte Carlo simulation (—), Limit load $Q\alpha_{lim}$ (—)*

launcher's acceleration in the atmosphere's denser part. Thirdly, the drift from the planned trajectory induced by the thrust disturbance already built up an (additional) α disturbance. After 65s into the flight in an altitude of around 23km, the expected wind turbulence is almost diminished. From this point in time, only the steadily increasing deviation from the planned gravity-turn causes a α build-up contributing to $Q\alpha$. Although α is continuously increasing, the significant drop in dynamic pressure due to the thinning atmosphere decreases $Q\alpha_{\mathrm{WC}}$. The LTV worst-case analysis of the T_i grid was finished in 10h30min on a standard PC equipped with an Intel i7 and 32GB memory. This cumulative time includes necessary re-runs for the final times 45s and 75s, respectively, due to too narrowed search bounds.

Nonlinear Aerodynamic Load Calculation

Subsequently, it is validated if the LTV worst-case envelope presents a valid and practical upper bound for the launcher's nonlinear simulation. Therefore, a Monte Carlo simulation in Matlab Simulink is conducted. The required analysis interconnection for the nonlinear analysis is similar to Fig. 6.9. However, the G_{ELV} block now represents the launcher's nonlinear dynamics, and the thrust is explicitly implemented as an uncertain parameter in the simulation. Furthermore, the altitude scheduled wind filter G_{w_z} with white noise input introduced in Section 6.2.7 is applied to generate the wind disturbance. The simulation begins at $t_{\mathrm{s}} = 25\mathrm{s}$ and finishes at $t_{\mathrm{f}} = 95\mathrm{s}$ after lift-off. A sufficient sample size of perturbation and disturbance signal combinations is necessary to achieve meaningful results by Monte Carlo simulation. For the disturbance signals, this is accomplished by generating 500 unique white noise signals $n_{w,n}(t)$ applying the band-limited white noise block with unique noise seeds s_n. A gridding approach is used for model perturbations. Therefore, the thrust uncertainty is considered by five uniform points to cover $\pm 10\%$ uncertainty, l_{GA} by five uniform points covering $\pm 20\%$ uncertainty, and the single uncertainties in the actuator dynamics (used to generate W_{TVC}) by three uniform points each to cover 5% uncertainty. Furthermore, two time-delays of 11ms and 22ms are evaluated.

Now, each noise signal n_{w_n} is evaluated over all possible combinations in the generated uncertainty grid/set. The maximum aerodynamic load $Q\alpha_{\text{MC,WC}} = 1.304 \cdot 10^5 \text{Pa}^\circ$ (48% of $Q\alpha_{\text{lim}}$) is identified at 54.1s. It corresponds to the perturbation combination $\delta_T = 0.1$, $\delta_{l_{GA}} = +0.2$, $\delta_{\text{TVC},\omega} = -0.05$, $\delta_{\text{TVC},k} = \delta_{\text{TVC},\zeta} = 0.05$, and a time delay of 22ms. The signal $Q\alpha_{\text{MC,WC}}$ is shown in Fig. 6.10. Additionally, the figure shows a selection of $Q\alpha$ signals causing local peaks inside of 5s second intervals spanning from 30s to 95s. None of the $Q\alpha$ signals generated in the Monte Carlo simulation violated the LTV worst-case envelope. Furthermore, the general characteristic of the nonlinear analysis and LTV analysis match, i.e. local peak of $Q\alpha$ at the beginning of the trajectory, the region of the highest $Q\alpha$ values around $Q_{\max}$, and the decrease of the expected $Q\alpha$ and the drift's growing influence. A total of 675000 model evaluations with an average simulation time of 3.2s were required for the nonlinear analysis. This cumulated to an overall analysis time of 25d, which was effectively quartered to 6d6h, distributing the analysis between four computers equipped with Intel Xeon E-5 1620 v4 processors and 32GB memory. Thus, the nonlinear simulation took approximately 15 times longer than the LTV worst-case analysis providing a lower bound on $Q\alpha_{\text{WC}}$.

Both analyses match better in the later stages of the trajectory than at the beginning of the flight. The main reason is that the strict BRL also considers non-white noise signals, which in combination with $G_{w_z,\text{LTV}}$ result in potentially higher turbulence amplitudes. The decreasing expected turbulence levels reduce this effect for later times, improving the match. As in this phase, the influence of the thrust and mass uncertainty is most dominant. It suggests a good approximation of the trajectory deviation using ΔT as disturbance input combined with a weighted dynamic uncertainty.

6.3.3 Lateral Drift Analysis

Chapter 6 concludes with the analysis of the worst-case lateral drift of the space launcher under crosswind. After calculating the LTV worst-case, the results are validated via a Monte Carlo Simulation conducted on the nonlinear launcher model.

LTV Worst-Case Drift Analysis

As only the launcher's lateral deviation at the end of its planned trajectory is relevant, only one lateral LTV worst-case analysis run for the final time $T = 95$ is required. Furthermore, the lateral LTV analysis does not require input scaling as the thrust disturbance is not explicitly respected. Hence, only a single wind disturbance input exists, and no norm distribution has to be respected.

The LTV worst analysis is conducted with similar settings for the worst-case gain optimization used for the first grid point in the $Q\alpha_{\mathrm{WC}}$ calculation. Only the initial downscaling of the uncertainty norm bounds is changed to 10% and the initial population size increased to 50, as identifying an initial solution set is more complicated due to the substantially longer analysis horizon. Recalling Fig. 4.8 in Section 4.6.2, extending the analysis horizon reduces the feasible search space. Again, the nominal worst-case gain γ_{nom} is used as the lower bound and $1000\gamma_{\mathrm{nom}}$ as the upper bound for the bisection using identical tolerances as before. The Log-L-SHADE settings are summarized in Tab. 6.3. A worst-case lateral deviation from the planned trajectory at $95s$ after lift-off of 405.2m was calculated. In total, 7h30min were required to complete the analysis on the same computer as used for the $Q\alpha$ analysis. The disproportionately high computation time compared to the previous analysis originates from the inherently more challenging optimization problem. The longer (initial) analysis horizon can be identified, making the RDE harder to solve and narrowing the feasible search space. Simultaneously, the total number of required function evaluations over the full analysis horizon is significantly higher, as no initial guesses can be used to narrow the search space.

Nonlinear Drift Analysis

Subsequently, a corresponding Monte Carlo simulation is run to validate the LTV worst-case analysis results. Only the vertices of the uncertainty grid and maximum time delay used in the aerodynamic load analysis are evaluated as only one dedicated computer was available for the Monte Carlo analysis. Note that the thrust disturbance is included in the analysis as the effects are covered in the lateral trajectory uncertainty. Again 500 unique white noise signals $n_{w,i}$ are generated, which are separately evaluated for all uncertainty combinations. A maximum lateral deviation to the right of -146.6m and 113.2m to the left were identified. Both occurred for the uncertainty combination $\delta_T = 0.1$, $\delta_{l_{GA}} = +0.2$, $\delta_{\mathrm{TVC},\omega} = \delta_{\mathrm{TVC},\zeta} = -0.05$, $\delta_{\mathrm{TVC},k} = 0.05$, and a time delay of 22ms. The respective worst case trajectory is shown in Fig. 6.11. Hence, in a total of 16000 model evaluations, the LTV worst-case analysis provided a strict upper bound. In Fig. 6.12 the distribution of the lateral deviation at $T_\mathrm{f} = 95$s is shown. The nonlinear analysis finished 15h and 14min for a very coarse grid and small sample size. Note that this was purely due to the limitations in computational availability. However, due to the statistical nature of Fig. 6.11 and the sufficient margin, it can be concluded that the LTV analysis

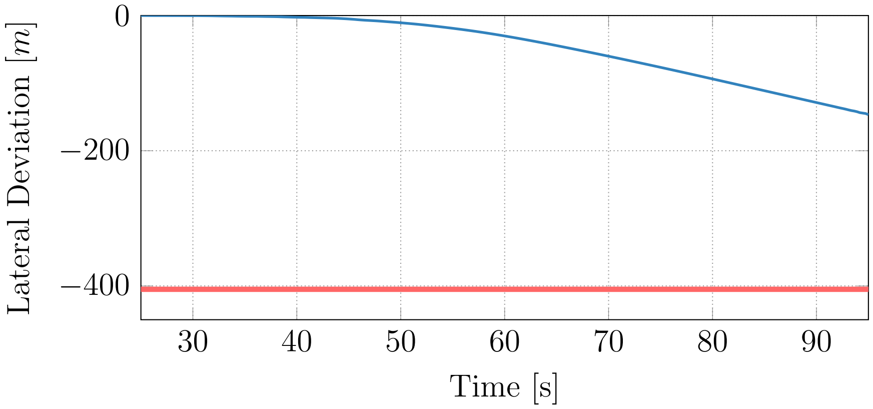


Figure 6.11: *Most critical trajectory with respect to lateral deviation: nonlinear analysis (——), upper bound provided by LTV worst-case analysis (——)*

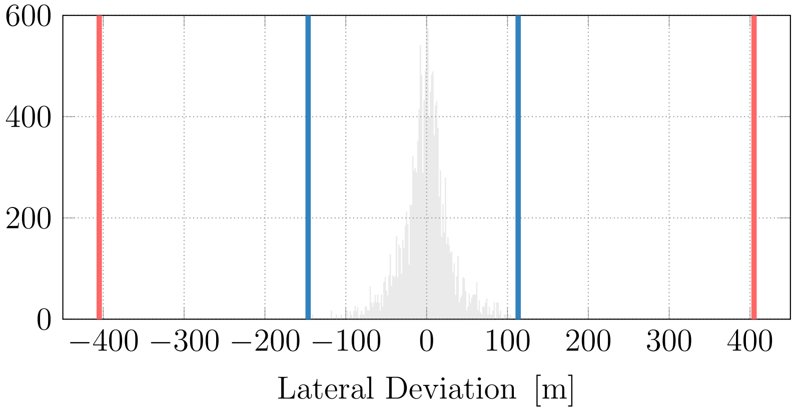


Figure 6.12: *Results lateral deviation analysis: Histogram Monte Carlo simulation (▨), most critical deviations Monte Carlo simulation (—), worst-case lateral deviation LTV analysis (—)*

provides a valid upper bound. The small sample size is also the main reason for the high conservatism of a factor of 2.76 concerning the LTV upper bound. Recall that a Monte Carlo simulation can also only provide a lower bound on the performance metric, which increases with larger sample size. Thus, the conservatism here can be interpreted as a form of safety margin. Hence, the LTV analysis provided a good insight into the maximal expected lateral deviation requiring significantly less time than an extremely coarse Monte Carlo simulation.

6.4 Summary

The presented LTV worst-case analyses of an ELV incorporate the influence of realistic wind disturbances and an elaborate uncertainty set covering coupled mass and weight and balance uncertainties. Unlike state-of-the-art approaches, the thrust uncertainty is included as an appropriately scaled input disturbance rather than a parametric uncertainty in the linear worst-case aerodynamic load analysis. Additionally, the LTV model is extended with an explicit mass state directly coupled with the thrust input. Thus, the thrust's direct influence on the trajectory and its indirect influence on it by affecting the mass state are covered. Therefore, the LTV pitch model's behavior fits the nonlinear model more precisely. Adding a weighted dynamic uncertainty to account for the dynamics of perturbed trajectory increases the match further and includes several other relevant uncertainties into the analysis. A similar trajectory uncertainty description is used in the lateral drift LTV analysis. Given this elaborate uncertainty model, the LTV worst-case analyses provide a feasible upper bound for the corresponding Monte Carlo simulation. Consequently, a fast and suitable complement in the validation and verification process of launch vehicles is provided.

Chapter 7

Conclusions and Future Work

The research in this thesis was based on the need for improved analytical tools in the V&V process for complex industrial applications. A significant inconsistency between the outputs of linear analysis and corresponding simulation-based methods is apparent, especially for systems with highly time-varying dynamics over finite horizons. This can result in both overly conservative limits as well as incorrectly satisfied safety margins, depending on the case. To bridge this gap, the development of novel reliable, computationally efficient, and not overly conservative worst-case analysis tools is required.

This thesis investigates this problem by developing a finite horizon linear time-varying (LTV) robustness analysis framework. It is specifically designed to cover long time horizons and provide fast guaranteed upper bounds on corresponding simulation-based approaches for the nonlinear model. A sophisticated benchmark example and two elaborate analysis examples with clear industry relevance were used to test the proposed framework thoroughly. A review of the results and thesis contributions as well as recommendations for future work are included in the following subsections.

7.1 Review of Thesis Aims and Achieved Results

The first target addressed in the thesis was the identification of existing LTV worst-case analysis methods. The bounded real lemma (BRL) for nominal LTV systems was identified as the most promising approach. It was subsequently implemented on a simple space launcher model. It showed improvement over a linear time-invariant analysis, which still (wrongfully) represents the industrial standard. This encouraging result led to the extension of this framework to IQCs to analyze LTV systems in interconnection with perturbations becoming a crucial aim. The nominal LTV BRL and the linear parameter-varying (LPV) BRL, including IQCs, were used as a theoretical foundation to formulate an extension of the LTV BRL to IQCs. It is built upon a solvability condition imposed on a Riccati differential equation (RDE). Based on these theoretical advances, an LTV robustness analysis framework was to be developed.

This was achieved by rendering the analysis conditions computationally feasible using fixed IQC factorizations and set constrained IQC parameterizations. The result was a nonlinear optimization problem, constrained by the RDE's solvability. For this optimization problem, two custom-tailored algorithms, namely Ab-SCA-PR and Log-L-SHADE, were developed.

Proof of feasibility for the new algorithms was demonstrated using various aerospace applications. The first algorithm's performance in a simple but relevant launcher robustness margin analysis was benchmarked against various off-the-shelf algorithms with a favorable outcome.

Having demonstrated functionality and improved performance using a simple model the next step was to expand the approach and evaluate its performance using complex and more industrially relevant examples. Using the second of the two algorithms developed, this evaluation involved its deployment on the following two examples. Example one was the autoland controller validation originating from the joint Airbus and ONERA landing challenge of the IFAC world congress. The second example involved a part of the certification procedure for the ESA's space launchers. Both examples were successful.

Due to this thesis's time-frame, it was not possible to address all aspects of the developed framework. However, the outstanding areas of investigation that have been identified to date are the subject of the remaining sections.

7.2 Conclusions

Suitability of State-of-the-Art Analysis Methods

LTI worst-case analyses are not suitable for systems with highly time-varying dynamics. The LTI analysis's limitation to fixed grid points entirely denies the trajectory characteristics and the system's physical behavior. It consistently fails to provide meaningful results using even the most rudimentary examples. In addition, it shows a large deviation from the results of non-linear analyses. For unstable motions, even for very large time constants, no worst-case gain can be calculated. Compounding this issue is the fact that the non-linear simulations indicated a significant performance gain. However, even for this simple example, the simulation-based approach is computationally significantly more expensive. This results in an inability to provide fast feedback during the design process. In summary, it can be concluded that it is imperative to develop specific linear methods for systems with time-varying characteristics. Furthermore, a time-varying system must only be analyzed with these specific methods.

Potential Gains from a Linear Time-Varying Analysis Framework

Only LTV methods can provide a valid, not overly conservative, upper bound for a time-varying nonlinear system over a finite horizon. Even a simple nominal LTV reduces the validation gap between linear and nonlinear analyses. Note that the respective scaling and filtering of the disturbance input significantly influence the analysis results. Thus, the correct input modification must be a centerpiece of any LTV analysis. The potential benefits regarding computational efficiency are even more significant in the presence of uncertainty, as the computational effort of simulation-based methods directly scales with the covered perturbation sets. Consequentially, the extension of the LTV framework to cover perturbed systems is imperative to advance the V&V process for space launchers and other highly time-varying systems.

Robust Linear Time-Varying Analysis Framework

A robust LTV analysis framework allowing for the worst-case gain calculation using IQCs can be readily derived based on recent advances in the robustness analysis of LPV systems. The analysis condition upper bounding uncertain LTV systems worst-case gain can be stated based on the solvability of an RDE. Using a standard approach to implement the IQCs, an optimization problem directly optimizing gamma over the IQC parameterization can be proposed. The RDE's solvability condition constrains it. This optimization problem can be readily solved by bisecting the worst-case gain for a temporarily fixed parameterization in an inner-loop. A global optimization identifies the optimal parameterization in an outer-loop related to the minimal worst-case gain.

Robustness Analysis of Finite Horizon Problems

The proposed robust LTV analysis framework also applies to systems where the performance metrics at the end of the trajectory are relevant. These cannot be analyzed with classic LTI methods. Besides the finite horizon characteristic of the problem, the proposed framework also covers time-varying dynamics and trajectory triggered control law changes, e.g. time or altitude related. Applied to an autolanded aircraft, it provides valid upper bounds on worst-case touchdown conditions under crosswind. It requires a fraction of the time compared to the corresponding simulation-based approach. Therefore, it can provide fast feedback in the (iterative) design process of autoland controllers.

In summary, the proposed LTV robustness framework delivers a supplemental tool for autoland controllers' design and validation process.

Robustness Analysis of Highly Time-Varying Problems

The presented analysis framework offers an efficient approach to calculate worst-case gains for an uncertain system with highly time-varying dynamics. These worst-case gains provide valid, not overly conservative upper bounds for simulation-based approaches. These upper bounds were successfully calculated for performance criteria of a detailed ESA space launcher model. The results indicate that the presented analysis framework is feasible for the rapid worst-case analysis of industry-sized problems, providing a valuable supplemental certification process tool.

7.3 Main Contributions

Finite horizon time-varying systems gathered little interest in the controls community until recently. Thus, elaborate analysis tools and procedures are scarce and have not been applied to examples of industrial complexity. The shortage of sophisticated worst-case analysis tools is apparent in the space sector, where knowingly unsuitable LTI tools are applied.

The work on the robustness analysis of finite-horizon systems conducted in this thesis, in the author's opinion, contributes to various advances in this research field:

1. The nominal bounded real lemma for finite horizon LTV systems is extended to IQCs, allowing for the worst-case analysis of uncertain systems with time-varying dynamics over finite horizons.

2. A novel LTV robustness analysis framework based on a nonlinear optimization problem, especially suitable for extensive analysis horizons is proposed.

3. In a detailed study, the question of how to efficiently solve the optimization problem by exploiting its specific characteristics and nonlinear constraints is explored. This study further delivers a general blueprint to design custom-tailored nonlinear programs for the worst-case optimization problem.

4. Two novel meta-heuristic optimization algorithms are proposed. Both algorithms are custom-tailored to solve the nonlinear optimization problem arising in the LTV robustness analysis.

5. A novel approach is developed for designing wind disturbance shaping filters from arbitrary reference signals while respecting the constraints of the BRL. These filters generate realistic wind disturbances and allow the calculation of worst-case gains directly comparable to the nonlinear simulation.

6. A novel method to respect coupled mass and thrust uncertainties in the LTV analysis is proposed. It models the mass as state and the thrust as an external disturbance in the linearization process. Thus, thrust/mass perturbation-induced deviations from the design trajectory are covered. In combination with a time-varying trajectory uncertainty, the corresponding nonlinear model's behavior can be adequately covered.

7. A detailed finite horizon robustness analysis of the worst-case touchdown conditions of an autolanded aircraft is conducted, covering various aircraft and environmental parameters. It provides valid upper bounds on the results of simulation-based analyses of the nonlinear model. An equivalent analysis or procedure cannot be found in the literature.

8. A detailed finite horizon robustness analysis of a space launcher's certification criteria during atmospheric ascent is conducted incorporating coupled mass and thrust uncertainties. It delivers strict upper bounds

on the results of simulation-based analyses of the nonlinear model. An LTV analysis for launch vehicles cannot be found in the literature.

7.4 Limitations and Future Work

Although the proposed analysis framework showed promising results, some limitations can be addressed in future research work. The nonlinear optimization problem is assumed non-convex as it is not possible to prove its convexity for more than one design variable. Thus, a global optimizer has to be employed to find the global minimum. However, no guarantee exists that the global minimum is found. An alternative approach leading to a convex optimization problem would be to use analysis conditions based on LMIs, see e.g. [238], or [152]. However, this requires the analysis grid's gridding and a definition of basis functions for P. Identifying the right approximations for both is difficult and not advisable for highly time-varying dynamics over vast horizons. Therefore, future research to improve the direct optimization approach could focus on convexifying the nonlinear problem.

Although the direct optimization based on the RDE's solvability scales better for large analysis horizons, solving the RDE is a main computational cost driver. Given a combination of a large analysis horizon and extensive IQC parameterization, the RDE becomes numerically hard to solve. Avoiding the direct integration using an estimate of the finite escape time could significantly reduce the bisection's computational cost. However, to the author's best knowledge, no feasible approach to avoid integration exists. Nevertheless, substantial theoretical work on the existence theory of autonomous RDEs could identify suitable existence guarantees.

Furthermore, the modeling of useful disturbance inputs under the restrictions of the BRL remains a vital issue. Although the design approach in the thesis produces sufficient filters, it cannot cover real turbulence as is possible in the H_2 framework for LTI systems. However, achieving a white noise disturbance input, if possible at all given the finite horizon, will likely require an approach differing from the BRL.

Trajectory uncertainties have a significant influence on the results, as seen in the mass and thrust uncertainty example. Hence, alternative and more comprehensive approaches to trajectory uncertainties should be pursued, ideally inside the robust LTV IQC framework.

Bibliography

[1] J. M. Hanson and B. B. Beard, “Applying monte carlo simulation to launch vehicle design and requirements verification,” *Journal of Spacecraft and Rockets*, vol. 49, no. 1, pp. 136–144, Jan. 2012.

[2] A. Marcos, H. G. D. Marina, V. Mantini, C. Roux, and S. Bennani, “Optimization-based worst-case analysis of a launcher during the atmospheric ascent phase,” in *AIAA Guidance, Navigation, and Control (GNC) Conference*. American Institute of Aeronautics and Astronautics, Aug. 2013.

[3] A. Kamath, P. Menon, M. Ganet-Schoeller, G. Maurice, S. Bennani, and D. Bates, “Worst case analysis of a launcher vehicle using surrogate models,” *IFAC Proceedings Volumes*, vol. 45, no. 13, pp. 260–265, 2012.

[4] J.-M. Biannic and C. Roos, “Flare control law design via multi-channel H_∞ synthesis: Illustration on a freely available nonlinear aircraft benchmark,” in *2015 American Control Conference (ACC)*. IEEE, Jul. 2015.

[5] G. Misra and X. Bai, “Updated simulation results of UAV carrier landings,” in *AIAA Scitech 2020 Forum*. American Institute of Aeronautics and Astronautics, Jan. 2020.

[6] H. Pfifer and P. Seiler, “Less conservative robustness analysis of linear parameter varying systems using integral quadratic constraints,” *International Journal of Robust and Nonlinear Control*, vol. 26, no. 16, pp. 3580–3594, Feb. 2016.

[7] A. Knoblach, H. Pfifer, and P. J. Seiler, “Worst case analysis of a saturated gust loads alleviation system,” in *AIAA Guidance, Navigation, and Control Conference*. American Institute of Aeronautics and Astronautics, Jan. 2015.

[8] “Statistical summary of commercial jet airplanes accidents 1959–2017,” The Boeing Company, Worldwide Operations, Seattle, Washington,D.C, Tech. Rep., Dec. 2018.

[9] J. W. Birkle, “Automatic approach tests of a type d autopilot in a canberra t.4 aircraft at woodbridge airfield, r.a.e. technical note no: Bl.41,” UK National Archives DSIR 23/24675, techreport 41, 1956.

[10] C. R. Kehler and F. R. Starbuck, “Eastern and western range (ewr) 127-1 range safety requirements,” 45th Space Wing and 30th Space wing, techreport, 1997.

[11] W. R. Monteith, "Ac 450.101-1 high consequence event protection," Federal Aviation Administration, techreport, Oct. 2020.

[12] ——, "Ac 450.115-1 high fidelity flight safety analysis," Federal Aviation Administration, techreport, Oct. 2020.

[13] G. Dukeman and A. Hill, "Rapid trajectory optimization for the ARES i launch vehicle," in *AIAA Guidance, Navigation and Control Conference and Exhibit.* American Institute of Aeronautics and Astronautics, Aug. 2008.

[14] F. Wu, "Control of linear parameter varying systems," Ph.D. dissertation, University of California, Berkeley, CA, Jan. 1995.

[15] G. Looye and H.-D. Joos, "Design of autoland controller functions with multiobjective optimization," *Journal of Guidance, Control, and Dynamics*, vol. 29, no. 2, pp. 475–484, Mar. 2006.

[16] "Air trafic by the numbers," Federal Aviation Administration, techreport, Jun. 2019.

[17] J.-M. Biannic and J. Boada-Bauxell, "A civilian aircraft landing challenge," ONERA, Toulouse, Tech. Rep., Jul. 2017, on-line available from the aerospace benchmark section of the SMAC Toolbox, http://w3.onera.fr/smac.

[18] J. Theis, D. Ossmann, F. Thielecke, and H. Pfifer, "Robust autopilot design for landing a large civil aircraft in crosswind," *Control Engineering Practice*, vol. 76, pp. 54–64, Jul. 2018.

[19] "Investment implications of the final frontier," Morgan Stanley Research, Investment Report 12, Oct. 2017.

[20] OECD, *The Space Economy in Figures: How Space Contributes to the Global Economy.* OECD Publishing, Paris, Jul. 2019.

[21] I. P. S. Skogestad, *Multivariable Feedback Control: Analysis and Design.* John Wiley and Sons, 2005.

[22] E. ROUTH, *Treatise on the stability of a given state of motion, particulary steady motion.* Place of publication not identified: Forgotten Books, 2016.

[23] F. Kuo, *Network analysis and synthesis.* New York: Wiley, 1966.

[24] J. Mawhin, "Alexandr mikhailovich lyapunov, thesis on the stability of motion (1892)," in *Landmark Writings in Western Mathematics 1640-1940.* Elsevier, 2005, pp. 664–676.

[25] A. M. Lyapunov, "The general problem of the stability of motion," *International Journal of Control*, vol. 55, no. 3, pp. 531–534, Mar. 1992.

[26] J. LaSalle, *Stability by Liapunov's direct method with applications.* Amsterdam: Elsevier Science, 1961.

[27] N. G. Chetaev, "On stable trajectories of dynamics," *Kazan University Science Notes*, vol. 4, no. 1, 1936.

[28] I. G. Malkin, *Theory of stability of motion. Translated from a publication of the State Publishing House of Technical-Theoretical Literature, Moscow-Leningrad, 1952.* U.S. Atomic Energy Commission, Office of Technical Information.

[29] C. Bissell, "Inventing the 'black box': mathematics as a neglected enabling technology in the history of communications engineering," 2001.

[30] K. Reinschke, *Lineare Regelungs- und Steuerungstheorie.* Springer Berlin Heidelberg, 2014.

[31] H. Nyquist, "Regeneration theory," *The Bell System Technical Journal*, vol. 11, no. 1, pp. 126–147, Jan 1932.

[32] C. Desoer and Y.-T. Wang, "On the generalized nyquist stability criterion," *IEEE Transactions on Automatic Control*, vol. 25, no. 2, pp. 187–196, Apr. 1980.

[33] J. Bongiorno and D. Graham, "An extension of the nyquist-barkhausen stability criterion to linear lumped-parameter systems with time-varying elements," *IEEE Transactions on Automatic Control*, vol. 8, no. 2, pp. 166–170, Apr. 1963.

[34] H. W. Bode, *Network Analysis andFeedback Amplifier Design.* D. Van Nostrand Co., Inc., 250 Fourth Ave., New York, 1945.

[35] V. A. Yakubovich, "The solution of some matrix inequalities encountered in automatic control theory," *Dokl. Akad. Nauk SSSR*, vol. 143, no. 6, pp. 1304–1307, 1962.

[36] T. Basar, "The solution of certain matrix inequalities in automatic control theory," in *Control Theory.* IEEE, 2009.

[37] R. E. Kalman, "LYAPUNOV FUNCTIONS FOR THE PROBLEM OF LUR'e IN AUTOMATIC CONTROL," *Proceedings of the National Academy of Sciences*, vol. 49, no. 2, pp. 201–205, Feb. 1963.

[38] F. Gantmacher and V. Yakubovich, "Absolute stability of nonlinear control systems," in *Proceedings of the Second All-union Congress on Theoretical and Applied Mechanics, Nauka, Moscow*, 1966.

[39] V. Popov, "Hyperstability and optimality of automatic systems with several control functions," *Rev Roumaine Si Techn Electrotechn Energ,*, vol. 9, no. 4, pp. 629–690, 1964.

[40] S. V. Gusev and A. L. Likhtarnikov, "Kalman-popov-yakubovich lemma and the s-procedure: A historical essay," *Automation and Remote Control*, vol. 67, no. 11, pp. 1768–1810, Nov. 2006.

[41] L. A.I., "Certain nonlinear problems in the theory of automatic control," *Original: Nekotorye Nelineinye Zadachi Teorii Avtomaticheskogo Regulirovaniya (Gos. Isdat. Tekh. Teor. Lit., 1951, U.S.S.R.), H.M. Stationery, Transl., 1957*, 1951.

[42] L. A.I. and P. VN, "On the theory of stability of control systems," *Appl Math Mech 8(3). (Prikl. Matem. i, Mekh., vol IX, no 5)*, 1945.

[43] J. C. Willems, "Dissipative dynamical systems part II: Linear systems with quadratic supply rates," *Archive for Rational Mechanics and Analysis*, vol. 45, no. 5, pp. 352–393, 1972.

[44] B. Anderson, "Algebraic description of bounded real matrixes," *Electronics Letters*, vol. 2, no. 12, p. 464, 1966.

[45] D. Clements and K. Glover, "Spectral factorization via hermitian pencils," *Linear Algebra and its Applications*, vol. 122-124, pp. 797 – 846, 1989, special Issue on Linear Systems and Control.

[46] S. Gugercin and A. C. Antoulas, "A survey of model reduction by balanced truncation and some new results," *International Journal of Control*, vol. 77, no. 8, pp. 748–766, May 2004.

[47] D. Clements, B. Anderson, and P. Moylan, "Matrix inequality solution to linear-quadratic singular control problems," *IEEE Transactions on Automatic Control*, vol. 22, no. 1, pp. 55–57, Feb. 1977.

[48] V. A. Yakubovich, "Singular problem in optimal control of linear stationary system with quadratic functional," *Siberian Mathematical Journal*, vol. 26, no. 1, pp. 148–158, 1985.

[49] K. Zhou, J. C. Doyle, and K. Glover, *Robust and optimal control.* Upper Saddle River, N.J: Prentice Hall, 1996.

[50] M. R. Liberzon, "Lur'e problem of absolute stability - a historical essay," *IFAC Proceedings Volumes*, vol. 34, no. 6, pp. 25–28, Jul. 2001.

[51] K. Narendra and R. Goldwyn, "A geometrical criterion for the stability of certain nonlinear nonautonomous systems," *IEEE Transactions on Circuit Theory*, vol. 11, no. 3, pp. 406–408, 1964.

[52] I. W. Sandberg, "A frequency-domain condition for the stability of feedback systems containing a single time-varying nonlinear element," *Bell System Technical Journal*, vol. 43, no. 4, pp. 1601–1608, Jul. 1964.

[53] G. Zames, "On the input-output stability of time-varying nonlinear feedback systems–part II: Conditions involving circles in the frequency plane and sector nonlinearities," *IEEE Transactions on Automatic Control*, vol. 11, no. 3, pp. 465–476, Jul. 1966.

[54] V. M. Popov, "Absolute stability of nanlinear syyterns of automatic control," *Auto. and Remote Control*, vol. 22, no. 8, pp. 857–875, Mar. 1962.

[55] T. Basar, “Absolute stability of nonlinear systems of automatic control (englishtranslation),” in *Control Theory.* IEEE, 2009.

[56] J. Moore and B. Anderson, “A generalization of the popov criterion,” *Journal of the Franklin Institute*, vol. 285, no. 6, pp. 488–492, Jun. 1968.

[57] R. Sepulchre, D. Aeyels, and J. Willems, “From the nyquist criterion to the circle criterion,” *IFAC Proceedings Volumes*, vol. 29, no. 1, pp. 1884–1889, Jun. 1996.

[58] R. O'Shea, “A combined frequency-time domain stability criterion for autonomous continuous systems,” *IEEE Transactions on Automatic Control*, vol. 11, no. 3, pp. 477–484, Jul. 1966.

[59] ——, “An improved frequency time domain stability criterion for autonomous continuous systems,” *IEEE Transactions on Automatic Control*, vol. 12, no. 6, pp. 725–731, Dec. 1967.

[60] J. Carrasco, M. C. Turner, and W. P. Heath, “Zames-falb multipliers for absolute stability: From o'shea's contribution to convex searches,” *European Journal of Control*, vol. 28, pp. 1–19, Mar. 2016.

[61] G. Zames and P. L. Falb, “Stability conditions for systems with monotone and slope-restricted nonlinearities,” *SIAM Journal on Control*, vol. 6, no. 1, pp. 89–108, Feb. 1968.

[62] R. Brockett and J. Willems, “Frequency domain stability criteria–part i,” *IEEE Transactions on Automatic Control*, vol. 10, no. 3, pp. 255–261, Jul. 1965.

[63] V. A. Yakubovich, “Frequency conditions for the absolute stability and dissipativity of control systems with a single differentiable nonlinearity,” *Sov. Math., Dokl.*, vol. 6, pp. 98–101, 1965.

[64] A. Megretski, “Combining l1 and l2 methods in the robust stability and performance analysis of nonlinear systems,” in *Proceedings of 1995 34th IEEE Conference on Decision and Control.* IEEE.

[65] G. Zames, “On the input-output stability of time-varying nonlinear feedback systems part one: Conditions derived using concepts of loop gain, conicity, and positivity,” *IEEE Transactions on Automatic Control*, vol. 11, no. 2, pp. 228–238, Apr. 1966.

[66] J. C. Willems, “Stability, instability, invertibility and causality,” *SIAM Journal on Control*, vol. 7, no. 4, pp. 645–671, Nov. 1969.

[67] B. Anderson, “The small-gain theorem, the passivity theorem and their equivalence,” *Journal of the Franklin Institute*, vol. 293, no. 2, pp. 105–115, Feb. 1972.

[68] C. Desoer and M. Vidyasagar, *Feedback Systems: Input–Output Properties.* Elsevier, 1975.

[69] I. Postlethwaite, J. Edmunds, and A. MacFarlane, “Principal gains and principal phases in the analysis of linear multivariable feedback systems,” *IEEE Transactions on Automatic Control*, vol. 26, no. 1, pp. 32–46, Feb. 1981.

[70] M. Safonov, A. Laub, and G. Hartmann, “Feedback properties of multivariable systems: The role and use of the return difference matrix,” *IEEE Transactions on Automatic Control*, vol. 26, no. 1, pp. 47–65, Feb. 1981.

[71] J. Cruz, J. Freudenberg, and D. Looze, “A relationship between sensitivity and stability of multivariable feedback systems,” *IEEE Transactions on Automatic Control*, vol. 26, no. 1, pp. 66–74, Feb. 1981.

[72] N. Lehtomaki, N. Sandell, and M. Athans, “Robustness results in linear-quadratic gaussian based multivariable control designs,” *IEEE Transactions on Automatic Control*, vol. 26, no. 1, pp. 75–93, Feb. 1981.

[73] J. Doyle, “Analysis of feedback systems with structured uncertainties,” *IEE Proceedings D Control Theory and Applications*, vol. 129, no. 6, p. 242, 1982.

[74] M. Safonov, “Stability margins of diagonally perturbed multivariable feedback systems,” in *1981 20th IEEE Conference on Decision and Control including the Symposium on Adaptive Processes.* IEEE, Dec. 1981.

[75] J. Doyle, J. Wall, and G. Stein, “Performance and robustness analysis for structured uncertainty,” in *1982 21st IEEE Conference on Decision and Control.* IEEE, Dec. 1982.

[76] A. Packard, M. Fan, and J. Doyle, “A power method for the structured singular value,” in *Proceedings of the 27th IEEE Conference on Decision and Control.* IEEE, Dec. 1988.

[77] M. Fan and A. Tits, “Characterization and efficient computation of the structured singular value,” *IEEE Transactions on Automatic Control*, vol. 31, no. 8, pp. 734–743, Aug. 1986.

[78] A. Packard and J. Doyle, “The complex structured singular value,” *Automatica*, vol. 29, no. 1, pp. 71–109, Jan. 1993.

[79] G. J. Balas, J. C. Doyle, K. Glover, A. K. Packard, and R. S. Smith, *The μ analysis and syntheses tool box*, MathWorks and MUSYN.

[80] P. Young, M. Newlin, and J. Doyle, “mu analysis with real parametric uncertainty,” in *Proceedings of the 30th IEEE Conference on Decision and Control.* IEEE, 1991.

[81] M. Fan, A. Tits, and J. Doyle, “Robustness in the presence of mixed parametric uncertainty and unmodeled dynamics,” *IEEE Transactions on Automatic Control*, vol. 36, no. 1, pp. 25–38, 1991.

[82] P. M. Young, M. P. Newlin, and J. C. Doyle, "Computing bounds for the mixed μ problem," *International Journal of Robust and Nonlinear Control*, vol. 5, no. 6, pp. 573–590, 1995.

[83] A. Megretski and A. Rantzer, "System analysis via integral quadratic constraints," *IEEE Transactions on Automatic Control*, vol. 42, no. 6, pp. 819–830, Jun 1997.

[84] S. Boyd, L. E. Ghaoui, E. Feron, and V. Balakrishnan, *Linear Matrix Inequalities in System and Control Theory.* Society for Industrial and Applied Mathematics, Jan. 1994.

[85] A. Megretski, C.-Y. Kao, U. Jönsson, and A. Rantzer, "A guide to iqc-beta: Software for robustness analysis," 1998.

[86] C.-Y. Kao, A. Megretski, U. Jonsson, and A. Rantzer, "A MATLAB toolbox for robustness analysis," in *IEEE International Conference on Robotics and Automation (IEEE Cat. No.04CH37508).* IEEE, 2004.

[87] C.-Y. Kao, A. Megretski, and U. Jönsson, "Specialized fast algorithms for IQC feasibility and optimization problems," *Automatica*, vol. 40, no. 2, pp. 239–252, Feb. 2004.

[88] F. Demourant, "New algorithmic approach based on integral quadratic constraints for stability analysis of high order models," in *2013 European Control Conference (ECC).* IEEE, Jul. 2013.

[89] J. C. Willems, "Dissipative dynamical systems part i: General theory," *Archive for Rational Mechanics and Analysis*, vol. 45, no. 5, pp. 321–351, 1972.

[90] U. Jönsson, "Stability analysis with popov multipliers and integral quadratic constraints," *Systems & Control Letters*, vol. 31, no. 2, pp. 85–92, Jul. 1997.

[91] M. Turner, M. Kerr, and I. Postlethwaite, "On the existence of stable, causal multipliers for systems with slope-restricted nonlinearities," *IEEE Transactions on Automatic Control*, vol. 54, no. 11, pp. 2697–2702, Nov. 2009.

[92] M. C. Turner and J. R. Forbes, "Conic sector analysis using integral quadratic constraints," *International Journal of Robust and Nonlinear Control*, vol. 30, no. 2, pp. 741–755, Nov. 2019.

[93] J. Shamma and M. Athans, "Analysis of gain scheduled control for nonlinear plants," *IEEE Transactions on Automatic Control*, vol. 35, no. 8, pp. 898–907, 1990.

[94] J. S. Shamma and M. Athans, "Guaranteed properties of gain scheduled control for linear parameter-varying plants," *Automatica*, vol. 27, no. 3, pp. 559–564, May 1991.

[95] N. M. Wereley, "Analysis and control of linear periodically time varying systems," Ph.D. dissertation, Dept. Aeron. and Astron., Massachusetts Institute of Technology, Boston, MA, Feb. 1991.

[96] D. A. Peters and K. H. Hohenemser, "Application of the floquet transition matrix to problems of lifting rotor stability," *Journal of the American Helicopter Society*, vol. 16, no. 2, pp. 25–33, Apr. 1971.

[97] D. Ossmann, J. Theis, and P. Seiler, "Load reduction on a clipper liberty wind turbine with linear parameter-varying individual blade pitch control," *Wind Energy*, vol. 20, no. 10, pp. 1771–1786, Jun. 2017.

[98] M. J. Balas and Y. J. Lee, "Controller design of linear periodic time-varying systems," in *Proceedings of the 1997 American Control Conference*, vol. 5, Jun 1997, pp. 2667–2671 vol.5.

[99] V. A. Yakubovich and V. M. Starzhinskii, *Linear Differential Equatians with Periodic Coefficients.* John-Wiley & Sons New York-Toronto, Israel Program for Scientific Translations, Jerusalem-London, 1975.

[100] J. A. Richards, *Analysis of Periodically Time-Varying Systems.* Springer Berlin Heidelberg, 1983.

[101] P. Montagnier, R. J. Spiteri, and J. Angeles, "The control of linear time-periodic systems using floquet–lyapunov theory," *International Journal of Control*, vol. 77, no. 5, pp. 472–490, Mar. 2004.

[102] D. H. Wu, "Development of an efficient computational technique for the analysis of linear dynamic systems withperiodically varying parameters." phdthesis, Auburn University, Alabama, USA, 1991.

[103] G. Floquet, "Sur les équations différentielles linéaires à coefficients périodiques," *Annales scientifiques de l'École Normale Supérieure*, vol. 2e série, 12, pp. 47–88, 1883.

[104] V. A. Yakubovich, "A remark on the floquet–lyapunov theorem," *Vestnik Leningrad University*, vol. 25, pp. 88–92, 1970.

[105] F. He, L. Wang, Y. Yao, H. Qi, and Z. Jiang, "Analysis and design of linear time-varying systems based on a finite-time performance index," in *2015 American Control Conference (ACC)*, July 2015, pp. 5948–5953.

[106] S. A. Melchior, P. V. Dooren, and K. A. Gallivan, "Finite horizon approximation of linear time-varying systems*," *IFAC Proceedings Volumes*, vol. 45, no. 16, pp. 734 – 738, Jul. 2012, 16th IFAC Symposium on System Identification.

[107] R. M. Murray, *A Mathematical Introduction to Robotic Manipulation.* Taylor and Francis Inc, 1994.

[108] A. Tewari, *Automatic Control of Atmospheric and Space Flight Vehicles: Design and Analysis with MATLAB and Simulink*, 1st ed. Birkhäuser Basel, Aug. 2011.

[109] G. V. Kamenkov, “On stability of motion over a finite interval of time [in russian],” *Journal of Applied Math. and Mechanics (PMM)*, vol. 17, pp. 529–540, 1954.

[110] K. Abgarian, “Stability of motion over a finite time interval,” *Journal of Applied Mathematics and Mechanics*, vol. 32, no. 6, pp. 999–1007, Jan. 1968.

[111] A. Lebedev, “On stability of motion during a given interval of time [in russian],” *Journal of Applied Math. and Mechanics*, vol. 18, pp. 139–148, 1954.

[112] P. Dorato, “Short-time stability in linear time-varying systems,” Ph.D. dissertation, Polytechnic Institute of Brooklyn, 1961.

[113] ——, “Short-time stability,” *IRE Trans. Automat. Contr.*, vol. 6, p. 86, 1961.

[114] H. D’Angelo, *Time-varying systems: analysis and design.* Allyn and-Bacon, Boston, MA, USA, 1970.

[115] L. Weiss and E. Infante, “Finite time stability under perturbing forces and on product spaces,” *IEEE Transactions on Automatic Control*, vol. 12, no. 1, pp. 54–59, Feb. 1967.

[116] F. Amato, M. Ariola, and P. Dorato, “Finite-time control of linear systems subject to parametric uncertainties and disturbances,” *Automatica*, vol. 37, no. 9, pp. 1459–1463, Sep. 2001.

[117] P. Dorato, “An overview of finite-time stability,” in *Systems and Control: Foundations & Applications.* Birkhäuser Boston, 2006, pp. 185–194.

[118] G. Tadmor, “Worst-case design in the time domain: The maximum principle and the standard Hinf problem,” *Mathematics of Control, Signals, and Systems*, vol. 3, no. 4, pp. 301–324, Dec. 1990.

[119] D. J. N. Limebeer, B. D. O. Anderson, P. P. Khargonekar, and M. Green, “A game theoretic approach to Hinf control for time-varying systems,” *SIAM Journal on Control and Optimization*, vol. 30, no. 2, pp. 262–283, Mar. 1992.

[120] R. Ravi, K. M. Nagpal, and P. P. Khargonekar, “Hinf control of linear time-varying systems: A state-space approach,” *SIAM Journal on Control and Optimization*, vol. 29, no. 6, pp. 1394–1413, Nov. 1991.

[121] G. Tadmor, “Input/output norms in general linear systems,” *International Journal of Control*, vol. 51, no. 4, pp. 911–921, Jan. 1990.

[122] D. Gilbarg and N. S. Trudinger, “The classical maximum principle,” in *Grundlehren der mathematischen Wissenschaften.* Springer Berlin Heidelberg, 1977, pp. 30–49.

[123] H. Kwakernaak, *Linear optimal control systems.* New York: Wiley Interscience, 1972.

[124] D. P. Reid, *Riccati Differential Equations.* Academic Press, 1972.

[125] H. W. Knobloch and M. Pohl, "On riccati matrix differential equations," *Results in Mathematics*, vol. 31, no. 3, pp. 337–364, May 1997.

[126] H. Abou-Kandil, G. Freiling, V. Ionescu, and G. Jank, *Matrix Riccati Equations in Control and Systems Theory.* Birkhäuser Basel, 2003.

[127] M. Green and D. J. N. Limebeer, *Linear Robust Control.* Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 1995.

[128] W. Chen and F. Tu, "The strict bounded real lemma for linear time-varying systems," *Journal of Mathematical Analysis and Applications*, vol. 244, no. 1, pp. 120–132, Apr. 2000.

[129] J. Buch, M. Arcak, and P. Seiler, "An efficient algorithm to compute norms for finite horizon, linear time-varying systems," *IEEE Control Systems Letters*, vol. 5, no. 5, pp. 1597–1602, Nov. 2020.

[130] G. Balas, I. Fialho, A. Packard, J. Renfrow, and C. Mullaney, "On the design of LPV controllers for the f-14 aircraft lateral-directional axis during powered approach," in *Proceedings of the 1997 American Control Conference (Cat. No.97CH36041).* IEEE, 1997.

[131] I. Postlethwaite, I. K. Konstantopoulos, X.-D. Sun, D. J. Walker, and A. G. Alford, "Design, flight simulation, and handling qualities evaluation of an LPV gain-scheduled helicopter flight control system," *European Journal of Control*, vol. 6, no. 6, pp. 553–566, Jan. 2000.

[132] C. Weiser, D. Ossmann, and G. Looye, "Design and flight test of a linear parameter varying flight controller," *CEAS Aeronautical Journal*, vol. 11, no. 4, pp. 955–969, Aug. 2020.

[133] H. Pfifer, C. P. Moreno, J. Theis, A. Kotikapuldi, A. Gupta, B. Takarics, and P. Seiler, "Linear parameter varying techniques applied to aeroservoelastic aircraft: In memory of gary balas," *IFAC-PapersOnLine*, vol. 48, no. 26, pp. 103–108, 2015.

[134] J. Theis, H. Pfifer, and P. J. Seiler, "Robust control design for active flutter suppression," in *AIAA Atmospheric Flight Mechanics Conference, AIAA SciTech Forum*, 2016.

[135] G. J. Balas, "Linear, parameter-varying control and its application to a turbofan engine," *International Journal of Robust and Nonlinear Control*, vol. 12, no. 9, pp. 763–796, 2002.

[136] I. R. Petersen and C. V. Hollot, "A riccati equation approach to the stabilization of uncertain linear systems," *Automatica*, vol. 22, no. 4, pp. 397–411, Jul. 1986.

[137] C. W. Scherer, "Robust mixed control and linear parameter-varying control with full block scalings," in *Advances in Linear Matrix Inequality Methods in Control.* Society for Industrial and Applied Mathematics, Jan. 2000, pp. 187–207.

[138] P. Apkarian and P. Gahinet, "A convex characterization of gain-scheduled h/sub ∞/ controllers," *IEEE Transactions on Automatic Control*, vol. 40, no. 5, pp. 853–864, May 1995.

[139] A. Packard, "Gain scheduling via linear fractional transformations," *Systems & Control Letters*, vol. 22, no. 2, pp. 79–92, Feb. 1994.

[140] F. Wu, X. H. Yang, A. Packard, and G. Becker, "Induced L2-norm control for LPV systems with bounded parameter variation rates," *International Journal of Robust and Nonlinear Control*, vol. 6, no. 9-10, pp. 983–998, Nov. 1996.

[141] J. C. Willems and K. Takaba, "Dissipativity and stability of interconnections," *International Journal of Robust and Nonlinear Control*, vol. 17, no. 5-6, pp. 563–586, 2007.

[142] J. Carrasco, W. P. Heath, and A. Lanzon, "Factorization of multipliers in passivity and IQC analysis," *Automatica*, vol. 48, no. 5, pp. 909–916, May 2012.

[143] J. Veenman and C. W. Scherer, "Stability analysis with integral quadratic constraints: A dissipativity based proof," in *52nd IEEE Conference on Decision and Control*, Dec 2013, pp. 3770–3775.

[144] J. Carrasco and P. Seiler, "Integral quadratic constraint theorem: A topological separation approach," in *54th IEEE Conference on Decision and Control (CDC)*. IEEE, dec 2015.

[145] P. Seiler, "Stability analysis with dissipation inequalities and integral quadratic constraints," *IEEE Transactions on Automatic Control*, vol. 60, no. 6, pp. 1704–1709, Jun. 2015.

[146] B. A. Francis, *A Course in H∞ Control Theory*. Springer-Verlag, 1987.

[147] A. Rantzer, "On the kalman—yakubovich—popov lemma," *Systems & Control Letters*, vol. 28, no. 1, pp. 7–10, Jun. 1996.

[148] H. Pfifer and P. Seiler, "Less conservative robustness analysis of linear parameter varying systems using integral quadratic constraints," *International Journal of Robust and Nonlinear Control*, vol. 26, no. 16, pp. 3580–3594, Feb. 2016.

[149] P. Gahinet, A. Nemirovskii, A. Laub, and M. Chilali, "The LMI control toolbox," in *Proceedings of 1994 33rd IEEE Conference on Decision and Control*. IEEE, 1994.

[150] H. Pfifer and P. Seiler, "Robustness analysis with parameter-varying integral quadratic constraints," in *2015 American Control Conference (ACC)*, Jul. 2015, pp. 138–143.

[151] R. M. Moore, "Finite horizon robustness analysis using integral quadratic constraints," Master's thesis, University of California, Berkeley, CA, Sep. 2015.

[152] P. Seiler, R. M. Moore, C. Meissen, M. Arcak, and A. Packard, "Finite horizon robustness analysis of LTV systems using integral quadratic constraints," *Automatica*, vol. 100, pp. 135–143, Feb. 2019.

[153] U. Jönsson, "Robustness of trajectories with finite time extent," *Automatica*, vol. 38, no. 9, pp. 1485–1497, Sep. 2002.

[154] I. R. Petersen, V. A. Ugrinovskii, and A. V. Savkin, *Robust Control Design Using H-∞ Methods.* Springer London, 2000.

[155] M. Cantoni and H. Pfifer, "Gap metric computation for time-varying linear systems on finite horizons," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 14 513 – 14 518, Jul. 2017, 20th IFAC World Congress.

[156] S. Vijayshankar and P. Seiler, "Time-varying robustness margins for wind turbines," in *2018 American Control Conference, Wisconsin Center, Milwaukee, USA*, Jun. 2018.

[157] J. Kim, D. G. Bates, and I. Postlethwaite, "Robustness analysis of linear periodic time-varying systems subject to structured uncertainty," *Systems & Control Letters*, vol. 55, no. 9, pp. 719–725, Sep. 2006.

[158] Y. Ebihara, D. Peaucelle, and D. Arzelier, "Analysis of uncertain discrete-time linear periodic systems based on system lifting and LMIs," *European Journal of Control*, vol. 16, no. 5, pp. 532–544, Jan. 2010.

[159] Z. Zhang and A. Serrani, "Adaptive robust output regulation of uncertain linear periodic systems," *IEEE Transactions on Automatic Control*, vol. 54, no. 2, pp. 266–278, Feb. 2009.

[160] K. Sun and G. ming Xie, "Analysis and control of a class of uncertain linear periodic discrete-time systems," *Applied Mathematics and Mechanics*, vol. 30, no. 4, pp. 475–488, Apr. 2009.

[161] D. Ossmann and H. Pfifer, "Robustness analysis of continuous periodic systems using integral quadratic constraints," in *2019 IEEE 58th Conference on Decision and Control (CDC).* IEEE, Dec. 2019.

[162] D. B. Fogel, "Artificial intelligence through simulated evolution," in *Evolutionary Computation: The Fossil Record.* Wiley-IEEE Press, 1998, ch. 7, pp. 227–296.

[163] K. Deb, "An introduction to genetic algorithms," *Sadhana*, vol. 24, no. 4, pp. 293–315, Sep. 1999.

[164] A. Kaveh, "Charged System Search Algorithm," in *Advances in Metaheuristic Algorithms for Optimal Design of Structures*, A. Kaveh, Ed. Cham: Springer International Publishing, Sep. 2017, pp. 45–89.

[165] E. Rashedi, H. Nezamabadi-Pour, and S. Saryazdi, "GSA: a gravitational search algorithm," *Information sciences*, vol. 179, no. 13, pp. 2232–2248, 2009.

[166] S. Mirjalili, “Moth-flame optimization algorithm: A novel nature-inspired heuristic paradigm,” *Knowledge-Based Systems*, vol. 89, pp. 228–249, Jul. 2015.

[167] ——, “Dragonfly algorithm: a new meta-heuristic optimization technique for solving single-objective, discrete, and multi-objective problems,” *Neural Computing and Applications*, vol. 27, no. 4, pp. 1053–1073, Jul. 2016.

[168] L. Wang, Y. Xiong, S. Li, and Y.-R. Zeng, “New fruit fly optimization algorithm with joint search strategies for function optimization problems,” *Knowledge-Based Systems*, vol. 176, pp. 77–96, Feb. 2019.

[169] S. Mirjalili, S. M. Mirjalili, and A. Lewis, “Grey Wolf Optimizer,” *Advances in Engineering Software*, vol. 69, pp. 46–61, Jul. 2014.

[170] S. Mirjalili, “The Ant Lion Optimizer,” *Advances in Engineering Software*, vol. 83, pp. 80–98, Jul. 2015.

[171] S. Bureerat and N. Pholdee, “Optimal truss sizing using an adaptive differential evolution algorithm,” *Journal of Computing in Civil Engineering*, vol. 30, no. 2, p. 04015019, Mar. 2016.

[172] N. Pholdee and S. Bureerat, “Hybrid real-code ant colony optimisation for constrained mechanical design,” *International Journal of Systems Science*, vol. 47, no. 2, pp. 474–491, Feb. 2014.

[173] A. R. Yildiz, H. Abderazek, and S. Mirjalili, “A comparative study of recent non-traditional methods for mechanical design optimization,” *Archives of Computational Methods in Engineering*, May 2019.

[174] B. S. Yıldız and A. R. Yıldız, “Comparison of grey wolf, whale, water cycle, ant lion and sine-cosine algorithms for the optimization of a vehicle engine connecting rod,” *Materials Testing*, vol. 60, no. 3, pp. 311–315, Mar. 2018.

[175] N. Pholdee, S. Bureerat, and A. R. Yıldız, “Hybrid real-code population-based incremental learning and differential evolution for many-objective optimisation of an automotive floor-frame,” *International Journal of Vehicle Design*, vol. 73, no. 1/2/3, p. 20, 2017.

[176] A. R. Yıldız, B. S. Yıldız, S. M. Sait, and X. Li, “The harris hawks, grasshopper and multi-verse optimization algorithms for the selection of optimal machining parameters in manufacturing operations,” *Materials Testing*, vol. 61, no. 8, pp. 725–733, Aug. 2019.

[177] E. Y. Bejarbaneh, A. Bagheri, B. Y. Bejarbaneh, S. Buyamin, and S. N. Chegini, “A new adjusting technique for PID type fuzzy logic controller using PSOSCALF optimization algorithm,” *Applied Soft Computing*, vol. 85, p. 105822, Feb. 2019.

[178] A. ElSaid, F. El Jamiy, J. Higgins, B. Wild, and T. Desell, "Optimizing long short-term memory recurrent neural networks using ant colony optimization to predict turbine engine vibration," *Applied Soft Computing*, vol. 73, pp. 969–991, Feb. 2018.

[179] X. Zhang, X. Lu, S. Jia, and X. Li, "A novel phase angle-encoded fruit fly optimization algorithm with mutation adaptation mechanism applied to UAV path planning," *Applied Soft Computing*, vol. 70, pp. 371–388, Feb. 2018.

[180] C. Qu, W. Gai, M. Zhong, and J. Zhang, "A novel reinforcement learning based grey wolf optimizer algorithm for unmanned aerial vehicles (UAVs) path planning," *Applied Soft Computing*, vol. 89, p. 106099, Feb. 2020.

[181] P. K. Ray and A. Mohanty, "A robust firefly–swarm hybrid optimization for frequency control in wind/PV/FC based microgrid," *Applied Soft Computing*, vol. 85, p. 105823, Feb. 2019.

[182] S.-H. Chen, W.-H. Ho, J.-H. Chou, and L.-A. Zheng, "Design of robust-stable and quadratic finite-horizon optimal active vibration controllers with low trajectory sensitivity for uncertain flexible mechanical systems using an integrative computational method," *Applied Soft Computing*, vol. 11, no. 8, pp. 4830–4838, Feb. 2011.

[183] M. Yang, Y. Li, H. Du, C. Li, and Z. He, "Hierarchical Multiobjective H-Infinity Robust Control Design for Wireless Power Transfer System Using Genetic Algorithm," *IEEE Transactions on Control Systems Technology*, vol. 27, no. 4, pp. 1753–1761, 2019.

[184] Y. Boughari, R. M. Botez, G. Ghazi, and F. Theel, "Flight control clearance of the cessna citation x using evolutionary algorithms," *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, vol. 231, no. 3, pp. 510–532, Aug. 2016.

[185] Y. Boughari, , G. Ghazi, R. M. Boyez, and F. Theel, "New methodology for optimal flight control using differential evolution algorithms applied on the cessna citation x business aircraft – part 2. validation on aircraft research flight level d simulator," *INCAS BULLETIN*, vol. 9, no. 2, pp. 45–59, Jun. 2017.

[186] Q. Bian, B. Nener, and X. Wang, "Control parameter tuning for aircraft crosswind landing via multi-solution particle swarm optimization," *Engineering Optimization*, vol. 50, no. 11, pp. 1914–1925, Apr. 2018.

[187] V. S. Rajput, R. K. Jatoth, P. Dhanuka, and B. Bhasker, "Hybrid DE-PSO based pitch controller design for aircraft control system," in *2016 International Conference on Circuits, Controls, Communications and Computing (I4C)*. IEEE, Oct. 2016.

[188] R. Dou and H. Duan, "Lévy flight based pigeon-inspired optimization for control parameters optimization in automatic carrier landing system," *Aerospace Science and Technology*, vol. 61, pp. 11–20, Feb. 2017.

[189] H. J. Wright, R. Strydom, and M. V. Srinivasan, “A generalized algorithm for tuning UAS flight controllers,” in *2018 International Conference on Unmanned Aircraft Systems (ICUAS)*. IEEE, Jun. 2018.

[190] E. S. Ayra, D. R. Insua, and J. Cano, “Bayesian network for managing runway overruns in aviation safety,” *Journal of Aerospace Information Systems*, vol. 16, no. 12, pp. 546–558, Dec. 2019.

[191] A. L. H. M. T. N. H. Murphy, Tim, “Cat iii simulated landing performance for gls and ils systems,” in *Proceedings of the 14th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GPS 2001)*, Salt Lake City, UT, Sep. 2001, pp. pp. 1679–1688.

[192] P. P. Menon, J. Kim, D. G. Bates, and I. Postlethwaite, “Clearance of nonlinear flight control laws using hybrid evolutionary optimization,” *IEEE Transactions on Evolutionary Computation*, vol. 10, no. 6, pp. 689–699, 2006.

[193] H.-D. Joos and H. Pfifer, “Robust flight control system design verification and validation by multi-objective worst-case search,” in *AIAA Guidance, Navigation, and Control Conference*. American Institute of Aeronautics and Astronautics, Aug. 2012.

[194] H.-D. Joos, “Worst-case parameter search based clearance using parallel nonlinear programming methods,” in *Lecture Notes in Control and Information Sciences*. Springer Berlin Heidelberg, 2012, pp. 149–159.

[195] A. Iannelli, P. Simplício, D. Navarro-Tapia, and A. Marcos, “LFT modeling and μ analysis of the aircraft landing benchmark,” *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 3965–3970, Jul. 2017.

[196] M. Althoff, B. H. Krogh, and O. Stursberg, “Analyzing reachability of linear dynamic systems with parametric uncertainties,” in *Modeling, Design, and Simulation of Systems with Uncertainties*. Springer Berlin Heidelberg, 2011, pp. 69–94.

[197] A. Girard, C. L. Guernic, and O. Maler, “Efficient computation of reachable sets of linear time-invariant systems with inputs,” in *Hybrid Systems: Computation and Control*. Springer Berlin Heidelberg, 2006, pp. 257–271.

[198] I. Ben Makhlouf, P. Hänsch, and S. Kowalewski, “Comparison of Reachability Methods for Uncertain Linear Time-Invariant Systems,” in *Proceedings of the 12th European Control Conference (ECC) : July 17 -19, 2013, Zuerich, Switzerland*. Zürich: Omnipress, 2013, pp. 1101–1106.

[199] M. Althoff, C. L. Guernic, and B. H. Krogh, “Reachable set computation for uncertain time-varying linear systems,” in *Proceedings of the 14th international conference on Hybrid systems: computation and control - HSCC '11*. ACM Press, 2011.

[200] H. Pfifer and P. Seiler, “Less conservative robustness analysis of linear parameter varying systems using integral quadratic constraints,” *International Journal of Robust and Nonlinear Control*, vol. 26, no. 16, pp. 3580–3594, Feb. 2016.

[201] H. Yin, P. Seiler, and M. Arcak, “Backward reachability using integral quadratic constraints for uncertain nonlinear systems,” *IEEE Control Systems Letters*, vol. 5, no. 2, pp. 707–712, Apr. 2021.

[202] A. Marcos, V. Mantini, C. Roux, and S. Bennani, “Bridging the gap between linear and nonlinear worst-case analysis: an application case to the atmospheric phase of the VEGA launcher,” *IFAC Proceedings Volumes*, vol. 46, no. 19, pp. 42–47, 2013.

[203] A. Marcos, S. Bennani, C. Roux, and M. Valli, “Uncertainty modeling and robust analysis of atmospheric launchers: Incremental steps for industrial transfer,” *IFAC-PapersOnLine*, vol. 48, no. 14, pp. 426–431, 2015.

[204] P. Simplicio, S. Bennani, X. Lefort, A. .Marcos, and C. Roux, “Structured singular value analysis of the vega launcher in atmospheric flight,” *Journal of Guidance, Control, and Dynamics*, vol. 39, no. 6, pp. 1342–1355, 2016.

[205] M. Ganet-Schoeller, G. Maurice, and S. Bennani, “SAFE-v launcher validation framework and controller optimization,” *IFAC Proceedings Volumes*, vol. 46, no. 19, pp. 482–487, 2013.

[206] J. Pei and J. Newsom, “Robust stability evaluation of the space launch system control design: A singular value approach,” in *AIAA Atmospheric Flight Mechanics Conference.* American Institute of Aeronautics and Astronautics, Jan. 2015.

[207] W. L. Brogan, *Modern Control Theory (3rd Ed.).* Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 1991.

[208] D. Navarro-Tapia, A. Marcos, S. Bennani, and C. Roux, “Linear parameter varying control synthesis for the atmospheric phase VEGA launcher,” *IFAC-PapersOnLine*, vol. 51, no. 26, pp. 68–73, 2018.

[209] A. L. Greensite, “Analysis and design of space vehicle flight control systems. volume vii - attitude control during launch,” NASA Marshall Space Flight Center; Huntsville, AL, United States, techreport, 1967.

[210] J. Orr, M. Johnson, J. Wetherbee, and J. McDuffie, “State space implementation of linear perturbation dynamics equations for flexible launch vehicles,” in *AIAA Guidance, Navigation, and Control Conference.* American Institute of Aeronautics and Astronautics, Aug. 2009.

[211] A. Miele, *Flight Mechanics Theory of Flight Paths.* Addison-Wesley Publishing Company Inc., 1962.

[212] A. L. Greensite, "Analysis and design of space vehicle flight control systems. volume vii - attitude control during launch," NASA Marshall Space Flight Center; Huntsville, AL, United States, techreport, 1967.

[213] M. A. Rotea, "The generalized h2 control problem," *Automatica*, vol. 29, no. 2, pp. 373–385, Mar. 1993.

[214] *MATLAB version 9.3.0.713579 (R2017b)*, The Mathworks, Inc., Natick, Massachusetts, 2017.

[215] N. M. Barr, "Wind models for flight simulator certification of landing and approach guaidance and control systems," Boeing Commercial Aircraft Company, Tech. Rep., Dec. 1974.

[216] F. M. Hoblit, *Gust Loads on Aircraft: Concepts and Applications.* American Institute of Aeronautics and Astronautics, Jan. 1988.

[217] N. Aeronautics and S. Administration, "Terrestrial environment (climatic) criteria handbook for use in aerospace vehicle development," Marshall Space Flight Center, techreport, Aug. 2000, nASA-HDBK-1001.

[218] "Flying qualities of piloted airplanes u.s. military specification mil-f-8785c," U.S. Department of Defense, techreport, 1980.

[219] A. Knoblach and G. Looye, "Efficient determination of worst-case gust loads using system norms," *Journal of Aircraft*, vol. 54, no. 3, pp. 1205–1210, May 2017.

[220] F. Biertümpfel and H. Pfifer, "Finite time horizon worst case analysis of launch vehicles," in *21st IFAC Symposium on Automatic Control in Aerospace*, 2019.

[221] G. Boothroyd, "Use of robots in assembly automation," *CIRP Annals*, vol. 33, no. 2, pp. 475–484, Feb. 1984.

[222] R. Hessert, W. Satzger, A. Haase, and A. Schafmeister, "A new type of x-ray diffractometer with cooperating robots for residual stress analysis on large components," *Materials Science Forum*, vol. 524-525, pp. 749–754, Sep. 2006.

[223] A. Klimchik, A. Ambiehl, S. Garnier, B. Furet, and A. Pashkevich, "Efficiency evaluation of robots in machining applications using industrial performance measure," *Robotics and Computer-Integrated Manufacturing*, vol. 48, pp. 12–29, Dec. 2017.

[224] A. E. K. Mohammad, J. Hong, and D. Wang, "Design of a force-controlled end-effector with low-inertia effect for robotic polishing using macro-mini robot approach," *Robotics and Computer-Integrated Manufacturing*, vol. 49, pp. 54–65, Feb. 2018.

[225] F. Zhang, Ed., *The Schur Complement and Its Applications.* Springer-Verlag, 2005.

[226] A. Megretski and A. Rantzer, "System analysis via integral quadratic constraints," *IEEE Transactions on Automatic Control*, vol. 42, no. 6, pp. 819–830, Jun 1997.

[227] J. Veenman, C. W. Scherer, and H. Köroğlu, "Robust stability and performance analysis based on integral quadratic constraints," *European Journal of Control*, vol. 31, pp. 1–32, Sep. 2016.

[228] H. Abou-Kandil, G. Freiling, V. Ionescu, and G. Jank, *Matrix Riccati Equations in Control and Systems Theory.* Birkhäuser Basel, 2003.

[229] G. Freiling, G. Jank, and A. Sarychev, "Non — blow — up conditions for riccati — type matrix differential and difference equations," *Results in Mathematics*, vol. 37, no. 1-2, pp. 84–103, Mar. 2000.

[230] H. Pfifer and P. Seiler, "Integral quadratic constraints for delayed nonlinear and parameter-varying systems," *Automatica*, vol. 56, pp. 36 – 43, Jun. 2015.

[231] ——, "Robustness analysis of linear parameter varying systems using integral quadratic constraints," *International Journal of Robust and Nonlinear Control*, vol. 25, no. 15, pp. 2843–2864, Aug. 2014.

[232] S. Mirjalili, "SCA: A Sine Cosine Algorithm for solving optimization problems," *Knowledge-Based Systems*, vol. 96, pp. 120–133, Jul. 2016.

[233] M. D. McKay, R. J. Beckman, and W. J. Conover, "A comparison of three methods for selecting values of input variables in the analysis of output from a computer code," *Technometrics*, vol. 21, no. 2, p. 239, May 1979.

[234] D. Arthur and S. Vassilvitskii, "k-means++: The advantages of careful seeding," Stanford InfoLab, techreport 2006-13, 2006.

[235] J. MacQueen, "Some methods for classification and analysis of multivariate observations," in *In 5-th Berkeley Symposium on Mathematical Statistics and Probability*, 1967, pp. 281–297.

[236] R. Tanabe and A. S. Fukunaga, "Improving the search performance of SHADE using linear population size reduction," in *2014 IEEE Congress on Evolutionary Computation (CEC).* IEEE, Jul. 2014.

[237] D. Lehmer, "On the compounding of certain means," *Journal of Mathematical Analysis and Applications*, vol. 36, no. 1, pp. 183–200, Oct. 1971.

[238] F. Biertümpfel and H. Pfifer, "Worst Case Gain Computation of Linear Time-Varying Systems over a Finite Horizon," in *2018 IEEE Conference on Control Technology and Applications (CCTA)*, 2018, pp. 952–957.

[239] NIMA, "Department of defense world geodetic system 1984, its definition and relationships with local geodetic systems," Department of Defense, Tech. Rep. 3, Jul. 1997.

[240] J. D. BLIGHT, R. L. DAILEY, and D. GANGSAAS, "Practical control law design for aircraft using multivariable techniques," *International Journal of Control*, vol. 59, no. 1, pp. 93–137, jan 1994.

[241] J. Zhang and A. C. Sanderson, "JADE: adaptive differential evolution with optional external archive," *IEEE Transactions on evolutionary computation*, vol. 13, no. 5, pp. 945–958, 2009.

[242] R. Tanabe and A. Fukunaga, "Evaluating the performance of SHADE on CEC 2013 benchmark problems." IEEE, 2013, pp. 1952–1959.

[243] K. M. Sallam, R. A. Sarker, D. L. Essam, and S. M. Elsayed, "Neurodynamic differential evolution algorithm and solving CEC2015 competition problems," in *2015 IEEE Congress on Evolutionary Computation (CEC)*, 2015, pp. 1033–1040.

[244] S.-M. Guo, J. S.-H. Tsai, C.-C. Yang, and P.-H. Hsu, "A self-optimization approach for L-SHADE incorporated with eigenvector-based crossover and successful-parent-selecting framework on CEC 2015 benchmark set." IEEE, 2015, pp. 1003–1010.

[245] S. Mirjalili and A. Lewis, "The Whale Optimization Algorithm," *Advances in Engineering Software*, vol. 95, pp. 51–67, Jul. 2016.

[246] S. Gupta and K. Deep, "Improved sine cosine algorithm with crossover scheme for global optimization," *Knowledge-Based Systems*, vol. 165, pp. 374–406, Jul. 2019.

[247] ——, "A hybrid self-adaptive sine cosine algorithm with opposition based learning," *Expert Systems with Applications*, vol. 119, pp. 210–230, Jul. 2019.

[248] D. T. McRuer, D. Graham, and I. Ashkenas, *Aircraft Dynamics and Automatic Control.* Princeton University Press, 1974.

[249] H. Hindi, C.-Y. Seong, and S. Boyd, "Computing optimal uncertainty models from frequency domain data," in *Proceedings of the 41st IEEE Conference on Decision and Control, 2002.* IEEE, 2002.

[250] W. Haeussermann, "Description and performance of the saturn launch vehicles navigation, guidance, and control system," *IFAC Proceedings Volumes*, vol. 3, no. 1, pp. 275–312, Mar. 1970.

[251] D. F. Enns, "Structured singular value synthesis design example: Rocket stabilization," in *1990 American Control Conference*, May 1990, pp. 2514–2520.

[252] A. L. Greensite, "Analysis and design of space vehicle flight control systems. volume i - short period dynamics," NASA Marshall Space Flight Center; Huntsville, AL, United States, techreport, 1967.

[253] J. A. Blevins, J. R. Campbell, D. W. Bennett, R. Rausch, R. J. Gomez, and C. C. Kiris, "An overview of the characterization of the space launch system aerodynamic environments," in *52nd Aerospace Sciences Meeting.* American Institute of Aeronautics and Astronautics, Jan. 2014.

[254] M. Applebaum, W. Eppard, L. Hall, and J. Blevins, "Protuberance aerodynamic loads for space launch vehicle systems using CFD," in *29th AIAA Applied Aerodynamics Conference.* American Institute of Aeronautics and Astronautics, Jun. 2011.

[255] D. L. Johnson, "Terrestrial environment (climatic) criteria guidelines for use in aerospace vehicle development. 2008 revision," NASA Marshall Space Flight Center; Huntsville, AL, United States, techreport, 2008.

[256] B. Wie, *Space Vehicle Dynamics and Control, Second Edition.* American Institute of Aeronautics and Astronautics, Jan. 2008.

[257] "Manual of the icao standard artmosphere," International Civil Aviation Organization and Langley Aerodynamical Labratory, Tech. Rep., May 1954.

[258] C. Roux, V. Mantini, A. Marcos, L. Peñín, and S. Bennani, "Robust flight control system design verification & validation for launchers," in *AIAA Guidance, Navigation, and Control Conference.* American Institute of Aeronautics and Astronautics, Aug. 2012.

[259] I. Rongier and G. Droz, "Robustness of ariane 5 gnc algorithms," in *Proceedings of the 4th ESA International Conference on Spacecraft Guidance, Navigation and Control Systems*, ESTEC Noordwijk, Netherlands, 1999.

[260] "Terrestrial environment criteria guidlines for use in aerospace vehicle development 1982 revision," National Aeronautics and Space Administration, techreport, 1982.

Appendix A

Simple Launcher Model

Here, the launcher dynamics used for the introductory example in Chapter 2 are derived. It starts with the formulation of the nonlinear equations of motion.Based on these, linear state-space models are derived via analytical linearization.

A.1 Nonlinear Launcher Dynamics

The forces acting on the launcher are pictured in the schematic diagram given in Fig. A.1. A launcher fixed coordinate system denoted by the subscript b is used to formulate the launcher's nonlinear equations of motion. Its origin is fixed to the launcher's center of gravity G. The x_b-axis is aligned with the launcher's symmetry axis and is defined as positive pointing forward. Its z_b-axis points downward, building a right-hand system with the y_b-axis pointing out of the page. Corresponding to this convention, the rigid body motion in

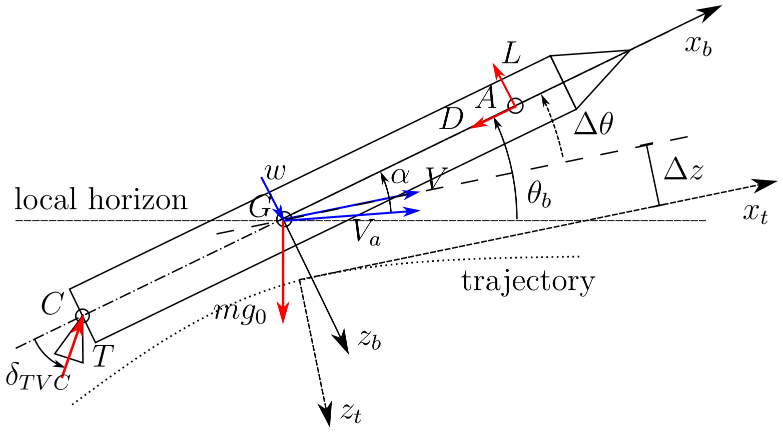


Figure A.1: *Launcher vehicle and trajectory frame dynamics*

the pitch plane is given by:

$$\begin{aligned}
\ddot{\theta}_b(t) &= \frac{L(t)l_{GA}(t)}{J_y(t)} - \frac{T(t)l_{CG}(t)}{J_y(t)} - \dot{\theta}_b\frac{\dot{J}_y}{J_y} \\
\ddot{x}_b(t) &= \frac{T(t)\cos(\delta_{TVC(t)}) - D(t)}{m(t)} - g_0(t)\sin\theta_b(t) - \dot{\theta}_b(t)\dot{z}_b(t) \\
\ddot{z}_b(t) &= -\frac{L}{m} - \frac{T(t)}{m(t)}\sin\delta_{TVC}(t) + g_0(t)\cos\theta_b(t) + \dot{\theta}_b(t)\dot{x}_b(t),
\end{aligned} \tag{A.1}$$

where θ_b is the pitch angle of the launcher describing the angle between the body axis and the local horizon. The forward and downward accelerations are denoted by $\ddot{x}_b$ and $\ddot{z}_b$, respectively. V_a is the air relative velocity of the launcher, whereas V is the kinematic velocity. L denotes the aerodynamic lift. It is defined as positive in upward direction parallel to the z_b-axis as

$$L = \underbrace{QSC_{L_\alpha}}_{L_\alpha}\alpha, \tag{A.2}$$

where S is the reference area, C_{L_α} the (linear) lift coefficient and Q the dynamic pressure. The latter is defined as

$$Q = \frac{1}{2}\rho V_a^2 \tag{A.3}$$

with ρ being the altitude dependent air density calculated according to the international standard atmosphere (ISA) [257]. The aerodynamic drag D is defined in the same way with respect to the x_b-axis and defined as

$$D = \underbrace{QSC_{D_0}}_{D_0} + \underbrace{QSC_{D_\alpha}}_{D_\alpha}\alpha, \tag{A.4}$$

where C_{D_0} is the zero-lift drag coefficient and C_{D_α} the induced drag coefficient. L as well as D act at the aerodynamic center A. By m, the total mass of the ELV is denoted. J_y denotes the overall mass moment of inertia with respect to G. The thrust is denoted as T. It acts at the nozzle reference point C. The deflection of the thrust vector by the TVC is denoted as δ_{TVC}. The geometric variables l_{CG} and l_{GA} are defined as the absolute distances between C and G, and G and A, respectively. The gravitational acceleration g_0 is modeled according to the world geodetic system 1984 (WGS84), see [239], assuming a launch at the equator. The angle of attack α is approximated as

$$\alpha \approx \frac{\dot{z}_b - w}{V}, \tag{A.5}$$

where w denotes the wind velocity in z_b direction.

Linear Dynamics

The launcher is linearized along a so-called gravity turn trajectory for $\delta_{\mathrm{TVC}} = \alpha \approx 0$ with respect to a non-stationary trajectory fixed frame. A gravity turn trajectory is based on tracking a pre-calculated time scheduled $\theta_{\mathrm{ref}}(t)$ profile resulting in a pitch rate of approximately:

$$\dot{\theta}_b(t) = \frac{g_0(t)\sin\theta_{\text{ref}}(t)}{V(t)} \tag{A.6}$$

Accordingly, the centrifugal and gravitational term in (A.1) are offset. The trajectory fixed frame is denoted by the subscript t in Fig. A.1. It translates axially with the launcher but remains tangent to the ascent trajectory. It is illustrated in Fig. A.2 for three points in time along the trajectory. The transformation matrix from the actual body reference frame to the reference trajectory frame is introduced as

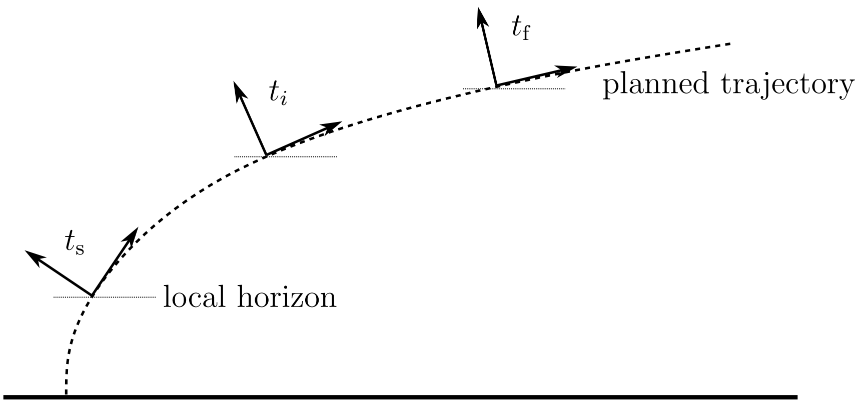


Figure A.2: *Trajectory frame along ascent trajectory [210]*

$$\mathcal{T}_{tb} = \begin{bmatrix} \cos\Delta\theta & \sin\Delta\theta \\ -\sin\Delta\theta & \cos\Delta\theta \end{bmatrix}. \tag{A.7}$$

Accordingly, the perturbed equations of motion in the reference frame are:

$$\begin{aligned} \begin{bmatrix} \Delta\ddot{x} \\ \Delta\ddot{z} \end{bmatrix} &= \mathcal{T}_{tb}\begin{bmatrix} \ddot{x}_b \\ \ddot{z}_b \end{bmatrix} - \begin{bmatrix} \ddot{x}_t \\ \ddot{z}_t \end{bmatrix} \\ \Delta\ddot{x} &= \ddot{x}_b\cos(\Delta\theta) + \ddot{z}_b\sin(\Delta\theta) - \ddot{x}_t \\ \Delta\ddot{z} &= -\ddot{x}_b\sin(\Delta\theta) + \ddot{z}_b\cos(\Delta\theta) - \ddot{z}_t. \end{aligned} \tag{A.8}$$

Here, the subscript b represents actual coordinates, while t represents the ones of the reference trajectory. The states in the body frame can be formulated in terms of the reference variables as:

$$\begin{aligned} \theta_b &= \theta_t + \Delta\theta \\ \dot{\theta}_b &= \Delta\dot{\theta} \\ \dot{z}_b &= \dot{z}_t + \Delta\dot{z} \\ \alpha_b &= \alpha_t + \Delta\alpha \\ \delta_{\text{TVC},b} &= \delta_{\text{TVC},t} + \Delta\delta_{\text{TVC}}. \end{aligned} \tag{A.9}$$

Accordingly, the equations in (A.1) expressed for small deviations from the reference can be written as:

$$
\begin{aligned}
\ddot{\theta}_t + \Delta\ddot{\theta} =& a_6(\alpha_t + \Delta\alpha) - k_1(\delta_{\mathrm{TVC},t} + \Delta\delta_{\mathrm{TVC}}) - \frac{\dot{J}_y}{J_y}(\dot{\theta}_R + \Delta\dot{\theta}) \\
\ddot{x}_b =& \frac{T-D}{m} - g\sin(\theta_t + \Delta\theta) - (\dot{\theta}_t + \Delta\dot{\theta})(\dot{z}_t + \Delta\dot{z}) \\
\ddot{z}_b =& -\frac{L_\alpha}{m}(\alpha_t + \Delta\alpha) - \frac{T}{m}(\delta_t + \Delta\delta) + g\cos(\theta_t + \Delta\theta) \\
&+ (\dot{\theta}_t + \Delta\dot{\theta})(\dot{x}_t + \Delta\dot{x}),
\end{aligned}
\tag{A.10}
$$

where $k_1 = \frac{Tl_{CG}}{J_y}$. Assuming small angles $\Delta\theta$

$$
\begin{aligned}
\cos(\Delta\theta) &\approx 1 \\
\sin(\Delta\theta) &\approx \Delta\theta
\end{aligned}
\tag{A.11}
$$

and applying the addition theorem for sine and cosine, the theta equation of (A.10) can be simplified to:

$$
\begin{aligned}
\Delta\ddot{\theta} =& a_6(\alpha_t + \Delta\alpha) - k_1(\delta_{\mathrm{TVC},t} + \Delta\delta_{\mathrm{TVC}}) - \frac{\dot{J}_y}{J_y}(\dot{\theta}_t + \Delta\dot{\theta}) - \ddot{\theta}_R \\
\Delta\ddot{\theta} =& a_6\Delta\alpha - k_1\Delta\delta_{\mathrm{TVC}} \\
\Delta\ddot{x} =& \frac{T-D}{m} - g\cos(\theta_t)\Delta\theta - g\sin(\theta_t) - \dot{\theta}_t\dot{z}_t - \dot{\theta}_t\Delta\dot{z} - \Delta\dot{\theta}\dot{z}_t \\
&- \Delta\dot{\theta}\Delta\dot{z} - \frac{L_\alpha}{m}(\alpha_t + \Delta\alpha)\Delta\theta - \frac{T}{m}(\delta_t + \Delta\delta)\Delta\theta + g\cos(\theta_t)\Delta\theta \\
&- g\sin(\theta_t)(\Delta\theta)^2 + \dot{\theta}_t\dot{x}_t\Delta\theta + \dot{\theta}_t\Delta\dot{x}\Delta\theta + \Delta\dot{\theta}\dot{x}_t\Delta\theta + \Delta\dot{\theta}\Delta\dot{x}\Delta\theta - \ddot{x}_t \\
\Delta\ddot{z} =& -\frac{T-D}{m}\Delta\theta + g\cos(\theta_t)(\Delta\theta)^2 + g\sin(\theta_t)\Delta\theta + \dot{\theta}_t\dot{z}_t\Delta\theta + \dot{\theta}_t\Delta\dot{z}\Delta\theta \\
&+ \dot{z}_t(\Delta\dot{\theta})^2 + \Delta\dot{z}(\Delta\dot{\theta})^2 - \frac{L_\alpha}{m}(\alpha_t + \Delta\alpha) - \frac{T}{m}(\delta_{\mathrm{TVC},t} + \Delta\delta_{\mathrm{TVC}}) + g\cos(\theta_t) \\
&- g\sin(\theta_t)\Delta\theta + \dot{\theta}_t\dot{x}_t + \dot{\theta}_t\Delta\dot{x} + \Delta\dot{\theta}\dot{x}_t + \Delta\dot{\theta}\Delta\dot{x} - \ddot{z}_t.
\end{aligned}
\tag{A.12}
$$

Afterwards, all higher-order terms, as well as products of variables of small sizes, are neglected in (A.12) leading to the linear equations with respect to an arbitrary reference trajectory:

$$
\begin{aligned}
\Delta\ddot{\theta} &= a_6\Delta\alpha - k_1\Delta\delta_{\mathrm{TVC}} - \frac{\dot{J}_y}{J_y}\Delta\dot{\theta} \\
\Delta\ddot{x} &= -(\frac{L_\alpha}{m}\alpha_t + \frac{T}{m}\delta_{\mathrm{TVC},t})\Delta\theta \\
\Delta\ddot{z} &= \frac{T-D}{m}\Delta\theta - \frac{L_\alpha}{m}\Delta\alpha - \frac{T}{m}\Delta\delta_{\mathrm{TVC}}.
\end{aligned}
\tag{A.13}
$$

For the gravity turn maneuver the equation can be further simplified, as $\alpha_t \approx 0$ and $\delta_{\mathrm{TVC},t} \approx 0$. Accordingly, the simplified linearized pitch equations of motion for the analyzed launcher are

$$\begin{aligned} \Delta\ddot{\theta} &= a_6\Delta\alpha - k_1\Delta\delta - \frac{\dot{J}_y}{J_y}\Delta\dot{\theta} \\ \Delta\ddot{z} &= \frac{T-D}{m}\Delta\theta - \frac{L_\alpha}{m}\Delta\alpha - \frac{T}{m}\Delta\delta_{\text{TVC}}, \end{aligned} \tag{A.14}$$

where the $\Delta\ddot{x}$ equation is neglectable in total. The angle of attack $\Delta\alpha$ is defined as

$$\Delta\alpha = \Delta\theta + \frac{\Delta\dot{z} - \Delta w}{V_a}. \tag{A.15}$$

In state space form the launchers linear representation can be written as G_{LV}:

$$\begin{gathered} \begin{bmatrix} \Delta\dot{\theta} \\ \Delta\ddot{\theta} \\ \Delta\dot{z} \\ \Delta\ddot{z} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ a_6 & -a_4 & 0 & \frac{a_6}{V} \\ 0 & 0 & 0 & 1 \\ -a_1 & 0 & 0 & -a_2 \end{bmatrix} \begin{bmatrix} \Delta\theta \\ \Delta\dot{\theta} \\ \Delta z \\ \Delta\dot{z} \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ -\frac{a_6}{V} & -k_1 \\ a_2 & -a_3 \end{bmatrix} \begin{bmatrix} \Delta w \\ \Delta\delta_{\text{TVC}} \end{bmatrix} \\ \begin{bmatrix} \Delta\alpha \\ \Delta\theta \\ \Delta\dot{\theta} \\ \Delta z \\ \Delta\dot{z} \\ \Delta Q\alpha \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & \frac{1}{V} \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ Q & 0 & 0 & \frac{Q}{V} \end{bmatrix} \begin{bmatrix} \Delta\theta \\ \Delta\dot{\theta} \\ \Delta z \\ \Delta\dot{z} \end{bmatrix} + \begin{bmatrix} -\frac{1}{V} & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ -\frac{Q}{V} & 0 \end{bmatrix} \begin{bmatrix} \Delta w \\ \Delta\delta \end{bmatrix} \end{gathered} \tag{A.16}$$

, with

$$\begin{aligned} a_1(t) = \frac{L_\alpha(t) + T(t) - D_0(t)}{m_0(t)}, \quad a_2(t) = \frac{L_\alpha(t)}{m(t)V(t)}, \quad a_3(t) = \frac{T(t)}{m(t)} \\ a_4(t) = \frac{\dot{J}_y(t)}{J_y(t)}, \quad \textit{and} \quad a_6(t) = \frac{L_\alpha(t)l_{GA}(t)}{J_y(t)}. \end{aligned} \tag{A.17}$$

In (A.16), the states are $\Delta\theta$, $\Delta\dot{\theta}$, Δz, and $\Delta\dot{z}$ d. The input signals are the $\Delta\delta_{\text{TVC}}$ and Δw, and the output signals $\Delta\theta$, $\Delta\dot{\theta}$, Δz, $\Delta\dot{z}$, and the aerodynamic load $\Delta Q\alpha$. $Q\alpha$ is simply the product of the dynamic pressure Q and the angle of attack α. All matrix coefficients in (A.16) are time-dependent as a result of the predefined trajectory. The explicit time-dependence of the system matrices is only omitted to shorten the notation.

Appendix B

Additional Results Ab-SCA-PR Benchmark

The following tables contain the results for the remaining analyzed norm bounds:

Table B.1: Additional optimization results for fixed bisection bounds (n.f. = not feasible, and N/A = not available, S = SHADE, LS = L-SHADE, LSND = L-SHADE-ND, SPS = SPS-L-SHADE-EIG, New = Ab-SCA-PR)

b		DE	JADE	S	LS	LSND	SPS	SCA	DA	GWO	MFO	MVO	WOA	ISCA	mSCA	New
0.06	γ_{best}	14.521	n.f	n.f.	n.f.	n.f.	n.f.	4.0151	4.0145	4.0066	n.f.	n.f.	4.0070	n.f.	n.f.	4.0071
	μ_γ	N/A	N/A	N/A	N/A	N/A	N/A	4.0151	N/A	6.9674	N/A	N/A	N/A	N/A	N/A	4.0105
	σ_γ	N/A	N/A	N/A	N/A	N/A	N/A	0.0	N/A	4.1872	N/A	N/A	N/A	N/A	N/A	0.0032
	n_{feas}	1	0	0	0	0	0	2	1	2	0	0	1	0	0	5
0.075	γ_{best}	n.f.	n.f	n.f.	n.f.	n.f.	n.f.	5.0705	n.f	5.0511	n.f.	n.f.	5.0624	n.f.	14.359	5.0733
	μ_γ	N/A	N/A	N/A	N/A	N/A	N/A	5.0719	N/A	13.143	N/A	N/A	5.0680	N/A	N/A	5.0735
	σ_γ	N/A	N/A	N/A	N/A	N/A	N/A	0.0020	N/A	7.1196	N/A	N/A	0.0049	N/A	N/A	0.0001
	n_{feas}	0	0	0	0	0	0	2	0	3	0	0	3	0	1	5
0.08	γ_{best}	n.f.	n.f	n.f.	n.f.	n.f.	n.f.	6.3345	n.f.	n.f.	6.3459	n.f.	6.4081	6.8475	6.3195	6.2975
	μ_γ	N/A	N/A	N/A	N/A	N/A	N/A	7.1484	N/A	N/A	N/A	N/A	N/A	N/A	7.6963	6.3193
	σ_γ	N/A	N/A	N/A	N/A	N/A	N/A	1.4097	N/A	N/A	N/A	N/A	N/A	N/A	1.9471	0.0136
	n_{feas}	0	0	0	0	0	0	4	0	0	1	0	1	1	2	5

Table B.2: Additional optimization results for adaptive bisection bounds (n.f. = not feasible, and N/A = not available, S = SHADE, LS = L-SHADE, LSND = L-SHADE-ND, SPS = SPS-L-SHADE-EIG, New = Ab-SCA-PR)

b		DE	JADE	S	LS	LSND	SPS	SCA	DA	GWO	MFO	MVO	WOA	ISCA	mSCA	New
0.06	γ_{best}	4.0143	n.f	n.f.	n.f.	n.f.	n.f.	n.f.	n.f.	n.f.	n.f.	n.f.	4.0129	4.7757	n.f.	4.0127
	μ_γ	4.0143	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	4.0129	N/A	N/A	4.0131
	σ_γ	0.0	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	0.0	N/A	N/A	0.0005
	n_{feas}	3	0	0	0	0	0	0	0	0	0	0	3	1	0	5
0.075	γ_{best}	5.2537	n.f	n.f.	n.f.	n.f.	n.f.	n.f.	n.f	5.0696	n.f.	n.f.	n.f.	n.f.	11.108	5.0733
	μ_γ	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	5.0741	N/A	N/A	N/A	N/A	12.721	5.0735
	σ_γ	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	0.0091	N/A	N/A	N/A	N/A	0.9312	0.0001
	n_{feas}	1	0	0	0	0	0	0	0	4	0	0	0	0	3	5
0.08	γ_{best}	n.f.	n.f	n.f.	n.f.	n.f.	n.f.	n.f.	n.f.	6.3167	13.442	n.f.	n.f.	6.4161	n.f.	6.3140
	μ_γ	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	6.3167	N/A	N/A	N/A	N/A	N/A	6.3238
	σ_γ	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	0.0	N/A	N/A	N/A	N/A	N/A	0.0139
	n_{feas}	0	0	0	0	0	0	0	0	3	1	0	0	1	0	5

Appendix C

Linear Aircraft Model States, Inputs, and Outputs

The following tables contain a summary of the linearized aircraft dynamic's states, inputs, and outputs.

Table C.1: Input, outputs and states of the lateral LTV model

	Variable	Name	Unit
States	v	lateral velocity	m/s
	p	roll rate	rad/s
	r	yaw rate	rad/s
	y	lateral position	m
	ϕ	roll angle	rad
	ψ	heading angle	rad
	x_{a}	aileron actuator	rad
	x_{r}	rudder actuator	rad
Inputs	δ_{a}	aileron deflection	rad
	δ_{r}	rudder deflection	rad
	v_w	lateral wind	m/s
Outputs	n_y	lateral load factor	$\mathrm{m/s}^2$
	p	roll rate	rad/s
	r	yaw rate	rad/s
	ϕ	bank angle	rad
	ψ	heading angle	rad
	V_{g}	ground speed	m/s
	χ	course angle	rad
	ΔY	localizer deviation	m
	y_{LG}	landing gear lateral offset w.r.t center line	m
	β_{LG}	sideslip angle landing gear	rad

Table C.2: Input, outputs and states of the longitudinal LTV model

	Variable	Name	Unit
States	u	longitudinal velocity	m/s
	w	vertical velocity	m/s
	q	pitch rate	rad/s
	x	longitudinal position	m
	z	vertical position	m
	θ	pitch angle	rad
	x_{th}	engine state	[-]
	x_{e}	elevator actuator state	rad
Inputs	δ_{e}	elevator deflection	rad
	δ_{th}	throttle position	-
	u_w	longitudinal wind	m/s
	w_w	vertical wind	m/s
Outputs	n_x	longitudinal load factor	m/s
	n_z	vertical load factor	m/s
	q	pitch rate	rad/s
	θ	pitch angle	rad
	α	angle of attack	rad
	V_{CAS}	calibrated airspeed	m/s
	V_{TAS}	true airspeed	m/s
	V_g	ground speed	m/s
	V_z	Inertial vertical airspeed	m/s
	H	altitude	m
	H_{AGL}	landing gear height above ground level	m
	ΔZ	glideslope deviation	m
	$V_{z_{\text{LG}}}$	landing gear vertical speed	m/s
	D_{LG}	distance to threshold	m