

Optimal Control for Safety-Critical Systems



Nikolaus Ceri Vertovec
Keble College
University of Oxford

A thesis submitted for the degree of
Doctor of Philosophy
Hillary 2024

This thesis is dedicated to
my parents, Astrid Gräfe-Vertovec and Steven Vertovec, and my partner, Kori
Malia, for their incredible support throughout this journey.

Acknowledgements

Personal

I'm delighted to have reached this important stage in my career and would like to acknowledge and thank everyone who has contributed to this incredible, sometimes arduous, but pleasant and rewarding journey. I would like to use this space to thank those who have contributed and supported me throughout this journey.

I would like to begin by expressing my gratitude to my supervisors Prof Kostas Margellos and Prof Sina Ober-Blöbaum. Prof Ober-Blöbaum, thank you for exposing me to such a vast area of research topics and welcoming me into your research group in Paderborn. Prof Margellos, thank you for being so supportive, especially during the Covid-19 Pandemic. The amount of time you dedicated to working out proofs together, giving detailed feedback on papers and providing guidance for my career has allowed me to grow greatly as a young academic. I am deeply grateful to have been able to work with both of you.

I would like to thank both Prof. Maryam Kamgarpour and Prof. Mark Cannon for serving as examiners in my DPhil viva and for providing me with constructive feedback which I believe has greatly improved the final version of this document.

The Control Group at the University of Oxford is unique in its dedication to teaching and mentoring. The seminars and journal clubs have gotten me excited about numerous research projects I had previously not encountered. Thank you to all the academics of the group for organising these. In particular, I would also like to thank Prof. Konstantinos Gatsis and Prof. Jack Umenberger for their advice and openness to chat about research ideas.

Beyond my research, I believe I have also greatly developed my teaching skills while at Oxford, in particular, thanks to Dr Isaac Mear. Your passion for teaching and advice throughout the years has greatly influenced how I engage with students and has motivated me to stay in Higher Education.

I would also like to thank Dr Licio Romao, for providing me with guidance and advice over the years as I progressed through my DPhil and finally into my upcoming PostDoc.

During the past years, I encountered many friends both within and outside of the university that have helped me in completing this journey. I would like to

thank Yana Lishkova and Matthew Newton for the wonderful chats and coffee breaks over the years.

Thank you Steven Starr, for the long chats during the height of the Covid-19 Pandemic and for the deep friendship we have built over the years. Your mentorship, in particular while we were fundraising together, has been invaluable and I look back fondly on the work we did over the years.

I will be forever grateful to Lisa Takashima for allowing me to live with you and welcoming me into your home during the pandemic. My time with you and Ky was a wonderful experience and has taught me much.

Finally, I would like to thank my partner, Kori Malia. Over the last four years, you have been with me every step of the way throughout this journey. I am deeply grateful for your unwavering support. Completing this journey with you by my side has made it one of the greatest of my life. Thank you. Without the support of my parents, however, none of this would have been possible. Thank you to my mother, Astrid Gräfe-Vertovec and father, Steven Vertovec. You have supported me so much over the years and have given me guidance and advice, without which I would not be here today.

Kori, Astrid and Steve, I dedicate this thesis to you and am forever grateful for your patience with me, your continuous support and your love.

Abstract

Enforcing safety plays a crucial role within the optimisation and control literature. Despite notable advances in recent years, optimal control for safety-critical and high-dimensional systems remains a challenging problem. Developing a general theoretical framework for integrating safety within optimal control is not tractable as the numerical inaccuracy and computational cost often grow exponentially with the number of states. Furthermore, different notions of safety require different methodologies and present unique theoretical and computational challenges. This thesis focuses on the challenges that arise when addressing scalability and safety considerations simultaneously.

Safety is a multi-facet problem that involves hard constraint satisfaction, avoiding sharing information considered as private, as well as robustifying towards uncertainty that could otherwise compromise safety. The initial chapters of the thesis focus on Hamilton-Jacobi reachability, which has become a well-established method of computing reachable sets for complex nonlinear systems. In addition to enforcing that the system remains within a *safe* part of the state-space, we consider application-specific abstractions to deal with scalability, the interplay between competing performance objectives and safety objectives, and the challenges arising from multi-objective optimal control problems. We then investigate safety considerations due to the amount of information that needs to be shared between agents in a multi-agent networked control setting. Extending classical state-aggregation in approximate dynamic programming, we introduce a method of solving a large-scale Markov Decision Process in a fully distributed manner. The final chapter considers stochastic safety constraints under a statistical learning theoretic lens. Utilising randomised algorithms, we establish probably approximately correct bounds on predicting a future label in a binary classification problem whereby the classifier changes in an unknown structured manner.

Contents

List of Figures	xiii
List of Tables	xvii
1 Introduction	1
1.1 Motivation	1
1.2 Thesis outline	2
1.3 Contributions	5
2 Multi-objective Hamilton-Jacobi reachability analysis	11
2.1 Introduction	11
2.2 Mathematical description and physical modelling	14
2.2.1 Spacecraft equations of motion	14
2.2.2 State constraints	16
2.3 Problem Statement	18
2.3.1 Multi-objective optimal control problem	18
2.3.2 Pareto optimality	19
2.4 Solution to multi-objective optimal control problems	20
2.4.1 Extension to problems in Bolza form	22
2.4.2 Value function computation	28
2.5 Numerical Approximation and Results	31
2.5.1 Implementation	32
2.5.2 Simulation results	34
2.6 Conclusion	40
2.7 Appendix	41
3 Verification of safety-critical control policies	45
3.1 Introduction	45
3.2 Problem Setup	47
3.3 Using Gaussian Processes to verify a control policy	50
3.3.1 Error of the value function	50
3.3.2 Prediction using Gaussian Processes	52
3.4 Case Study	53

3.4.1	A Game of Two Identical Vehicles	53
3.4.2	Gaussian process error prediction	54
3.4.3	Effects of mismatch in dynamics	58
3.5	Conclusion	59
4	Safety-Aware Hybrid Control of Airborne Wind Energy Systems	61
4.1	Introduction	61
4.2	Modelling	65
4.2.1	Reference Frames	65
4.2.2	Aircraft Equations of Motion and Ground Station Model . .	67
4.2.3	Optimal Flight Path and Guidance	71
4.3	Model Abstraction and Safety Considerations	73
4.3.1	Simplified Tether Model	73
4.3.2	Adversarial Wind Turbulence	76
4.3.3	Γ Frame and Safety Control Model	77
4.4	HJ Reachability Analysis and Controller Synthesis	79
4.4.1	Problem Statement	79
4.4.2	Optimal Control and Disturbance Inputs	82
4.4.3	Hybrid Control Setup	86
4.5	Simulation	88
4.5.1	Simulation Setup	88
4.5.2	Simulation Results	90
4.6	Conclusion	93
4.7	Appendix	94
5	State Aggregation for Distributed Value Iteration in Dynamic Programming	97
5.1	Introduction	97
5.2	Problem Setup	99
5.2.1	Multi-agent Markov Decision Processes	99
5.2.2	Distributed Value Iteration	100
5.3	Algorithm analysis	103
5.4	Application to traffic routing	108
5.4.1	Simulation Set-up	108
5.4.2	Simulation results	109
5.5	Conclusion	112

6	Finite sample learning of moving targets	113
6.1	Introduction	113
6.2	Learning moving targets	115
6.2.1	Problem statement	115
6.2.2	Mathematical preliminaries and assumptions	118
6.3	Finite sample probabilistic certificates	120
6.3.1	Main result	120
6.3.2	Hypothesis computation	125
6.4	Numerical example	130
6.4.1	Problem set-up	130
6.4.2	Simulation results	134
6.5	Conclusion	137
7	Summary and future research directions	139
7.1	Safety in the sense of state constraint satisfaction	140
7.1.1	Multi-objective Hamilton-Jacobi reachability analysis	140
7.1.2	Verification of safety-critical control policies	141
7.1.3	Safety-Aware Hybrid Control of Airborne Wind Energy Systems	142
7.2	Safety in the sense of information privacy	142
7.3	Safety in the sense of robustness against uncertainty	143
	References	145

List of Figures

2.1	Gravitational acceleration comparing U_ρ and U_θ around Castalia 4769.	34
2.2	The evolution of the value function, ω , projected in the ρ - v_t plane for $v_\rho = 0$ and $\Delta m = 100$ g.	35
2.3	The evolution of the zero level set of the value function, ω , projected in the ρ - v_t plane for $v_\rho = 0$ and $\Delta m = 100$ g.	35
2.4	Initial orbit and transfer trajectory to a circular orbit ≈ 1 km further away from the asteroid.	36
2.5	Pareto front of the objective functions $J_1 = \Delta m$ in grams and $J_2 = t_f$ in seconds. The trajectory in Figure 2.4 is derived using the point marked by the black diamond, while the time-optimal solution used in Figure 2.6 uses the point denoted by the magenta pentagram.	37
2.6	A comparison of the thrust magnitude of the smoothed optimal control policy of the time-optimal solution and the trajectory presented in Figure 2.4.	38
2.7	Pareto front of the objective functions $J_1 = z$ in grams and $J_2 = t_f$ in seconds, for the Bolza problem. The point used for trajectory calculations is marked by the black diamond.	39
3.1	Value function of the pursuit-evasion example in the x_1 - x_2 plane sliced along $x_3 = 0.25$. The sub-zero level set (within the dark blue line) indicates the backwards reachable tube (BRT). The boundary of the avoid set is shown by the dark red line.	55
3.2	The prediction capabilities of the rational quadratic Gaussian process regression (GPR) model trained using 100 and 1000 observations respectively.	57
3.3	The correction capabilities of the rational quadratic GPR model trained using 100 (left) and 1000 (right) observations, respectively. As is expected, the regions of mismatch become more distinguished as we increase the number of observations.	58
3.4	The root-mean-square error (RMSE) of a GPR model trained using 1000 samples as we vary v_e and v_p	59

4.1	Tether rupture during the traction phase. The red dot marks the point during flight at which a tether rupture occurs due to the maximum tether force being exceeded.	63
4.2	Visualisation of the North-East-Down (NED) frame (blue), intermediate aerodynamic frame $\bar{A}$ (orange), the aerodynamic frame A (green), the body frame B (red) and the tangential frame (purple). .	67
4.3	(left) Visualisation of the tether modelled as a straight tether (red) and as a fixed number of lumped masses (blue). The velocities of the point masses are shown by the red and blue vectors respectively. (right) Visualisation of the guidance strategy to produce the optimal course heading (purple arrow).	70
4.4	$\dot{\Delta}_t$ during the traction phase of flight, using the primary path-following controller.	75
4.5	The behaviour of the wind gusts during 50 seconds of flight.	76
4.6	Visualisation of the Γ frame. s is used to denote the position along the reference curve. The geodesic vector, σ , describes the distance to the reference curve.	77
4.7	Comparison of a 31×7 grid in the NED frame vs the Γ frame. The Γ frame successfully captures only relevant nodes close to the optimal tracking curve.	82
4.8	Visualization of the value function used for safety controller synthesis projected along $[s = 4\pi/3, \sigma = 20m, h_\tau = 250m, v_a = 31m/s, \Delta_t = 0.3mm]$	86
4.9	State automaton of the hybrid control setup.	88
4.10	The tether force acting on the aircraft during the avoidance manoeuvre under worst-case disturbances. Each colour represents a random initial position, velocity, and tether force and the evolution of the tether force as the avoidance manoeuvre is executed.	89
4.11	The tensile force of the tether acting on the aircraft during flight. .	90
4.12	The tether force in % of the maximum allowed tether force before a rupture occurs.	91
4.13	The safety controller only needs to be activated at particular times. For the remainder of the flight, a more power-optimal controller, such as the nonlinear dynamic inversion (NDI) controller can be employed.	91
4.14	The power P_{mech} that is generated at the ground station during flight.	93
4.15	Generated power in relation to the aircraft's flight path. Negative (resp. positive) power indicates a reeling in (resp. out) by the winch thus power is exerted (resp. harvested).	94

5.1	Graph representation of the East-Oxford road network. Junctions are represented by vertices and edges represent roads.	108
5.2	The complete Oxford road network subdivided into 5 subgraphs representing the aggregation sets.	109
5.3	The node cost sorted by size. The blue line indicates the <i>true cost</i> computed using conventional value iteration, the red markers represent the cost obtained by the proposed distributed algorithm. .	111
5.4	The deviation of the cost obtained by the proposed distributed algorithm with respect to the <i>true cost</i> . The node numbers correspond to those of Figure 5.3.	111
5.5	The aggregated cost r evolving over time with minimal communication between agents.	112
6.1	For each i , the set B_i changes and we receive a single sample, x_i , labelled either by 1 (red star) or 0 (blue diamond), to indicate if the drawn sample lies within the set B_i . We seek to find a hypothesis, h_m on the basis of m observed samples labelled according to the procedure that involves a moving labelling mechanism, such that with certain confidence we can predict the (unknown) label, f_{m+1} , of the subsequent sample.	117
6.2	Number of samples required according to (6.11) for different accuracy levels ϵ and $\delta = 10^{-6}$. The dashed line indicates the first bound in (6.11), while the solid line the second one. The colour code corresponds to different values of $\underline{\mu}, \bar{\mu}$. For a given accuracy level the minimum number of samples required is denoted by the intersection of these lines, e.g., considering $\epsilon = 5\%$ and $\bar{\mu} = 10\%$, as long as $\underline{\mu} \geq 1.25\%$ (dotted red line), choosing $m \geq 4.7 \cdot 10^4$ samples will result in the desired $\epsilon - \delta$ guarantees.	124
6.3	The evolution of the braking performance over time. Green circles indicate samples with label 0, while blue triangles show samples with label 1. The bold red halfplane represents the true safety label at the given iteration, while the opaque halfplanes show the safety boundary at previous iterations.	131
6.4	Illustration of the facets of the convex polytope. Since the safety label relies in this case on a single halfplane (drawn in red), we only need to consider facet 3.	133
6.5	All samples in black are discarded, while the red samples are kept for the computation of the hypothesis. This results in 95% of the samples being omitted, greatly improving the computational feasibility.	134

6.6	The generated hypothesis; we only show the halfplane responsible for the labelling, illustrated by red. For visual ease, we randomly omit samples close to one another. Red samples are violations as defined in (6.33).	135
6.7	Empirical distribution of the disagreement $\hat{e}_m(f_{m+1}, h_m)$, constructed by means of 500 Monte Carlo runs.	136

List of Tables

2.1	Normalisation values	32
2.2	Spatial grid configuration	34
2.3	Comparison of the accuracy, CPU time, and memory usage as the size of the grid changes.	37
2.4	Spatial grid configuration for the Bolza problem.	38
2.5	Comparison of the accuracy, CPU time, and memory usage for the second formulation in Bolza form.	40
3.1	Comparison of commonly used regression models to approximate the numerical error of the value function.	56
3.2	Comparison of the ability to correct the value function.	58
4.1	Simulation Parameters	89
5.1	Normalised average error with respect to the number of partitions/agents.	110

Acronyms

- AEB** Autonomous Emergency Braking. 5, 9, 130, 132, 133, 137
- AWE** Airborne Wind Energy. 4, 7, 61, 62, 64, 65, 68, 71, 73, 82, 92, 93, 142
- BRS** backwards reachable set. 63, 80, 81, 86, 88
- BRT** backwards reachable tube. 46–50, 54, 55, 141
- DOF** degrees of freedom. 67
- ENO** essentially non-oscillatory. 54, 82
- GP** Gaussian process. 47, 52, 53
- GPR** Gaussian process regression. 6, 52, 55–59
- HJ** Hamilton-Jacobi. 3–7, 12, 13, 40, 46, 47, 59, 63–65, 79, 86, 88, 93, 94, 140–142
- HJB** Hamilton-Jacobi-Bellman. 3, 12, 13, 40
- MDP** Markov Decision Process. 8, 98, 99
- MILP** Mixed Integer Linear Program. 9, 115, 126, 129, 130, 133–137, 143, 144
- MOCP** multi-objective optimal control problem. 2, 3, 6, 11, 13, 14, 18–21, 24, 25, 28, 36, 38, 40, 140
- NDI** nonlinear dynamic inversion. 62, 63, 86, 91
- NED** North-East-Down. 66, 67, 78, 81, 82, 84, 142
- PAC** probably approximately correct. 5, 8, 9, 114, 117, 118, 123, 137, 143
- RMSE** root-mean-square error. 55, 56, 58, 59
- SOI** sphere of influence. 16

SVM support vector machine. 56, 62

VC Vapnik-Chervonenkis. 5, 8, 113, 116, 119, 120, 132

WENO weighted essentially non-oscillatory. 31

1

Introduction

1.1 Motivation

Safety-critical optimal control problems are problems whereby we seek to minimise one or more objective functions subject to safety constraints. Since safety is a multifaceted problem, such safety constraints will appear in a variety of forms, including the satisfaction of state-space constraints, avoiding the sharing of information considered to be private, as well as being robust towards uncertainty that could otherwise compromise safety. These safety-critical optimal control problems are found in a variety of fields such as control theory [83, 84, 87, 125], medicine [100, 134, 152], emergency rescue [91, 154, 155], and power systems [42, 139, 156]. While we have seen several notable advances in the past decades [6, 13], many optimal control approaches in the safety-critical control literature still become intractable when applied to high-dimensional systems. To this end, it becomes increasingly relevant to exploit application-specific properties so as to overcome the scalability issues arising from high-dimensional optimal control problems.

As an example, consider the design of a trajectory around a rotating asteroid. The high dimensional and complex dynamics make this a particularly challenging safety-critical control problem. Without exploiting specific knowledge of the application, conventional safety-critical optimal control approaches would not be able to solve the

underlying control problem. However, by exploiting knowledge of the gravitational field to omit states and reformulating the multi-objective optimal control problem (MOCP), we are able to solve the problem to high degrees of accuracy.

With safety-critical applications found in a plethora of fields, it becomes vital to consider scalability issues in the context of varying notions of safety. Each notion of safety brings with it its own set of challenges. In the context of reachability, safety can be considered by means of meeting certain safety-critical constraints that prevent the system from entering unsafe regions of the state-space. These regions might, for instance, equate to a collision or mechanical failure. Another notion of safety is robustness towards uncertainty and/or external disturbances. Additional challenges of safety emanate from application-specific constraints, such as limitations on available data, or application-specific requirements such as when we seek to optimise an optimal control problem while maintaining a compromise between two competing objectives. This necessitates the consideration of MOCPs. This thesis provides a step towards overcoming these challenges and will address the incorporation of different notions of safety into a variety of high-dimensional optimal control problems.

To address the large variations in the notions of safety, we have identified three distinct notions which will be dealt with separately. The first will consider the classical notion of safety as found in the control theory literature, i.e. the satisfaction of safety-critical state-space constraints. The second will consider how we can avoid sharing information considered as potentially private. The third will consider robustifying towards uncertainty that could otherwise compromise safety.

1.2 Thesis outline

In this thesis, we consider reachability analysis as our primary method of synthesising controllers within the classical notion of safety, i.e. the satisfaction of safety-critical state-space constraints. Reachability analysis aims to find the set of points from which a target can be reached within a given time, subject to constraints. It forms a fundamental part of the dynamics and control literature and has been used extensively for controller synthesis of complex systems [9, 94, 104]. In recent

years we have seen considerable research being conducted into computing reachable sets using Hamilton-Jacobi (HJ) reachability analysis, whereby the reachable set is derived from the viscosity solution of a Hamilton-Jacobi-Bellman (HJB) equation accounting also for the presence of state constraints. Such a HJB framework is presented in [23, 97] with more general value problems bypassing previous regularity issues presented in [5]. In [64] an extension of the HJB framework to time-varying targets and constraints is considered and in [46] the approach is extended to multi-objective optimal control problem (MOCP)s. The incorporation of MOCPs into the framework of Hamilton-Jacobi (HJ) reachability opens up the approach to a vast number of possible applications such as those found in spacecraft trajectory design. Common competing objectives in spacecraft trajectory design are minimising the amount of propellant as well as the flight time required to execute an orbital transfer. For safe operation, a compromise between these objectives needs to be found. However, in the classical formulation of multi-objective HJ reachability, it is not possible to incorporate the flight time as an optimisation variable. Furthermore, each incorporated additional objective adds an additional state, thus the scalability issues are greatly exacerbated. In light of this, Chapter 2 uses a low thrust trajectory design problem to highlight and address key issues concerning scalability.

Proceeding with our study of applying HJ reachability to high-dimensional optimal control problems, Chapter 3 introduces an approach for analysing and quantifying numerical inaccuracies within HJ reachability by means of a Gaussian process. By empirically quantifying the error/accuracy of the obtained control policy we confine the numerical precision to within a predefined limit. At the same time, we leverage the marginal likelihood of the Gaussian process as a robustness metric in a hybrid switching law. We further extend the capabilities of utilising hybrid switching laws to address scalability in Chapter 4, where we consider the interplay between safety considerations and performance.

Since considering safety and robustness at all times during controller synthesis, in general leads to conservative results that can deteriorate overall performance, it is often of interest to only enforce safety under certain conditions or within

predefined areas of the state-space. The core concept underpinning the hybrid switching law approach involves combining a safety-critical control policy obtained from HJ reachability with another control policy synthesised to operate under a broader set of conditions [65, 74]. To operationalise this, a customised switching law tailored to the application’s requirements needs to be devised. In light of this, Chapter 4 provides a comprehensive analysis of integrating safety-critical control within the emerging field of Airborne Wind Energy (AWE). Traditional models of AWE systems often encompass more than 30 states and involve complex nonlinear and non-affine dynamics, making many safety-critical control approaches, including HJ reachability, intractable. Consequently, AWE offers an ideal opportunity to showcase the effectiveness of safety-aware hybrid switching laws.

The issues of scalability, however, are not limited to HJ reachability and can be found across the field of safety-critical optimal control. Outside the *classical* notion of safety, scalability issues are found in multi-agent settings, where the safety constraints limit the information shared between agents, due to, for instance, privacy concerns. We thus focus on dynamic programming problems in discrete-time systems with a large number of states and attempt to devise a scalable solution using tools from state aggregation and multi-agent optimisation. In particular, states with a similar expected cost are aggregated into virtual auxiliary states, reducing the total number of states considered during optimisation [21]. Conventional state aggregation has several drawbacks. One notable drawback when considering very large problems is that we require complete information on the cost-to-go and transition capabilities. For systems where the cost-to-go or transition probabilities are constantly being updated and the optimal control problem needs to be solved repeatedly, this information overhead considerably limits the possibilities that state aggregation could provide. In light of this, Chapter 5 of the thesis addresses our second distinct notion of safety and presents a fully distributed method of approximate dynamic programming using state aggregation with minimal communication overhead between agents.

Chapter 6 of the thesis considers a sample based approach to tackling the third distinct notion of safety identified, i.e. robustifying towards uncertainty that could otherwise compromise safety. Sample based approaches allow us to overcome dimensionality issues implicitly and can address the interplay of stochastic (soft) and worst-case (hard) safety constraints by utilising algorithms based on uncertainty randomisation, known as *randomised algorithms* [136]. Randomised algorithms and Vapnik-Chervonenkis (VC) theory have been successfully applied to control in the past [4, 136, 149], yet many safety-critical applications, such as Autonomous Emergency Braking (AEB) systems are still an open research topic, as the underlying labelling mechanism used to classify obtained samples as safe or unsafe, is subject to change. In light of this, we will derive a probably approximately correct (PAC) learning algorithm to deal with labelling mechanisms subject to (structured) change.

1.3 Contributions

The contributions of this thesis span several related areas. Below, we provide a brief outline of each chapter of the thesis and the main contributions.

- In Chapter 2, we investigate a spacecraft trajectory design problem around an asteroid, where the objective is to use minimal amounts of propellant to raise an orbit while keeping flight times as short as possible. To find the compromise between two competing objectives in conjunction with solving the trajectory design problem, we introduce a generalisable methodology of combining the value function inherent in HJ reachability with standard multi-objective optimisation problem solvers. Conventionally, such spacecraft trajectory design problems are solved by either indirect methods, based on the calculus of variations, or direct methods, whereby the optimal control problem is reformulated as a nonlinear programme. Dynamic programming is usually not tractable in the context of trajectory design as it is hampered by the curse of dimensionality. Yet unlike direct approaches, dynamic programming can guarantee optimality of the solution, and unlike indirect approaches,

the solution does not rely on a carefully chosen initial approximation of the optimal trajectory. Our approach enables the use of dynamic programming for arbitrary multi-objective problems in Bolza form, without the use of auxiliary states. This proposed generalisable methodology constitutes our first contribution. As alluded to earlier, spacecraft trajectory design is an excellent case study to demonstrate how application-specific properties can be exploited to further overcome dimensionality issues. Thus the second contribution of Chapter 2 constitutes the formulation of an efficient constrained MOCP for low-thrust spacecraft trajectory design, whereby we optimise only over the set of admissible initial states and transition times, while the third contribution is the reduction of the state-space through the use of appropriate approximations. This chapter is based on the papers

- Nikolaus Vertovec, Sina Ober-Blöbaum, and Kostas Margellos. “Multi-objective minimum time optimal control for low-thrust trajectory design”. In: *2021 European Control Conference (ECC)*. 2021, pp. 1975–1980. DOI: 10.23919/ECC54610.2021.9654919
- Nikolaus Vertovec, Sina Ober-Blöbaum, and Kostas Margellos. “Multi-objective low-thrust spacecraft trajectory design using reachability analysis”. In: *European Journal of Control* 69 (2023), p. 100758. DOI: 10.1016/j.ejcon.2022.100758
- In Chapter 3, we analyse the effects of numerical error and model mismatches on the value function of an underlying HJ reachability problem. This empirical work focuses on exploring the feasibility of applying non-parametric approaches such as Gaussian processes to learn deviations and provide a confidence metric for a hybrid control law. Furthermore, we are able to analyse the efficacy of the value function and control policy obtained by means of HJ reachability, as well as the use of the empirical error to iteratively correct and improve the value function and control policy. By demonstrating the efficacy of using a Gaussian process regression (GPR) in conjunction with HJ reachability,

we provide a promising tool for reducing the computational complexity of synthesising controllers by means of HJ reachability. This, in turn, increases the potential of applying HJ reachability to higher dimensional systems. This chapter is based on

- Nikolaus Vertovec, Sina Ober-Blöbaum, and Kostas Margellos. “Verification of safety critical control policies using kernel methods”. In: *2022 European Control Conference (ECC)*. 2022, pp. 1870–1875. DOI: 10.23919/ECC55457.2022.9838224
- In Chapter 4 we consider enforcing safety with minimal implications on performance. This is a common challenge in many safety-critical applications, especially within the energy sector. To this end, we consider Airborne Wind Energy (AWE) systems, a promising renewable technology that harnesses the power of strong and reliable winds found at high altitudes. As with many power systems, the safety objectives are directly competing with the performance objectives, as the highest power is generated when the system is pushed to its mechanical (and thus safety) limits. Rather than enforcing safety at all times, we introduce a hybrid control approach to integrate a safety control policy into an existing control architecture. The high dimensional, complex and non-affine dynamics of AWE introduce additional challenges for controller synthesis, necessitating the derivation of a simplified low dimensional AWE model that is accurate enough for safety-critical control purposes. This spans the first contribution, with the second contribution involving the synthesis and deployment of an avoidance control strategy based on the developed abstractions and HJ reachability analysis. Furthermore, we introduce and discuss a hybrid switching law tuned to the underlying system and validate the performance of the hybrid controller by applying it to a high-fidelity model. This chapter is based on

- Nikolaus Vertovec, Sina Ober-Blöbaum, and Kostas Margellos. “Safety-Aware Hybrid Control of Airborne Wind Energy Systems”. In: *Journal*

of Guidance, Control, and Dynamics 47.2 (2024), pp. 326–338. DOI: 10.2514/1.G007254

- In Chapter 5 we consider a discrete, high dimensional control problem formulated as a Markov Decision Process (MDP). The notion of safety treated in this chapter is to curtail the amount of information that needs to be shared between agents in a distributed control context. This could be motivated by privacy concerns or limitations on communication bandwidth. In light of this, we propose a distributed value iteration algorithm for multi-agent dynamic programming, preventing transition probabilities and state costs from constituting global information among agents. We provide a rigorous mathematical analysis of the presented algorithms proving both convergence and optimality properties. Finally, the presented methodology is demonstrated on a non-trivial traffic routing problem. This chapter is based on
 - Nikolaus Vertovec and Kostas Margellos. “State Aggregation for Distributed Value Iteration in Dynamic Programming”. In: *IEEE Control Systems Letters* 7 (2023), pp. 2269–2274. DOI: 10.1109/lcsys.2023.3285655
- In Chapter 6 we consider safety from a stochastic viewpoint. Enforcing safety by means of a *randomised algorithm* involves the random extraction of uncertainty samples and returning a probability of satisfying a certain safety constraint. This is analogous to a binary classification problem whereby a 0-1 labelling mechanism classifies a randomly drawn sample as safe or unsafe. When both the probability of drawing a certain sample as well as the labelling mechanism are unknown, results from Vapnik-Chervonenkis (VC) theory can be applied to provide a probably approximately correct (PAC) algorithm for generating a hypothesis that will, with high likelihood, correctly predict the label of a subsequently received sample. We extend these results to consider problems whereby the labelling mechanism changes with some structure after each drawn sample. In addition to proving that such a problem remains

probably approximately correct (PAC), we present a constructive method of generating the hypothesis using a Mixed Integer Linear Program (MILP). We illustrate the computational viability of our approach by applying it to an Autonomous Emergency Braking (AEB) procedure. This chapter is based on

- Nikolaus Vertovec, Kostas Margellos, and Maria Prandini. “Finite sample learning of moving targets”. to be submitted to *Automatica*

2

Multi-objective Hamilton-Jacobi reachability analysis

2.1 Introduction

Since the Galileo mission in 1991, we have seen a steady increase in proposed missions to asteroids and comets, as they might hold the key to many scientific questions including the origins of life on earth [7]. The Dawn mission to Vesta and Ceres proved the viability of low-thrust electric propulsion for asteroid exploration [3, 111], and it is expected that many upcoming missions will rely on similar low-thrust propulsion. While there has been a significant study of interplanetary transfer trajectories using low-thrust propulsion, comparatively little research has been conducted on the trajectory design in the vicinity of asteroids. Such a spacecraft trajectory design problem is a classical safety-critical control problem whereby we can formulate the safety constraints as state-space constraints. Yet additional challenges arise since the objective is to use minimal amounts of propellant to raise an orbit while keeping flight times as short as possible. This is a multi-objective optimal control problem (MOCP), whereby one seeks to find the optimal way a dynamical system can perform a certain task, while minimising or maximising a set of, usually contradictory and incommensurable, objective functions [108]. Conventional optimisation techniques for spacecraft trajectory design often fall

into two categories, indirect methods, based on the calculus of variations, and direct methods, whereby the optimal control problem is reformulated as a nonlinear programme. Direct approaches rely on parameterisation and while a candidate solution is found, there are no guarantees on the optimality of the solution. Indirect methods, meanwhile rely on necessary analytic conditions for optimality using Lagrange multipliers. Yet while optimality of the obtained solutions may be guaranteed, indirect approaches, such as multiple shooting methods, rely on a good initial approximation of the optimal trajectory [133, 150]. A third approach is dynamic programming, whereby the optimality conditions are formulated in continuous time based on the HJB equation [133], however, it is hampered by the so-called curse of dimensionality. Despite this, unlike direct approaches, optimality is guaranteed, and unlike indirect approaches, the solution does not rely on an initial approximation of the optimal trajectory. We expand on this third approach by taking advantage of recent developments in reachability analysis.

Reachability analysis aims to find the set of points from which a target can be reached within a given time, subject to constraints. It forms a fundamental part of the dynamics and control literature and has been used extensively for controller synthesis of complex systems [9, 94, 104]. In recent years we have seen considerable research being conducted into computing reachable sets using Hamilton-Jacobi (HJ) reachability analysis, whereby the reachable set is derived from the viscosity solution of an HJB equation accounting also for the presence of state constraints. Such an HJB framework is presented in [23, 97] with more general value problems bypassing previous regularity issues presented in [5]. In [64] an extension of the HJB framework to time-varying targets and constraints is considered and in [46] the approach is extended to multi-objective control problems. HJ reachability has also been successfully applied to various aerospace applications including air traffic control [98], the climbing problem of multi-stage launchers [24], payload optimisation [25], as well as most recently to the complete model of the ascent problem of multi-stage launchers [22]. One of the advantages of using HJ reachability is that the optimal trajectory can easily be constructed once the reachable set has

been computed. This makes HJ reachability attractive for problems that require computing trajectories for various different initial states.

For the spacecraft trajectory design problem considered in this chapter, there are two possible formulations for minimising the burnt propellant. The first assumes that the initial mass is a free optimisation variable. This approach is common during mission design where the total required fuel budget is being calculated. The second formulation of the spacecraft trajectory design problem assumes a fixed initial mass and the objective is to maximise the remaining mass after completing a given orbital manoeuvre. This approach is more common when the manoeuvre needs to be added to a given mission and the available fuel is non-negotiable.

To solve the first formulation of the spacecraft trajectory design problem, we formulate the problem as a MOCP and show that it can be equivalently stated as an optimisation problem subject to a constraint that involves the sub-level set of a certain value function. The latter is shown to be the unique continuous viscosity solution of a quasi-variational inequality that involves an HJB equation. Such value functions have been defined in [23, 97] to account for the presence of state constraints. This formulation allows characterising the Pareto front of the formulated MOCP and also facilitates its computation by means of available numerical tools.

To solve this second formulation of the spacecraft trajectory design problem, we draw on research from [46] and extend our formulation of the MOCP to introduce an auxiliary state, allowing us to account for arbitrary problems in Bolza form, and, together with appropriate normalisation and approximations allowing for a reduction of the state-space, greatly improving on the method presented in [145].

Thus our contributions can be summarised as

1. the formulation of an efficient constrained MOCP for low-thrust spacecraft trajectory design that optimises only over the set of admissible initial states and transition times,
2. the reduction of the state-space through the use of appropriate approximations,

3. the expansion of the MOCP to allow for a generalisation of the proposed methodology for arbitrary multi-objective problems in Bolza form.

This chapter is organised into six sections. Section 2.2 contains details regarding the derivations of the spacecraft dynamics as well as the definitions of the constraints pertaining to its behaviour. In Section 2.3 the optimal control problem is formulated while Section 2.4 describes how the set of admissible initial states is derived from the viscosity solution of a quasi-variational inequality. Section 2.5 is dedicated to the numerical computation and case study of an orbital transfer around a rotating asteroid. Finally, Section 2.6 provides concluding remarks.

2.2 Mathematical description and physical modelling

2.2.1 Spacecraft equations of motion

We begin by modelling the dynamics of the spacecraft. The spacecraft thrust is defined in spherical coordinates as

$$u(t) := [\alpha(t), \delta(t), T(t)] \in \mathcal{U}, \quad (2.1)$$

where $\alpha(t) \in [-\pi, \pi]$ is the incidence angle, $\delta(t) \in [-\frac{\pi}{2}, \frac{\pi}{2}]$ is the sideslip angle and $T(t) \in [0, T_{\max}]$ is the variable thrust, with $T_{\max}$ denoting the maximal allowable thrust. The Cartesian transformation of the thrust vector is denoted by $u_x(t)$, $u_y(t)$ and $u_z(t)$, respectively. The compact set $\mathcal{U} = [-\pi, \pi] \times [-\frac{\pi}{2}, \frac{\pi}{2}] \times [0, T_{\max}]$ is the set of possible control input values while $u(\cdot) \in \mathcal{U}_{ad}$ denotes the control policy and $\mathcal{U}_{ad}$ denotes the set of admissible policies which is the set of Lebesgue time measurable functions from $[-\infty, 0]$ to $\mathcal{U}$. Note that time here is considered to be non-positive to facilitate the reachability problem exposition in Section 2.3. The equations of motion of the spacecraft around a rotating body can be expressed in 3-dimensional Euclidean space as a second-order ordinary differential equation (see eg., [81])

$$\begin{aligned} 2\Omega(t) \times \frac{dR(t)}{dt} + \Omega(t) \times (\Omega(t) \times R(t)) + \frac{dU(R(t))}{dR} \\ + \frac{d\Omega(t)}{dt} \times R(t) - \frac{u(t)}{m(t)} = -\frac{d^2R(t)}{dt^2}, \end{aligned} \quad (2.2)$$

where $R(t)$ is the radius vector from the asteroid's centre of mass to the particle, the first and second time derivatives of $R(t)$ are with respect to the body-fixed coordinate system, $U(R(t))$ is the gravitational potential of the asteroid and Ω is the rotational angular velocity vector of the asteroid relative to inertial space. The term $2\Omega(t) \times \frac{dR(t)}{dt}$ describes the Coriolis forces, $\Omega(t) \times (\Omega(t) \times R(t))$, the centrifugal forces, and $\frac{d\Omega(t)}{dt} \times R(t)$ the Euler forces. We consider an asteroid rotating uniformly with constant magnitude ω around the z-axis. Therefore, the Euler forces can be neglected and we can express the rotation vector as $\Omega := \omega e_z$, where e_z is the unit vector along the z-axis. Following [70], the radius vector and its derivatives are given by

$$R(t) := \begin{bmatrix} x(t) \\ y(t) \\ z(t) \end{bmatrix}, \quad \frac{dR(t)}{dt} = \begin{bmatrix} v_x(t) \\ v_y(t) \\ v_z(t) \end{bmatrix}. \quad (2.3)$$

The Coriolis and centrifugal forces acting on the spacecraft (the first two terms in Equation (2.2)) are thus

$$2\Omega \times \frac{dR(t)}{dt} = \begin{bmatrix} -2v_y(t) \\ 2\omega v_x(t) \\ 0 \end{bmatrix}, \quad (2.4)$$

$$\Omega \times (\Omega \times R(t)) = \begin{bmatrix} -\omega^2 x(t) \\ -\omega^2 y(t) \\ 0 \end{bmatrix}. \quad (2.5)$$

To model the current position, velocity, and available propellant, we define the state vector

$$r := [x, y, z, v_x, v_y, v_z, \Delta m]^T \in \mathbb{R}^7, \quad (2.6)$$

where $\Delta m \in \mathbb{R}_{\geq 0}$ denotes the available propellant. The total spacecraft mass can be expressed as $m(t) = m_0 + \Delta m(t)$, where m_0 denotes the dry mass of the spacecraft. Following our derivations from Equation (2.2), we can formulate the dynamics of the spacecraft, $\dot{r} = f(r, u)$, as

$$f(r, u) = \begin{bmatrix} v_x \\ v_y \\ v_z \\ U_x(x, y, z) + \omega^2 x + 2\omega v_y + \frac{u_x}{m_0 + \Delta m} \\ U_y(x, y, z) + \omega^2 y - 2\omega v_x + \frac{u_y}{m_0 + \Delta m} \\ U_z(x, y, z) + \frac{u_z}{m_0 + \Delta m} \\ -\frac{\sqrt{u_x^2 + u_y^2 + u_z^2}}{v_{\text{exhaust}}} \end{bmatrix}, \quad (2.7)$$

where $v_{\text{exhaust}} \in \mathbb{R}_{\geq 0}$ is the exhaust velocity used to express the depletion of mass as propellant is burned, U_x , U_y and U_z are the derivatives of the gravitational potential in the direction of the unit vectors e_x , e_y and e_z , respectively, and where for brevity we neglect the time dependence by denoting $r = r(t)$, $v_x = v_x(t)$, and similarly for the other states.

2.2.2 State constraints

Since the dynamics of the spacecraft were derived for orbits in the vicinity of the asteroid, we need to enforce state constraints on x, y, z . We naturally also need to ensure that we bound the amount of propellant available. Assuming that the burnout mass of the spacecraft is the same as the dry mass, we set $m_{\min} := 0$ and $m_{\max} := m_{\text{propellant}}$ and impose $m_{\min} \leq \Delta m \leq m_{\max}$.

Due to particles ejected from the asteroid, we do not want to fall below a circular orbit with radius $\rho := \sqrt{x^2 + y^2 + z^2}$ of approximately $\rho_{\min} = 1$ km. Furthermore, in order for the two-body problem under discussion to be valid and the influence of other bodies in the solar system to be negligible, we need to stay within the sphere of influence (SOI) of the asteroid. The SOI can be approximated as in [131] by $\rho_{\text{SOI}} \approx a \left(\frac{M_1}{M_2} \right)^{\frac{2}{5}}$, where a is the semi-major axis of the asteroid's orbit around the sun ($1.5907 \cdot 10^8$ km), M_1 is the Mass of the asteroid ($1.4091 \cdot 10^{12}$ kg) and M_2 is the mass of the sun ($1.9890 \cdot 10^{30}$ kg), resulting in $\rho_{\max} = \rho_{\text{SOI}} \approx 8.74$ km. The set of states that satisfy the aforementioned restrictions is given by

$$\mathcal{K} := \left\{ r \in \mathbb{R}^7 : \rho \in [\rho_{\min}, \rho_{\max}], m \in [m_{\min}, m_{\max}] \right\}.$$

The target orbit that we would like to transfer to is denoted by the closed target set $\mathcal{C} \subset \mathcal{K}$.

The initial orbit that we start at is denoted by the closed initial set $\mathcal{I} \subset \mathcal{K}$. Note that the initial and the target orbit restrict only the position and the velocity, but allow the mass to take any admissible value within $[m_{\min}, m_{\max}]$.

Since we restrict all admissible states to lie within the set $\mathcal{K}$, which constrains the radius ρ , it is more efficient to recast our problem in spherical coordinates. Thus

we define a_ρ, a_θ, a_ψ as the spherical coordinate transformations of the acceleration terms, defined in Cartesian coordinates by

$$a_x := U_x(x, y, z) + \omega^2 x + 2\omega v_y,$$

$$a_y := U_y(x, y, z) + \omega^2 y - 2\omega v_x,$$

$$a_z := U_z(x, y, z).$$

The tangential velocity in the x-y plane, v_t , and its perpendicular counterpart, $v_\perp$, can then be defined as follows:

$$\begin{bmatrix} v_t \\ v_\perp \end{bmatrix} = \begin{bmatrix} \rho \dot{\theta} \sin \psi \\ \rho \dot{\psi} \end{bmatrix}, \quad (2.8)$$

$$\begin{bmatrix} a_t \\ a_\perp \end{bmatrix} = \begin{bmatrix} \sin \psi [a_\theta \rho + \dot{\theta} v_\rho] + \dot{\theta} v_\perp \cos \psi \\ v_\rho \dot{\psi} + a_\psi \rho \end{bmatrix}. \quad (2.9)$$

Then we can restate the system dynamics in spherical coordinates as

$$r = [\rho, \theta, \psi, v_\rho, v_t, v_\perp, \Delta m]^T \in \mathbb{R}^7, \quad (2.10)$$

$$f(r, u) = \begin{bmatrix} v_\rho \\ v_t \\ \rho \sin \psi \\ v_\perp \\ \frac{\rho}{T} \\ a_\rho + \frac{\rho}{m_0 + \Delta m} \cos \alpha \\ a_t + \frac{\rho}{m_0 + \Delta m} \sin \alpha \sin \delta \\ a_\perp + \frac{\rho}{m_0 + \Delta m} \sin \alpha \cos \delta \\ -\frac{1}{v_{\text{exhaust}}} \end{bmatrix}. \quad (2.11)$$

With a slight abuse of notation, we also redefine $\mathcal{I}$, $\mathcal{C}$, and $\mathcal{K}$ in spherical coordinates. Finally, we make the following assumptions on the spacecraft dynamics.

Assumption 2.1. *For every $r \in \mathcal{K}$ the set $\{f(r, u) : u \in \mathcal{U}\}$ is a compact convex subset of $\mathbb{R}^7$.*

Assumption 2.2. *$f : \mathbb{R}^7 \times \mathcal{U} \rightarrow \mathbb{R}^7$ is bounded and there exists an $L_f > 0$ such that for every $u_1, u_2 \in \mathcal{U}$,*

$$\|f(r_1, u_1) - f(r_2, u_2)\| \leq L_f \|r_1 - r_2\|.$$

Due to norm equivalence, the choice of norm is irrelevant and not further discussed.

Remark 2.1. *Electric thrusters often have a minimum power at which they may be turned on as well as a performance model that describes the thrust between the upper and lower bounds [50]. This commonly necessitates the inclusion of a nonzero minimum thrust. As in [95, 151], it can be shown that the inclusion of a nonzero minimum thrust and a convexification of the dynamics does not alter the optimal minimisers of the Hamiltonian in Section 2.4.2. This is because the dynamics are affine with respect to the thrust and as such the optimal thrust will be bang bang, equating to thrust arcs (constant thrust) and coast arcs (no thrust) [51]. It is therefore reasonable to not consider the minimum power constraints for medium-fidelity trajectory design and we assume Assumption 2.1 to hold.*

Using Assumptions 2.1 and 2.2, for any control policy $u(\cdot) \in \mathcal{U}_{ad}$, any initial state $r_0 \in \mathcal{K}$ and transfer time $t_f > 0$, the system admits a unique, absolutely continuous solution on $[-t_f, 0]$ (see [125]).

2.3 Problem Statement

2.3.1 Multi-objective optimal control problem

Having defined the system dynamics, we are now in a position to discuss how to find trajectories that start on an initial orbit, $\mathcal{I}$, and take the spacecraft to some final orbit, $\mathcal{C}$. Additionally, the objective is to keep the flight time and required propellant as small as possible. Thus, the multi-objective optimal control problem can be formulated as a minimisation problem whereby the first goal is to minimise the required propellant, Δm , and the second is to minimise the required time for the orbit change, i.e., the transfer time, denoted by t_f . The trajectory, $r(\cdot)$, which is the solution of Equation (2.10), belongs to the Sobolev space $\mathbb{W}^{1,1}(\mathbb{R}^7)$. The set of trajectory-control pairs on $[-t_f, 0]$ starting at r_0 with transfer time t_f is denoted as:

$$\Pi_{r_0, t_f} := \{(r(\cdot), u(\cdot)) \mid \dot{r}(t) = f(r(t), u(t)),$$

$$\forall t \in [-t_f, 0]; r(-t_f) = r_0\} \subset \mathbb{W}^{1,1}(\mathbb{R}^7) \times \mathcal{U}_{ad}.$$

Note that as in [39] we adopt the convention that 0 denotes the terminal time hence the transfer time t_f denotes the time duration. Under Assumption 2.1 and by Filippov's Theorem [87, p. 121], we can conclude, that Π_{r_0, t_f} is compact.

Remark 2.2. *For similar applications as the one discussed in this chapter, where Assumption 2.1 might not hold, we refer to [22] where convexification of the dynamics is considered in order to ensure that the set of absolutely continuous solutions of the problem is closed.*

The set of admissible (in the sense of satisfying the state constraints) trajectory-control pairs on $[-t_f, 0]$ starting at r_0 with transfer time t_f is denoted as:

$$\Pi_{r_0, t_f}^{\mathcal{K}, \mathcal{C}} := \{(r(\cdot), u(\cdot)) \in \Pi_{r_0, t_f} \mid r(t) \in \mathcal{K}, \quad \forall t \in [-t_f, 0]; r(0) \in \mathcal{C}\} \subset \Pi_{r_0, t_f}.$$

Finally, the set of admissible initial state and transfer time pairs is denoted as

$$\Gamma_{\mathcal{I}, \mathcal{C}} := \{(r_0, t_f) \in \mathcal{I} \times [0, +\infty) \mid \text{such that } \Pi_{r_0, t_f}^{\mathcal{K}, \mathcal{C}} \neq \emptyset\}.$$

For a given initial state $r_0 \in \mathbb{R}^7$ and transfer time $t_f \in [0, +\infty)$, we can define the cost functions as $J_1(r_0, t_f) := \Delta m$ and $J_2(r_0, t_f) := t_f$, where Δm is the 7-th element of the state vector r_0 . The 2-dimensional objective function $J : \mathbb{R}^7 \times [0, +\infty) \rightarrow \mathbb{R}^2$ can then be written as

$$J(r_0, t_f) := [J_1(r_0, t_f), J_2(r_0, t_f)]^T. \quad (2.12)$$

We are now in a position to formulate the multi-objective optimal control problem under study as

$$\begin{aligned} & \underset{(r_0, t_f) \in \mathcal{I} \times [0, \infty)}{\text{minimise}} && J(r_0, t_f) \\ & \text{subject to} && (r_0, t_f) \in \Gamma_{\mathcal{I}, \mathcal{C}} \end{aligned} \quad (2.13)$$

2.3.2 Pareto optimality

The solution of (2.13) in general does not consist of a single isolated point, but rather a set of optimal compromises between the objectives J_1 and J_2 [102].

Definition 2.1. A solution $(r_0, t_f) \in \Gamma_{\mathcal{I}, \mathcal{C}}$ is considered Pareto optimal if $\nexists (\hat{r}_0, \hat{t}_f) \in \Gamma_{\mathcal{I}, \mathcal{C}}$ such that $J(\hat{r}_0, \hat{t}_f) < J(r_0, t_f)$,

where a vector a is considered less than b (denoted $a < b$) if for every element a_i and b_i the relation $a_i < b_i$ holds. The relations $\leq, \geq, >$ are defined in an analogous way. Following Definition 2.1, a solution (r_0, t_f) is considered Pareto optimal if it is not possible to improve all its performance metrics $J_1(r_0, t_f), J_2(r_0, t_f)$ simultaneously. The set of Pareto optimal solutions is called the Pareto set $\mathcal{P}_S$, while its image is the Pareto front $\mathcal{P}_F$. Therefore, the solution of (2.13), i.e., the set of minimising (r_0, t_f) pairs, is the desired Pareto set, while the cost function corresponding to the minimising (r_0, t_f) pairs is the Pareto front. To allow mission designers to determine a compromise between minimising required propellant and transfer times, we wish to compute the Pareto front.

The constraint in (2.13) ensuring a solution (r_0, t_f) is feasible, makes solving (2.13) exceedingly difficult, as for each candidate solution it becomes necessary to check that a feasible trajectory exists. Furthermore, to obtain the Pareto front of the proposed multi-objective dynamic programming problems using conventional solution methods such as Normal Boundary Intersection or multi-objective evolutionary algorithms such as MOGA (Multi-Objective Gen. Alg.), NSGA, NSGA-II (Non-dominated Sorting Gen. Alg.), SPEA, SPEA2 (Strengthen Pareto Evolutionary Alg.), or NPGA (Niche Pareto Gen. Alg.) require the feasible region of constraints to be formulated using equality and inequality constraints [66, 113]. Therefore, to make our MOCP readily fit into the framework of conventional multi-objective optimisation problems, we will subsequently discuss how we can recast the constraint $(r_0, t_f) \in \Gamma_{\mathcal{I}, \mathcal{C}}$ as an inequality constraint.

2.4 Solution to multi-objective optimal control problems

To find an equivalent formulation for the constraint $(r_0, t_f) \in \Gamma_{\mathcal{I}, \mathcal{C}}$, let $g(r)$ and $\nu(r)$ be two Lipschitz functions (with Lipschitz constants L_g and L_ν , respectively)

chosen such that

$$g(r) \leq 0 \iff r \in \mathcal{K},$$

$$\nu(r) \leq 0 \iff r \in \mathcal{C}.$$

This can be achieved by choosing $g(r)$ and $\nu(r)$ as the signed distance to the set $\mathcal{K}$ and $\mathcal{C}$, respectively. Next, we consider the value function ω :

$$\omega(r_0, t_f) := \inf_{(r(\cdot), u(\cdot)) \in \Pi_{r_0, t_f}} \left\{ \nu(r(0)) \vee \max_{t \in [-t_f, 0]} g(r(t)) \right\}, \quad (2.14)$$

where $a \vee b$ denotes $\max(a, b)$. We are now in a position to use the value function to decide if, for a given initial state and transfer time, there exists a corresponding admissible trajectory. Thus we can introduce an equivalent formulation of (2.13).

Theorem 2.1. *The constrained MOCP, (2.13), is equivalent to*

$$\begin{aligned} & \underset{(r_0, t_f) \in \mathcal{I} \times [0, \infty)}{\text{minimise}} && J(r_0, t_f) \\ & \text{subject to} && \omega(r_0, t_f) \leq 0. \end{aligned} \quad (2.15)$$

Proof. We show that $(r_0, t_f) \in \Gamma_{\mathcal{I}, \mathcal{C}} \iff \omega(r_0, t_f) \leq 0$.

Case A: Consider $(r_0, t_f) \in \Gamma_{\mathcal{I}, \mathcal{C}}$. For the sake of contradiction assume that $\omega(r_0, t_f) > 0$. This then implies that for all $(r(\cdot), u(\cdot)) \in \Pi_{r_0, t_f}^{\mathcal{K}, \mathcal{C}}$ either $\nu(r(0)) > 0 \iff r(0) \notin \mathcal{C}$ or there exists $t \in [-t_f, 0]$ such that $g(r(t)) > 0 \iff r(t) \notin \mathcal{K}$. This contradicts the fact that $(r_0, t_f) \in \Gamma_{\mathcal{I}, \mathcal{C}}$ establishing that $(r_0, t_f) \in \Gamma_{\mathcal{I}, \mathcal{C}}$ implies $\omega(r_0, t_f) \leq 0$.

Case B: Consider $(r_0, t_f) \in \mathcal{I} \times [0, \infty)$, such that $\omega(r_0, t_f) \leq 0$. Under Assumption 2.1, applying Weierstrass' Theorem on the existence of minima for compact sets [5, 14], we can conclude that the infimum over Π_{r_0, t_f} exists, and thus, $\omega(r_0, t_f) \leq 0$ implies the existence of a trajectory-control pair $(r(\cdot), u(\cdot)) \in \Pi_{r_0, t_f}$, such that for all $t \in [-t_f, 0]$, $g(r(t)) \leq 0$ and $\nu(r(0)) \leq 0$. By definition of the function g and ν , we thus have $r(t) \in \mathcal{K}$ for all $t \in [-t_f, 0]$ and $r(0) \in \mathcal{C}$, which in turns implies $(r(\cdot), u(\cdot)) \in \Pi_{r_0, t_f}^{\mathcal{K}, \mathcal{C}}$. Therefore,

$$\omega(r_0, t_f) \leq 0 \Rightarrow (r_0, t_f) \in \Gamma_{\mathcal{I}, \mathcal{C}},$$

thus concluding the proof. $\square$

Theorem 2.1 implies that the Pareto front can be computed from the solution of (2.15).

2.4.1 Extension to problems in Bolza form

We will now generalise our approach to problems where the objective functions do not rely only on the initial state and are written in Bolza form. To achieve this we need to introduce auxiliary states. As in [5, 46], we show how problems in Bolza form are reformulated into Mayer form, and then show how problems in Mayer form are solved in a similar fashion as in Section 2.3. Consider the p -dimensional objective function defined as:

$$J^{\text{Bolza}}(r(\cdot), u(\cdot), t_f) := J^t(r(0)) + \int_{-t_f}^0 J^r(r(s), u(s)) ds, \quad (2.16)$$

where J^t denotes the terminal cost and J^r denotes the running cost. We impose the following assumptions, as in [46].

Assumption 2.3. J^t is locally Lipschitz continuous on $\mathbb{R}^7$ with Lipschitz constant $L^t(R)$ for every neighbourhood $R \subset \mathbb{R}^7$.

Assumption 2.4. J^r is continuous on $\mathbb{R}^7 \times \mathcal{U}$. Moreover, J^r is locally Lipschitz continuous on the first variable with Lipschitz constant $L^r(R)$ for every neighbourhood $R \subset \mathbb{R}^7$.

Remark 2.3. To ease notation, we omit the dependants on the neighbourhood for the Lipschitz constants and instead assume the existence of a global Lipschitz constant L^t and L^r , respectively.

In light of the objective function (2.16) we revisit Definition 2.1 to introduce the concept of Pareto optimality for problems in Bolza form.

Definition 2.2. Let $(r_0, t_f), (\hat{r}_0, \hat{t}_f) \in \Gamma_{\mathcal{L}, \mathcal{C}}$. We consider the trajectory-control pairs $(r(\cdot), u(\cdot)) \in \Pi_{r_0, t_f}^{\mathcal{K}, \mathcal{C}}$ and $(\hat{r}(\cdot), \hat{u}(\cdot)) \in \Pi_{\hat{r}_0, \hat{t}_f}^{\mathcal{K}, \mathcal{C}}$

1. The trajectory $r(\cdot)$ strictly dominates $\hat{r}(\cdot)$ if

$$J^{\text{Bolza}}(r(\cdot), u(\cdot), t_f) < J^{\text{Bolza}}(\hat{r}(\cdot), \hat{u}(\cdot), \hat{t}_f). \quad (2.17)$$

2. The trajectory $r(\cdot)$ is considered Pareto optimal if $\forall(\hat{r}_0, \hat{t}_f) \in \Gamma_{\mathcal{I}, \mathcal{C}}$,
 $\nexists(\hat{r}(\cdot), \hat{u}(\cdot)) \in \Pi_{\hat{r}_0, \hat{t}_f}^{\mathcal{K}, \mathcal{C}}$ such that $J^{\text{Bolza}}(\hat{r}(\cdot), \hat{u}(\cdot), \hat{t}_f) < J^{\text{Bolza}}(r(\cdot), u(\cdot), t_f)$.

Next, we define an auxiliary state $z_0 \in \mathbb{R}^p$ as:

$$\begin{cases} \dot{z}(s) = -J^r(r(s), u(s)), & \forall s \in [-t_f, 0] \\ z(0) = z_0, \end{cases} \quad (2.18)$$

where z_0 becomes an optimisation parameter and $z \in \mathbb{W}^{1,1}(\mathbb{R}^p)$. The auxiliary state captures the cumulative running cost and thus is treated as an additional state. In the same manner we previously ensured a trajectory, $r(\cdot)$, stayed within the set $\mathcal{K}$, we bound $z(\cdot)$ and ensure that the integrated running cost, added with the terminal cost, stays below some value z_0 . To capture all possible trajectories, we introduce the set

$$\begin{aligned} \mathcal{Z}_{r_0, t_f, z_0} &:= \{(r(\cdot), u(\cdot), z(\cdot)) \mid (r(\cdot), u(\cdot)) \in \Pi_{r_0, t_f}; \\ &\quad \dot{z}(s) = -J^r(r(s), u(s)), \forall s \in [-t_f, 0]; z(0) = z_0\}, \end{aligned} \quad (2.19)$$

and make the following assumption.

Assumption 2.5. For every $r \in \mathbb{R}^7$, the set

$$\left\{ \begin{bmatrix} f(r, u) \\ -J^r(r, u) \end{bmatrix} : u \in \mathcal{U}, \right\}$$

is a compact convex subset of $\mathbb{R}^7 \times \mathbb{R}^p$.

Under Assumptions 2.2, 2.3, 2.4 and 2.5, by Filippov's Theorem [87, p. 119], the problem (2.21) admits an optimal solution, which implies the existence of an admissible $r(\cdot)$ and $z(\cdot)$ [46]. This yields the following relationship due to (2.18)

$$\bigvee_i [J_i^t(r(0)) - z_i(-t_f)] = \bigvee_i \left[J_i^t(r(0)) + \int_{-t_f}^0 J_i^r(r(s), u(s)) ds - z_{0,i} \right], \quad (2.20)$$

where $\bigvee_i x_i$ denotes the maximum element of the vector x . We now introduce the auxiliary value function ϑ :

$$\begin{aligned} \vartheta(r_0, t_f, z_0) &:= \inf_{(r(\cdot), u(\cdot), z(\cdot)) \in \mathcal{Z}_{r_0, t_f, z_0}} \left\{ \bigvee_i [J_i^t(r(0)) - z_i(-t_f)] \right. \\ &\quad \left. \bigvee \nu(r(0)) \bigvee \max_{s \in [-t_f, 0]} g(r(s)) \right\}. \end{aligned} \quad (2.21)$$

As with ω , the term $\max_{s \in [-t_f, 0]} g(r(s))$ and $\nu(r(0))$ ensures that any trajectory $r(\cdot)$ remains in $\mathcal{K}$ and terminates in $\mathcal{C}$. The additional term $J_i^t(r(0)) - z_i(-t_f)$ ensures that the integrated running cost, J^r , combined with the terminal cost, J^t , never grows larger than z_0 . Thus, in addition to ensuring that $(r(\cdot), u(\cdot))$ are admissible trajectory control pairs, the sub-zero level set of ϑ bounds the terminal and integrated running cost. Therefore,

$$\vartheta(r_0, t_f, z_0) \leq 0 \iff [\exists(r(\cdot), u(\cdot)) \in \Pi_{r_0, t_f}^{\mathcal{K}, \mathcal{C}} | J^{\text{Bolza}}(r(\cdot), u(\cdot), t_f) \leq z_0]. \quad (2.22)$$

We are now in a position to introduce the generalised version of (2.13) for objective functions in Bolza form:

$$\begin{aligned} & \underset{(r_0, t_f, z_0) \in \mathcal{I} \times [0, \infty) \times \mathbb{R}^p}{\text{minimise}} && z_0 \\ & \text{subject to} && \vartheta(r_0, t_f, z_0) \leq 0, \end{aligned} \quad (2.23)$$

where z_0 represents an upper bound for the term $J^{\text{Bolza}}(r(\cdot), u(\cdot), t_f)$, without explicit knowledge of $r(\cdot)$ or $u(\cdot)$.

Unlike the MOCP (2.15), the cost function of (2.23) does not depend on t_f explicitly. By introducing a change of variables, we will incorporate the unknown terminal time, t_f , within the framework of a fixed finite horizon problem, allowing us to reduce the size of the constraint ϑ . Let us introduce

$$t_{t_f}(s) := t_0 + s(t_f + t_0) \quad \text{for } s \in [-1, 0].$$

The new dynamics of the fixed final horizon problem (where the final horizon is 1 and $t_0 := 0$) are subsequently defined as:

$$\tilde{f}(r, u, t_f) = (t_f - t_0)f(r, u) \Rightarrow \begin{cases} \dot{\tilde{r}}(s) = \tilde{f}(\tilde{r}(s), \tilde{u}(s), \iota(s)) & s \in [-\kappa, 0] \\ i(s) = 0 & s \in [-\kappa, 0] \\ \tilde{r}(-\kappa) = r_0 \\ \iota(-\kappa) = t_f \end{cases}, \quad (2.24)$$

where κ is chosen from $[0, 1]$ and r_0 is an initial state. Analogously, the augmented set of trajectory control pairs is defined as

$$\tilde{\Pi}_{r_0, t_f} := \{(\tilde{r}(\cdot), \tilde{u}(\cdot)) : \dot{\tilde{r}}(s) = \tilde{f}(\tilde{r}(s), \tilde{u}(s), \iota(s)) \forall s \in [-1, 0];$$

$$\tilde{r}(-1) = r_0; \iota(-1) = t_f\}.$$

$$\tilde{\Pi}_{r_0, t_f}^{\mathcal{K}, \mathcal{C}} := \{(\tilde{r}(\cdot), \tilde{u}(\cdot)) \in \tilde{\Pi}_{r_0, t_f} : \tilde{r}(s) \in \mathcal{K}, \quad \forall s \in [-1, 0]; r(0) \in \mathcal{C}\} \subset \tilde{\Pi}_{r_0, t_f}.$$

In light of the new dynamics, we will redefine the objective function (2.16) for the augmented dynamics as

$$J^{\text{Bolza}}(\tilde{r}(\cdot), \tilde{u}(\cdot), t_f) := J^t(\tilde{r}(0)) + \int_{-1}^0 J^r(\tilde{r}(s), \tilde{u}(s), t_f) ds, \quad (2.25)$$

Similarly, we define the augmented auxiliary state

$$\begin{cases} \dot{\tilde{z}}(s) = -J^r(\tilde{r}(s), \tilde{u}(s), t_f), & \forall s \in [-1, 0] \\ \tilde{z}(0) = z_0, \end{cases} \quad (2.26)$$

which in turn yields the set

$$\begin{aligned} \tilde{\mathcal{Z}}_{r_0, t_f, z_0} &:= \{(\tilde{r}(\cdot), \tilde{u}(\cdot), \tilde{z}(\cdot)) \mid (\tilde{r}(\cdot), \tilde{u}(\cdot)) \in \tilde{\Pi}_{r_0, t_f}; \\ &\quad \dot{\tilde{z}}(s) = -J^r(\tilde{r}(s), \tilde{u}(s), t_f) \quad \forall s \in [-1, 0]; \tilde{z}(0) = z_0\}, \end{aligned} \quad (2.27)$$

Using the augmented dynamics we can define an augmented value function, $\tilde{\omega}$:

$$\begin{aligned} \tilde{\omega}(\kappa, r_0, z_0, t_f) &:= \inf_{(\tilde{r}(\cdot), \tilde{u}(\cdot), \tilde{z}(\cdot)) \in \tilde{\mathcal{Z}}_{r_0, t_f, z_0}} \left\{ \bigvee_i \left[J_i^t(\tilde{r}(0)) - \tilde{z}_i(-\kappa) \right] \right. \\ &\quad \left. \bigvee \nu(\tilde{r}(0)) \bigvee_{s \in [-\kappa, 0]} g(\tilde{r}(s)) \right\}, \end{aligned} \quad (2.28)$$

where we make use of the relationship (2.20). Finally, we utilise the augmented value function to define

$$\tilde{\vartheta}(r_0, z_0) := \min_{t_f} \tilde{\omega}(1, r_0, z_0, t_f), \quad (2.29)$$

which can be used as a direct substitution within the MOCP (2.23). For a potential speed up of the MOCP, we will next show that the inequality constraint in (2.23) can in fact be replaced by an equality constraint. To this end, we will start by introducing the Utopian point $J_i^*(\mathcal{I}, \mathcal{C})$, which is the lower bound of J^{Bolza} , defined elementwise as

$$J_i^*(\mathcal{I}, \mathcal{C}) := \inf_{\{(\tilde{r}(\cdot), \tilde{u}(\cdot), t_f) \mid \exists (r_0, t_f) \in \Gamma_{\mathcal{I}, \mathcal{C}}; (\tilde{r}(\cdot), \tilde{u}(\cdot)) \in \tilde{\Pi}_{r_0, t_f}^{\mathcal{K}, \mathcal{C}}\}} J_i^{\text{Bolza}}(\tilde{r}(\cdot), \tilde{u}(\cdot), t_f). \quad (2.30)$$

Furthermore, we will introduce the following Lemma.

Lemma 2.1. *Let $r_0 \in \mathbb{R}^7$. Then for all $z_0, \hat{z}_0 \in \mathbb{R}^p$, $z_0 \leq \hat{z}_0$, it holds that*

$$z_0 \leq \hat{z}_0 \Rightarrow \tilde{\vartheta}(r_0, z_0) \geq \tilde{\vartheta}(r_0, \hat{z}_0).$$

Proof. Following the definition of $\tilde{\vartheta}$, there exists a $t_f, \hat{t}_f \in [0, +\infty)$ such that: $\tilde{\vartheta}(r_0, z_0) = \tilde{\omega}(1, r_0, z_0, t_f)$ and $\tilde{\vartheta}(r_0, \hat{z}_0) = \tilde{\omega}(1, r_0, \hat{z}_0, \hat{t}_f)$. Subsequently, we have:

$$\tilde{\vartheta}(r_0, \hat{z}_0) = \tilde{\omega}(1, r_0, \hat{z}_0, \hat{t}_f) \leq \tilde{\omega}(1, r_0, \hat{z}_0, t_f) \leq \tilde{\omega}(1, r_0, z_0, t_f) = \tilde{\vartheta}(r_0, z_0).$$

□

Next, we will consider a parameterisation of the Pareto front utilising the Value function $\tilde{\vartheta}$.

Theorem 2.2. *Let $z_i^* := J_i^*(\mathcal{I}, \mathcal{C})$ be the Utopian values associated with the initial orbit $\mathcal{I}$ and target orbit $\mathcal{C}$ and let us define $z^* := \begin{bmatrix} z_1^* \\ \vdots \\ z_p^* \end{bmatrix}$ and $\mu := \begin{bmatrix} \mu_1 \\ \vdots \\ \mu_p \end{bmatrix} \in \mathbb{R}_{\geq 0}^p$. We consider the extended function $\Theta_{\mathcal{I}, \mathcal{C}} : [0, 1]^p \rightarrow [0, \infty]$ defined as follows:*

$$\Theta_{\mathcal{I}, \mathcal{C}}(\mu) := \inf \left\{ \tau \geq 0 \mid \exists r_0 \in \mathcal{I} : \tilde{\vartheta}(r_0, z^* + \mu\tau) \leq 0 \right\}.$$

Let us define the $(p-1)$ -dimensional manifold:

$$\Sigma_{\mathcal{I}, \mathcal{C}} := \left\{ \begin{bmatrix} z_1^* \\ \vdots \\ z_p^* \end{bmatrix} + \mu \cdot \Theta_{\mathcal{I}, \mathcal{C}}(\mu) \mid \mu := \begin{bmatrix} \mu_1 \\ \vdots \\ \mu_p \end{bmatrix} \in [0, 1]^p \text{ with } \sum_{k=1}^p \mu_k = 1 \right\}.$$

Any trajectory reconstructed from $\Sigma_{\mathcal{I}, \mathcal{C}}$, is guaranteed to be Pareto optimal.

Proof. Let $(r_0, t_f, z_0) \in \Gamma_{\mathcal{I}, \mathcal{C}} \times \Sigma_{\mathcal{I}, \mathcal{C}}$ with $(\tilde{r}(\cdot), \tilde{u}(\cdot)) \in \tilde{\Pi}_{r_0, t_f}^{\mathcal{K}, \mathcal{C}}$ such that

$$J^{\text{Bolza}}(\tilde{r}(\cdot), \tilde{u}(\cdot), t_f) \leq z_0. \quad (2.31)$$

By definition of $\Sigma_{\mathcal{I}, \mathcal{C}}$, there exists a $\mu \in [0, 1]^p$ with $\sum_{k=1}^p \mu_k = 1$ such that $z_0 = z^* + \mu\Theta_{\mathcal{I}, \mathcal{C}}(\mu)$.

First note, that if $\exists i \in \{1, \dots, p\} : \mu_i = 0$, then $z_{0,i} = z_i^*$ and $\nexists(\hat{r}(\cdot), \hat{u}(\cdot)) \in \tilde{\Pi}_{r_0, t_f}^{\mathcal{K}, \mathcal{C}} : J^{\text{Bolza}}(\hat{r}(\cdot), \hat{u}(\cdot), t_f) < J^{\text{Bolza}}(\tilde{r}(\cdot), \tilde{u}(\cdot), t_f)$, as the Utopian point by definition cannot be dominated. Therefore, we will assume that $\mu_i \neq 0$ for all $i \in \{1, \dots, p\}$ and show by contradiction that $\nexists(\hat{r}_0, \hat{t}_f) \in \Gamma_{\mathcal{I}, \mathcal{C}}$ and $(\hat{r}(\cdot), \hat{u}(\cdot)) \in \tilde{\Pi}_{\hat{r}_0, \hat{t}_f}^{\mathcal{K}, \mathcal{C}}$ that can strictly dominate $(\tilde{r}(\cdot), \tilde{u}(\cdot))$.

Let us assume there exists a $(\hat{r}_0, \hat{t}_f) \in \Gamma_{\mathcal{I}, \mathcal{C}}$ and $(\hat{r}(\cdot), \hat{u}(\cdot)) \in \tilde{\Pi}_{\hat{r}_0, \hat{t}_f}^{\mathcal{K}, \mathcal{C}}$ that strictly dominates $(\tilde{r}(\cdot), \tilde{u}(\cdot))$, i.e., $\hat{z}_0 := J^{\text{Bolza}}(\hat{r}(\cdot), \hat{u}(\cdot), \hat{t}_f) < J^{\text{Bolza}}(\tilde{r}(\cdot), \tilde{u}(\cdot), t_f) \leq z_0$.

Therefore, $\tilde{\vartheta}(\hat{r}_0, \hat{z}_0) \leq 0$ and there exists a unique pair $(\hat{\mu}, \hat{\tau}) \in (0, 1]^p \times \mathbb{R}_{>0}^p$ such that $\sum_{k=1}^p \hat{\mu}_k = 1$ and $\hat{z}_0 = z^* + \hat{\mu}\hat{\tau} < z^* + \mu\Theta_{\mathcal{I},\mathcal{C}}(\mu) \leq z_0$.

If we assume that $\hat{\mu} = \mu$, then consequently $\hat{z}_0 = z^* + \mu\hat{\tau}$ which implies that $\hat{\tau} < \Theta_{\mathcal{I},\mathcal{C}}(\mu)$. However, since $\tilde{\vartheta}(\hat{r}_0, \hat{z}_0) \leq 0$ it must hold that $\hat{\tau} \geq \Theta_{\mathcal{I},\mathcal{C}}(\mu)$, a clear contradiction.

Alternatively, we consider the case $\hat{\mu} \neq \mu$. In this case we can define $\frac{\hat{\mu}_{i_{\max}}}{\mu_{i_{\max}}} := \bigvee_i \frac{\hat{\mu}_i}{\mu_i} > 1$ for $i_{\max} \in \{1, \dots, p\}$. Introducing $\hat{\tau}_0 := \frac{\hat{\mu}_{i_{\max}}}{\mu_{i_{\max}}} \hat{\tau} > \hat{\tau}$ yields $\hat{z}_{0,i_{\max}} = z_{i_{\max}}^* + \hat{\mu}_{i_{\max}} \hat{\tau} = z_{i_{\max}}^* + \mu_{i_{\max}} \hat{\tau}_0$. By definition of strict dominance, $\hat{z}_0 < z_0 \iff \forall i \in \{1, \dots, p\} \hat{z}_{0,i} < z_{0,i}$ and therefore $\hat{z}_{0,i_{\max}} < z_{0,i_{\max}} = z_{i_{\max}}^* + \mu_{i_{\max}} \Theta_{\mathcal{I},\mathcal{C}}(\mu)$. This implies that $\hat{\tau}_0 < \Theta_{\mathcal{I},\mathcal{C}}(\mu)$ (and thus that $\hat{\tau} < \Theta_{\mathcal{I},\mathcal{C}}(\mu)$), which is a contradiction since for $\tilde{\vartheta}(\hat{r}_0, \hat{z}_0) \leq 0$ it must hold that $\hat{\tau} \geq \Theta_{\mathcal{I},\mathcal{C}}(\mu)$. In conclusion, there is no admissible pair $(\hat{r}(\cdot), \hat{u}(\cdot))$ that strictly dominates $(\tilde{r}(\cdot), \tilde{u}(\cdot))$ and thus every trajectory reconstructed from $\Sigma_{\mathcal{I},\mathcal{C}}$ is Pareto optimal. $\square$

For the construction of the $\Sigma_{\mathcal{I},\mathcal{C}}$ we introduce the following theorem.

Theorem 2.3. $\Sigma_{\mathcal{I},\mathcal{C}}$ is a subset of the zero level set of the value function $\tilde{\vartheta}$:

$$\Sigma_{\mathcal{I},\mathcal{C}} \subseteq \left\{ z_0 \in \mathbb{R}^p \mid \exists r_0 \in \mathcal{I} : \tilde{\vartheta}(r_0, z_0) = 0 \right\}.$$

Proof. Let us assume that $\Sigma_{\mathcal{I},\mathcal{C}} \not\subseteq \left\{ z_0 \in \mathbb{R}^p \mid \exists r_0 \in \mathcal{I} : \tilde{\vartheta}(r_0, z_0) = 0 \right\}$ and therefore there exists a $r_0 \in \mathcal{I}$ and $z_0 = z^* + \mu\Theta_{\mathcal{I},\mathcal{C}}(\mu) \in \Sigma_{\mathcal{I},\mathcal{C}}$, such that $\tilde{\vartheta}(r_0, z_0) < 0$. The case $\tilde{\vartheta}(r_0, z_0) > 0$ can be excluded, as it follows from the definition of $\Theta_{\mathcal{I},\mathcal{C}}$, that $\forall z_0 \in \Sigma_{\mathcal{I},\mathcal{C}}, \tilde{\vartheta}(r_0, z_0) \leq 0$. $\tilde{\vartheta}(r_0, z_0) < 0$ implies that there exists a $(r_0, t_f) \in \Gamma_{\mathcal{I},\mathcal{C}}$ and $(\tilde{r}(\cdot), \tilde{u}(\cdot)) \in \tilde{\Pi}_{r_0, t_f}^{\mathcal{K}, \mathcal{C}}$ such that

$$\begin{aligned} \tilde{\vartheta}(r_0, z^* + \mu\Theta_{\mathcal{I},\mathcal{C}}(\mu)) &= \left[\bigvee_i J_i^{\text{Bolza}}(\tilde{r}(\cdot), \tilde{u}(\cdot), t_f) - z_i^* - \mu_i \Theta_{\mathcal{I},\mathcal{C}}(\mu) \right. \\ &\quad \left. \bigvee \nu(\tilde{r}(0)) \bigvee_{s \in [-1, 0]} g(\tilde{r}(s)) \right] < 0 \\ &\Rightarrow \bigvee_i J_i^{\text{Bolza}}(\tilde{r}(\cdot), \tilde{u}(\cdot), t_f) - z_i^* - \mu_i \Theta_{\mathcal{I},\mathcal{C}}(\mu) < 0. \end{aligned}$$

From the continuity of the value function, we can conclude that this implies the existence of a $\tau < \Theta_{\mathcal{I},\mathcal{C}}(\mu)$, such that

$$\bigvee_i J_i^{\text{Bolza}}(\tilde{r}(\cdot), \tilde{u}(\cdot), t_f) - z_i^* - \mu_i \Theta_{\mathcal{I},\mathcal{C}}(\mu) < \bigvee_i J_i^{\text{Bolza}}(\tilde{r}(\cdot), \tilde{u}(\cdot), t_f) - z_i^* - \mu_i \tau \leq 0.$$

However, such a τ is contradictory to the definition $\Theta_{\mathcal{I},\mathcal{C}}(\mu)$ and subsequently there is no z_0 in $\Sigma_{\mathcal{I},\mathcal{C}}$ such that $\tilde{\vartheta}(r_0, z_0) < 0$ and therefore

$$\Sigma_{\mathcal{I},\mathcal{C}} \subseteq \left\{ z_0 \in \mathbb{R}^p \mid \exists r_0 \in \mathcal{I} : \tilde{\vartheta}(r_0, z_0) = 0 \right\}.$$

□

Thus it immediately follows from Theorem 2.3 that the inequality in the MOCP (2.23) can be exchanged with an equality constraint yielding

$$\begin{aligned} & \underset{(r_0, z_0) \in \mathcal{I} \times \mathbb{R}^p}{\text{minimise}} && z_0 \\ & \text{subject to} && \tilde{\vartheta}(r_0, z_0) = 0. \end{aligned} \tag{2.32}$$

It is worth pointing out that while the constraint of the MOCP (2.32) is of a lower dimension compared to problem (2.23), the computational cost of the reachability problem will increase, since we include an additional auxiliary state to bound t_f . The number of states and objectives will thus determine which formulation ((2.32) or (2.23)) is the most computationally efficient approach.

2.4.2 Value function computation

We will next consider the construction of the value functions ω , ϑ and $\tilde{\vartheta}$. We will begin with ω and introduce the Hamiltonian $H : \mathbb{R}^7 \times \mathbb{R}^7 \rightarrow \mathbb{R}$,

$$H(r, q) := \min_{u \in \mathcal{U}} \left(q^T f(r, u) \right),^1 \tag{2.33}$$

where $q \in \mathbb{R}^7$ is the costate vector. It then follows that the value function ω is the unique continuous viscosity solution of the following quasi-variational inequality

$$\begin{cases} 0 = \max \{ g(r) - \omega(r, t), \partial_t \omega + H(r, \nabla_r \omega) \} \\ \quad \text{for all } t \in [0, \infty), r \in \mathbb{R}^7, \\ \omega(r, 0) = \left(\nu(r) \bigvee g(r) \right) \quad \text{for all } r \in \mathbb{R}^7. \end{cases} \tag{2.34}$$

Since the Dynamic Programming Principle [14] holds for $h \in [0, t_f]$, $(r_0, t_f) \in \mathcal{K} \times \mathbb{R}$ with $t_f \geq 0$:

$$\omega(r_0, t_f) = \inf_{(r(\cdot), u(\cdot)) \in \Pi_{r_0, t_f}} \left\{ \omega(r(h - t_f), t_f - h) \bigvee \max_{s \in [-t_f, h - t_f]} g(r(s)) \right\},$$

¹Note that a change of sign in the definition of the Hamiltonian was omitted in [147].

the derivation of (2.34) follows standard arguments for viscosity solutions, as shown in [5, 97]. Note that the infimum should be understood to be over the restriction of Π_{r_0, t_f} over $[-t_f, h - t_f]$.

As in [23, 97], a consequence following from (2.34) is the Lipschitz continuity of the value function.

Proposition 2.1. *The value function ω is Lipschitz continuous.*

The proof is similar to the more general proof of Proposition 2.2 that is introduced in the sequel. Proposition 2.1 allows us to make statements about the discrete-continuous error estimate, for which we refer to Theorem 5 in [23], as well as the convergence of the value function, for which we refer to Proposition 6 in [63].

We next consider the computation of the Hamiltonian. As the PDE is solved backwards in time using a finite difference scheme, the Hamiltonian is given by $H(r, q) := -\min_{u \in \mathcal{U}} (q^T f(r, u))$, which admits an explicit form. Consider the term

$$C(r, q) := q_1 v_\rho + q_2 v_t + q_3 v_\perp + q_4 a_\rho + q_5 a_t + q_6 a_\perp.$$

Then we can write the Hamiltonian as

$$H(r, q) := -\min_{u \in \mathcal{U}} \left(\frac{T}{m_0 + \Delta m} (q_4 \cos \alpha \right. \quad (2.35)$$

$$\left. + \sin \alpha (q_5 \sin \delta + q_6 \cos \delta) \right) - q_7 \frac{T}{v_{\text{exhaust}}} \Big) - C(r, q). \quad (2.36)$$

As T is always positive, the thrust angles can be optimised separately; see the Appendix for more details. Furthermore, the derivation of α^* and δ^* follows a similar procedure as in [22] and is listed in the Appendix. After applying the optimal thrust angles, the Hamiltonian becomes affine in T and we are able to find the optimal thrust magnitude. In particular, $H(r, q)$ becomes

$$H(r, q) = -\min_{T \in [0, T_{\max}]} \left(-\frac{T}{m_0 + \Delta m} \sqrt{q_4^2 + q_5^2 + q_6^2} - q_7 \frac{T}{v_{\text{exhaust}}} \right) - C(r, q), \quad (2.37)$$

$$\Rightarrow T^* := \begin{cases} T_{\max} & \text{if } \frac{q_7}{v_{\text{exhaust}}} + \frac{\sqrt{q_4^2 + q_5^2 + q_6^2}}{m_0 + \Delta m} \geq 0 \\ 0 & \text{otherwise} \end{cases} \quad (2.38)$$

Remark 2.4. For the optimal control computation during the trajectory calculation, we have numerically investigated the occurrence of singular arcs and found no instances in which $\frac{\sqrt{q_4^2 + q_5^2 + q_6^2}}{m_0 + \Delta m} = -\frac{q_7}{v_{\text{exhaust}}}$ over an extended interval, thus we do not further consider the singular control case.

Finally, applying T^* and rewriting the minimum as the maximum of the negation of the associated function, the Hamiltonian takes the following analytic form

$$H(r, q) = -C(r, q) + \max \left(q_7 \frac{T_{\max}}{v_{\text{exhaust}}} + \frac{T_{\max}}{m_0 + \Delta m} \sqrt{q_4^2 + q_5^2 + q_6^2}, 0 \right).$$

Following similar argumentation, we now consider the generalised value function which can again be obtained as the unique continuous viscosity solution of a quasi-variational inequality

$$\begin{cases} \max \{g(r) - \vartheta(r, t, z), \partial_t \vartheta + H(r, \nabla_r \vartheta, \nabla_z \vartheta)\} = 0 \\ \quad \text{for all } t \in [0, \infty), r \in \mathbb{R}^7, z \in \mathbb{R}^p, \\ \vartheta(r, 0, z) = \bigvee_i [J_i^t(r) - z_i] \bigvee \nu(r) \bigvee g(r) \\ \quad \text{for all } r \in \mathbb{R}^7, z \in \mathbb{R}^p \end{cases} \quad (2.39)$$

where the Hamiltonian is defined as

$$H(r, q_r, q_z) := \min_{u \in \mathcal{U}} \left(q_r^T f(r, u) - q_z^T J^r(r, u) \right).$$

Proposition 2.2. The value function ϑ is Lipschitz continuous.

The proof can be found in the Appendix. Proposition 2.2 can be used to show that a numerical solution of (2.39) (in the viscosity sense) can always be determined.

For completeness, the augmented value function $\tilde{\omega}$ is the unique continuous viscosity solution of the quasi-variational inequality

$$\begin{cases} \max \{g(r) - \tilde{\omega}(\kappa, r, z, t_f), \partial_\kappa \tilde{\omega} + (t_f - t_0)H(r, \nabla_r \tilde{\omega}, \nabla_z \tilde{\omega})\} = 0 \\ \quad \text{for all } \kappa \in [0, 1], t_f \in [0, \infty), r \in \mathbb{R}^7, z \in \mathbb{R}^p, \\ \tilde{\omega}(0, r, z, t_f) = \bigvee_i [J_i^t(r) - z_i] \bigvee \nu(r) \bigvee g(r) \\ \quad \text{for all } t_f \in [0, \infty), r \in \mathbb{R}^7, z \in \mathbb{R}^p \end{cases} \quad (2.40)$$

2.5 Numerical Approximation and Results

We will now discuss how the value functions can be obtained numerically, prior to discussing how the spacecraft trajectory design problem is solved. Following Proposition 2.1, a numerical solution to (2.39) can be found. To this end, we employ the Level Set Methods toolbox of [103]. For the computation of ω , we use a Lax-Friedrich Hamiltonian

$$\mathcal{H}(r, p^-, p^+) = H(r, \frac{p^- + p^+}{2}) - \sum_{k=1}^7 \frac{\alpha_k}{2} (p^+ - p^-), \quad (2.41)$$

where p^+ and p^- are the right and left derivatives computed using an appropriate fifth-order weighted essentially non-oscillatory (WENO) scheme [110]. The Lax-Friedrich Hamiltonian consists of an analytic expression of the Hamiltonian (derived previously), as well as a dissipation term, which is scaled by the dissipation coefficients α_k . The dissipation coefficients α_k needs to satisfy

$$\alpha_k \geq \left| \frac{\partial H}{\partial q_k} \right|. \quad (2.42)$$

Since we need α_k to serve as an upper bound, we consider the control input that maximises the Hamiltonian:

$$\left| \frac{\partial H}{\partial q_k} \right| = \left| -\frac{\partial C(r, q)}{\partial q_k} + \frac{\partial}{\partial q_k} \max \left(q_7 \frac{T_{\max}}{v_{\text{exhaust}}} + \frac{T_{\max}}{m_0 + \Delta m} \sqrt{q_4^2 + q_5^2 + q_6^2}, 0 \right) \right| \quad (2.43)$$

$$\alpha_k = \begin{cases} |v_\rho| & k = 1 \\ \left| \frac{v_t}{\rho \sin \psi} \right| & k = 2 \\ \left| \frac{v_\perp}{\rho} \right| & k = 3 \\ \left| a_\rho - \max \left(0, \frac{T_{\max} q_3}{(m_0 + \Delta m) \sqrt{q_3^2 + q_4^2 + q_5^2}} \right) \right| & k = 4 \\ \left| a_t - \max \left(0, \frac{T_{\max} q_4}{(m_0 + \Delta m) \sqrt{q_3^2 + q_4^2 + q_5^2}} \right) \right| & k = 5 \\ \left| a_\perp - \max \left(0, \frac{T_{\max} q_5}{(m_0 + \Delta m) \sqrt{q_3^2 + q_4^2 + q_5^2}} \right) \right| & k = 6 \\ \frac{T_{\max}}{v_{\text{exhaust}}} & k = 7, \end{cases} \quad (2.44)$$

For a further discussion of the Lax-Friedrich Hamiltonian and WENO scheme, we refer to [110], while for a discussion of the convergence of ω and the derivation of a necessary Courant-Friedrichs-Lewy condition, we refer to [23, 75, 103].

Scale	Values
Distance	Initial radius ρ_0
Velocity	2 m/s
Time	ρ_0/v_0
Mass	1 kg
Force	Maximum thrust $T_{\max}$

Table 2.1: Normalisation values

2.5.1 Implementation

To illustrate the theoretical results of the previous sections, we consider a spacecraft on an initial near circular orbit around asteroid Castalia 4769. The goal is to compute an efficient transfer trajectory that raises the spacecraft to a stable orbit at an altitude of 6117.5 m above the asteroid. For the derivation of a stable orbit around Castalia 4769 we refer to [85] and references therein. The gravity of Castalia 4769 was modelled by means of a spherical harmonic expansion as discussed in [78, 126, 127]. Even though the proposed theoretical framework allows us to tackle problems of any state dimension, the available numerical tools and computational power limit us to only study the lower-dimensional planar case of the application. We, therefore, omit the states ψ and $v_{\perp}$ to consider only

$$r = [\rho, \theta, v_{\rho}, v_t, \Delta m]^T \in \mathbb{R}^5. \quad (2.45)$$

To avoid ill-conditioning when solving the HJB equation, the state vector is normalised using the constants introduced in Table 2.1. This results in the following dynamics:

$$f(r, u) = \begin{bmatrix} \frac{v_{\rho}}{cT} \\ \frac{v_t}{cT} \\ a_{\rho} + \frac{\frac{\rho}{cT}}{m_0 + \Delta m} \cos \alpha \\ a_t + \frac{\frac{v_t}{cT}}{m_0 + \Delta m} \sin \alpha \\ -\frac{cT}{v_{\text{exhaust}}} \end{bmatrix}, \quad (2.46)$$

where $c = T_{\max}\rho_0/(m_0V_0^2)$ is a normalisation constant. ρ_0 , m_0 , and v_0 denote the initial radius, mass, and velocity of the initial orbit, respectively. The optimal control policy and trajectory $(r(\cdot), u(\cdot)) \in \Pi_{r_0, t_f}^{\mathcal{K}, \mathcal{C}}$ can be constructed efficiently using the numerical approximation of ω . For a given $N \in \mathbb{N}$ we consider the time

step $h = \frac{1}{N}$ and a uniform grid of $[-t_f, 0]$ with spacing $s^k = \frac{k}{N}$. Let us define the state $\{r^k\}_{k=0}^N$ and control $\{u^k\}_{k=0}^{N-1}$ for the numerical approximation of the optimal trajectory and control policy. Setting r_0 as the initial orbit, we proceed by iteratively computing the control value

$$u^k(r^k) \in \arg \min_{u \in \mathcal{U}} \omega(r^k + hf(r^k, u), s^k) \bigvee g(r^k).$$

For a given $\omega(r^k, s^k)$, this is done by numerically taking the partial derivatives along each grid direction to estimate the costate vector, q , and then determining the optimal control values as the minimiser of the Hamiltonian, H . After u^k is determined we compute r^{k+1} using the Matlab `ode113` function, a variable-order Adams-Bashforth-Moulton method of order 1 to 13 [132], and increment k . For the implementation, we discretized the interval $[-t_f, 0]$ using $N = 4000$ grid points.

As shown in Figure 2.1, when considering orbits further than 4 km away from the surface of the asteroid, the variation of the gravitational acceleration along θ becomes negligible. It is, therefore, possible to approximate the gravitational terms in spherical coordinates as

$$U_\rho(\rho, \theta) \approx U_\rho(\rho)$$

$$U_\theta(\rho, \theta) \approx 0.$$

Using this approximation makes a_ρ and a_t independent of θ . This allows us to omit a grid dimension while numerically solving the quasi-variational inequality (2.34), greatly reducing the computational cost. Thus the final set of states used for the computation of the value function is

$$r = [\rho, v_\rho, v_t, \Delta m]^T \in \mathbb{R}^4. \quad (2.47)$$

During the calculation of the optimal trajectory, θ can easily be reconstructed by forward integrating the dynamics at each time step s^k , i.e.,

$$\theta^{k+1} = \theta^k + \int_{s^k}^{s^{k+1}} \frac{v_t^k}{\rho^k} ds. \quad (2.48)$$

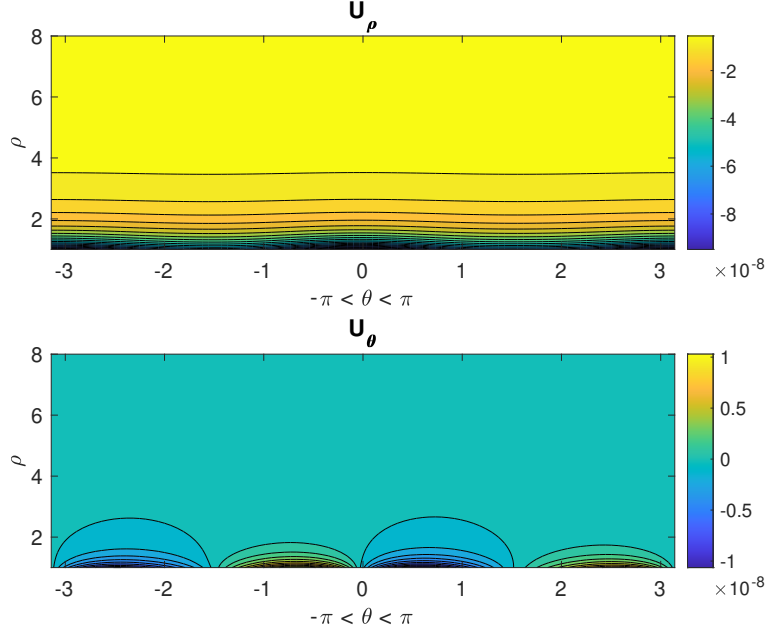


Figure 2.1: Gravitational acceleration comparing U_ρ and U_θ around Castalia 4769.

	ρ	v_ρ	v_t	Δm
Points	50	40	40	32
Spacing	0.0045	0.0416	0.0088	0.0067
Minimum	0.8067	-0.2495	-1.4154	-0.0533
Maximum	1.0270	1.3722	-1.0704	0.1533

Table 2.2: Spatial grid configuration

2.5.2 Simulation results

The spacecraft is modelled with 750 kg of dry mass, 600 mN of maximum thrust, and an exhaust velocity of 40 km/s. Using an initial orbit with radius 5.1 km and tangential velocity of -2.4 m/s, we are able to compute the numerical approximation of ω using the spatial grid described in Table 2.2 in combination with a temporal grid using $N = 1000$ grid points. The propagation of the zero level set over time is shown in Figure 2.2 and 2.3. Since the temporal grid for the calculation of the value function is more coarse than that used for the trajectory calculation, we need to interpolate the value function while computing the final trajectory. The final computed trajectory, for an initial propellant mass of 24.89 g and transfer time 2757 s is shown in Figure 2.4. The asteroid rendering for Figure 2.4 was computed

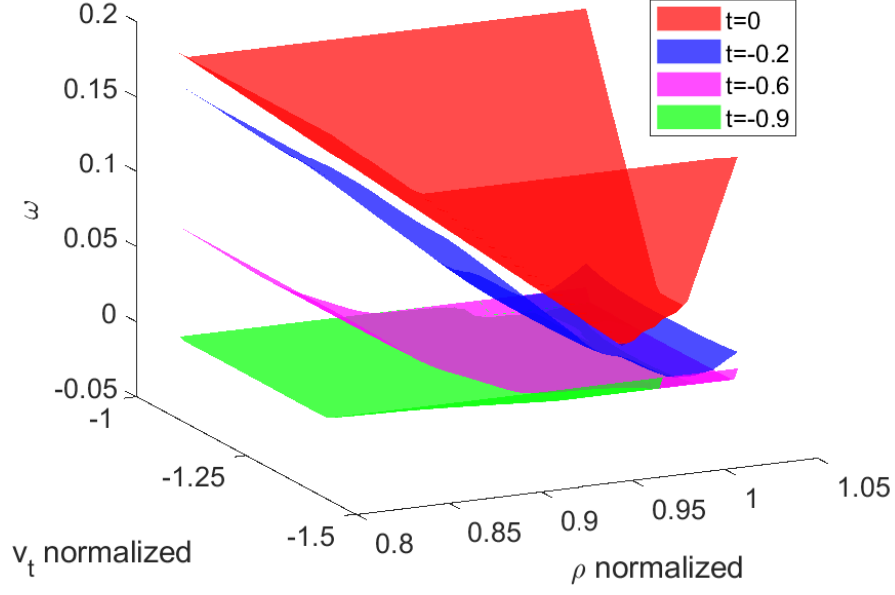


Figure 2.2: The evolution of the value function, ω , projected in the ρ - v_t plane for $v_\rho = 0$ and $\Delta m = 100$ g.

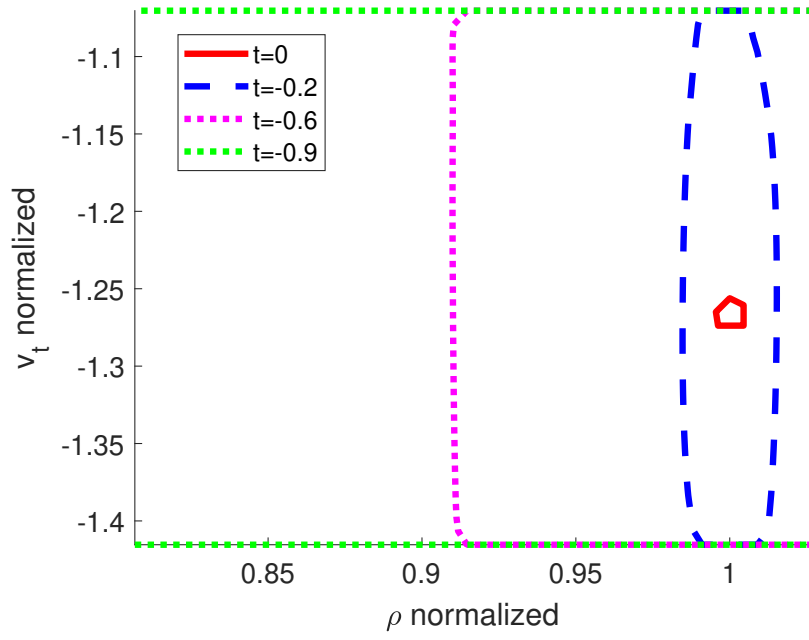


Figure 2.3: The evolution of the zero level set of the value function, ω , projected in the ρ - v_t plane for $v_\rho = 0$ and $\Delta m = 100$ g.

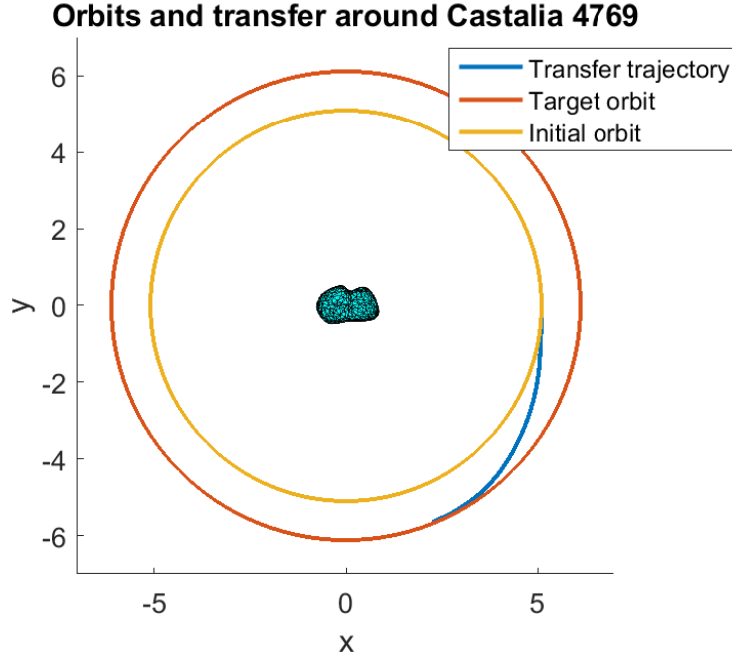


Figure 2.4: Initial orbit and transfer trajectory to a circular orbit ≈ 1 km further away from the asteroid.

as in [85]. The accuracy of the final orbit is within 41 meters of the target orbit. Calculating ω took 9 hours using a 3 GHz 8-Core Intel Core i7-9700 processor running Matlab with an extension of the Level Set Methods toolbox [40]. The run time and accuracy can be significantly improved upon when using optimised code such as [26, 38]. To show how the accuracy of the solution, the memory usage as well as the CPU time varies with the grid size, we recompute the value function and the trajectory with coarser spatial grids, as presented in Table 2.3. We use only single-precision arrays to store the value function, yet utilised double-precision arrays for all numerical calculations on the value function. The subscript *final* denotes the values of the final point of the computed trajectory, i.e., $r(0)$, while *target* refers to the target orbit used to define $\mathcal{C}$ and subsequently initialise the computation of the value function. Once ω is computed, it is incorporated into (2.15), which is solved using Matlab's `paretosearch` function. Solving the MOCP took 120 seconds and the resulting Pareto front is shown in Figure 2.5.

A comparison of the thrust magnitude of the smoothed optimal control policy is shown in Figure 2.6. As can be seen, the time-optimal solution uses near-continuous

	Low	Medium	High
Grid Size	[24 18 18 12]	[32 24 24 16]	[50 40 40 32]
CPU Time	20 min	44 min	9.28 hours
Memory to store ω	374 MB	1181 MB	10.25 GB
$ \rho_{\text{final}} - \rho_{\text{target}} $	82.72 m	64.72 m	40.83 m
$ v_{\rho,\text{final}} - v_{\rho,\text{target}} $	$6.80\text{e-}5 \frac{\text{m}}{\text{s}}$	$4.26\text{e-}5 \frac{\text{m}}{\text{s}}$	$8.31\text{e-}7 \frac{\text{m}}{\text{s}}$
$ v_{t,\text{final}} - v_{t,\text{target}} $	$2.40\text{e-}4 \frac{\text{m}}{\text{s}}$	$4.01\text{e-}4 \frac{\text{m}}{\text{s}}$	$2.41\text{e-}4 \frac{\text{m}}{\text{s}}$

Table 2.3: Comparison of the accuracy, CPU time, and memory usage as the size of the grid changes.

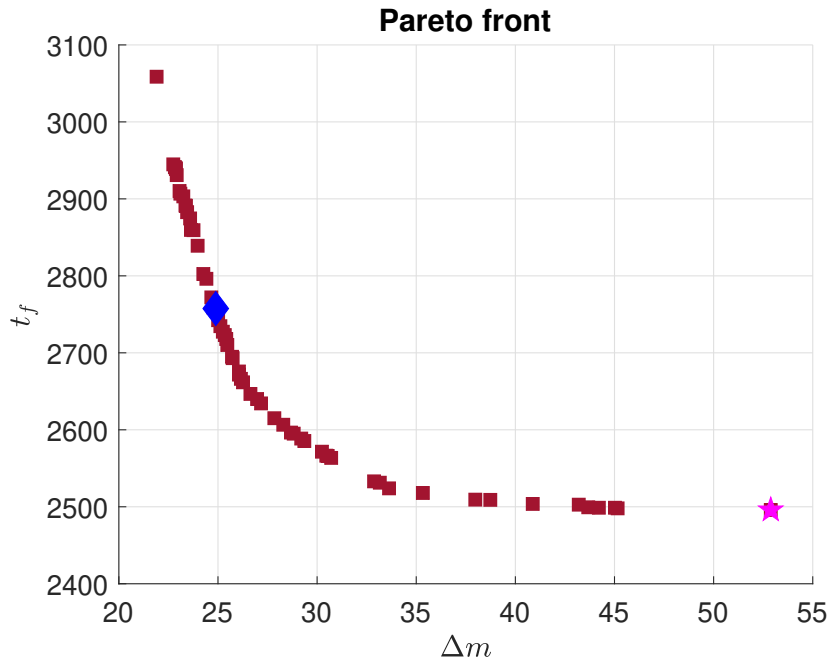


Figure 2.5: Pareto front of the objective functions $J_1 = \Delta m$ in grams and $J_2 = t_f$ in seconds. The trajectory in Figure 2.4 is derived using the point marked by the black diamond, while the time-optimal solution used in Figure 2.6 uses the point denoted by the magenta pentagram.

thrust to reach the target, at the cost of using a large amount of fuel. The control policy of the trajectory presented in Figure 2.4 meanwhile, has noticeable cruising phases where no fuel is consumed. As expected, the thrust magnitude of both policies follows a bang-bang structure.

To illustrate the results of problems in Bolza form, we consider the case of optimising the remaining propellant instead of the initial propellant. Therefore, let us fix the initial propellant to 100 g. We use a similar setup as in 2.5, with the addition of the auxiliary state, z , defined as a terminal cost. While the

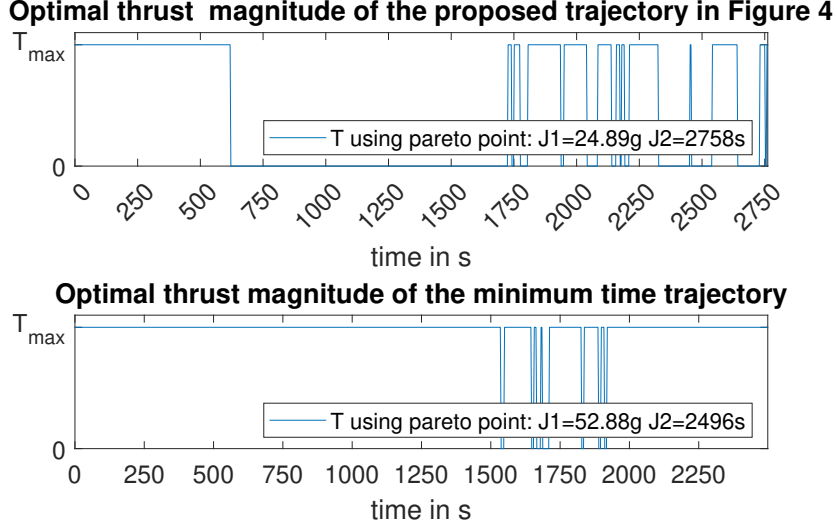


Figure 2.6: A comparison of the thrust magnitude of the smoothed optimal control policy of the time-optimal solution and the trajectory presented in Figure 2.4.

	ρ	v_ρ	v_t	Δm	z
Points	32	24	24	32	32
Spacing	0.007	0.069	0.016	0.0067	0.0067
Minimum	0.805	-0.277	-1.446	-0.0533	-0.1533
Maximum	1.029	1.317	-1.068	0.1533	0.0533

Table 2.4: Spatial grid configuration for the Bolza problem.

MOCP (2.32) is computationally cheaper for the multi-objective optimisation solver, the additional computational cost that results from the inclusion of the additional state t_f in the augmented value function $\tilde{\omega}$ makes the reachability problem (2.40) impractical for large time horizons. To this end, we refer to [145] for results of the augmented reachability problem, where only small orbital corrections are considered. For the remainder of this section, we instead solve the MOCP (2.23). We consider the uniformly spaced grid over r and z , defined in Table 2.4. Storing the value function using single precision requires 15.2 GB of memory. Since the objective is to maximise the remaining propellant, the optimisation problem needs to minimise $-\Delta m$. Therefore, for a given final state $r_f \in \mathbb{R}^5$ (defined as in Equation (2.45)) and transfer time $t_f \in [0, +\infty)$, we define the cost functions as $J_1(r_f, t_f) := -\Delta m$ and $J_2(r_f, t_f) := t_f$, where Δm denotes the 5-th element of

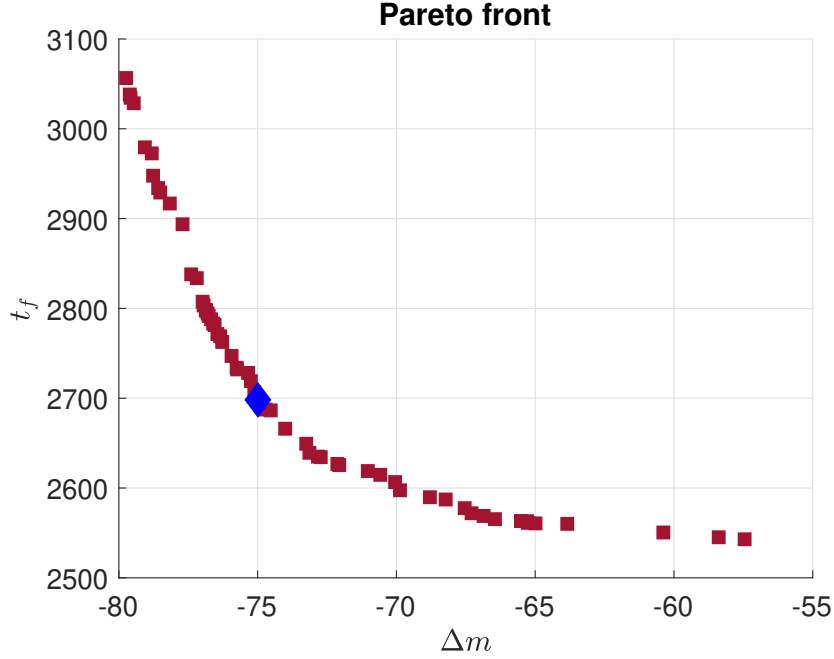


Figure 2.7: Pareto front of the objective functions $J_1 = z$ in grams and $J_2 = t_f$ in seconds, for the Bolza problem. The point used for trajectory calculations is marked by the black diamond.

the state vector r_f (the mass in our case). The 2-dimensional objective function $J : \mathbb{R}^5 \times [0, +\infty) \rightarrow \mathbb{R}^2$ can then be written as

$$J(r_f, t_f) := [J_1(r_f, t_f), J_2(r_f, t_f)]^T. \quad (2.49)$$

The resulting Pareto front is shown in Figure 2.7. Calculating the reachable set took approximately 32 hours of CPU time. Using the reachable set, calculating the Pareto front took approximately 258 seconds. As expected, the Pareto front looks similar to that of Figure 2.5, yet not identical due to numerical inaccuracy and change in the initial mass, resulting in modified dynamics.

Using an initial propellant mass of 100 g, z of -75 g, and transfer time of 2698 s, the transfer orbit is computed in the same way as before. As expected, the transfer trajectory is similar to the trajectory shown in Figure 2.4, and the accuracy of the transfer orbit is within 66 meters of the target orbit. For comparison to the first approach, the accuracy of the second approach is shown in Table 2.5. From comparing the Pareto front, it can be seen that both methods have a

	Bolza-formulation
Grid Size	[32 24 24 32 32]
CPU Time	32 hours
Memory to store ω	15.2 GB
$ \rho_{\text{final}} - \rho_{\text{target}} $	66.14 m
$ v_{\rho,\text{final}} - v_{\rho,\text{target}} $	$1.8\text{e-}6 \frac{m}{s}$
$ v_{t,\text{final}} - v_{t,\text{target}} $	$0.0018 \frac{m}{s}$

Table 2.5: Comparison of the accuracy, CPU time, and memory usage for the second formulation in Bolza form.

minimum transfer time of around 2500 seconds as well as a minimum propellant requirement of just over 20 g.

From the computation times of both the Bolza and original problem, it can be seen that, while the MOCP becomes considerably more tractable, the computational load of the overall problem has been shifted to the reachability problem. However, compared to previous approaches whereby the Pareto front is computed directly from the value function [145], the presented approach sees a significant overall computational improvement.

2.6 Conclusion

We have presented an extension of HJ reachability to MOCPs in Bolza form. Using the subzero level set of the value function, we express the feasible region for all solutions of the MOCP using an inequality constraint. We demonstrate the feasibility of the proposed approach by considering the design of a low-thrust trajectory around a rotating asteroid. As the underlying reachability problem is still subject to the curse of dimensionality, we rely on potentially coarse grids for solving the HJB equation. In light of this, the subsequent chapter will introduce an approach towards empirically estimating the numerical error and thus understanding when the grid size needs to be increased to reach a desired level of numerical accuracy

2.7 Appendix

Proposition 2.3

Proposition 2.3. *Under Assumption 2.2, any two trajectories $r(\cdot)$ and $\hat{r}(\cdot)$ reconstructed from f , with $r(-t_f) = r_0$ and $\hat{r}(-t_f) = \hat{r}_0$, respectively, are such that $\|r(t) - \hat{r}(t)\| \leq \|r_0 - \hat{r}_0\|e^{(t_f+t)L_f}$ for all $t \in [-t_f, 0]$.*

Proof. Let $r_0, \hat{r}_0 \in \mathbb{R}^5$ be two initial states, and $t_f \in [0, \infty)$. For the same t_f , we choose two trajectory control pairs $(r(\cdot), u(\cdot)) \in \Pi_{r_0, t_f}$ and $(\hat{r}(\cdot), \hat{u}(\cdot)) \in \Pi_{\hat{r}_0, t_f}$. Then by Carathéodory's existence of solutions [125], the following relation holds:

$$\begin{aligned} \|r(-t) - \hat{r}(-t)\| &\leq \|r_0 - \hat{r}_0\| + \int_{-t_f}^{-t} \|f(r(s), u(s)) - f(\hat{r}(s), \hat{u}(s))\| ds \\ &\leq \|r_0 - \hat{r}_0\| + L_f \int_{-t_f}^{-t} \|r(s) - \hat{r}(s)\| ds \\ &\leq \|r_0 - \hat{r}_0\| e^{(t_f-t)L_f}, \end{aligned}$$

where the second inequality is due to Assumption 2.2, while the last inequality is due to the Bellman-Gronwall Lemma [125]. $\square$

Proof of Proposition 2.2

Proof. Fix $(r_0, z_0), (\hat{r}_0, \hat{z}_0) \in \mathbb{R}^7 \times \mathbb{R}^p$, $t_f \in [0, \infty)$ and let $\epsilon > 0$. We choose $(\hat{r}(\cdot), \hat{u}(\cdot), \hat{z}(\cdot)) \in \mathcal{Z}_{\hat{r}_0, t_f, \hat{z}_0}$, such that

$$\vartheta(\hat{r}_0, t_f, \hat{z}_0) \geq \bigvee_i \left[J_i^t(\hat{r}(0)) - \hat{z}_i(-t_f) \right] \bigvee \nu(\hat{r}(0)) \bigvee \max_{s \in [-t_f, 0]} g(\hat{r}(s)) - \epsilon.$$

By definition of ϑ , for any $(r(\cdot), u(\cdot)) \in \Pi_{r_0, t}$, this yields the following relation

$$\begin{aligned} \vartheta(r_0, t_f, z_0) - \vartheta(\hat{r}_0, t_f, \hat{z}_0) &\leq \bigvee_i \left[J_i^t(r(0)) - z_i(-t_f) \right] \bigvee \nu(r(0)) \bigvee \max_{s \in [-t_f, 0]} g(r(s)) \\ &\quad - \bigvee_i \left[J_i^t(\hat{r}(0)) - \hat{z}_i(-t_f) \right] \bigvee \nu(\hat{r}(0)) \bigvee \max_{s \in [-t_f, 0]} g(\hat{r}(s)) + \epsilon. \end{aligned}$$

Let $\kappa \in [-t_f, 0]$ be such that

$$g(r(\kappa)) = \max_{s \in [-t_f, 0]} g(r(s)).$$

We then have

$$\begin{aligned} \vartheta(r_0, t_f, z_0) - \vartheta(\hat{r}_0, t_f, \hat{z}_0) &\leq \bigvee_i [J_i^t(r(0)) - z_i(-t_f)] \bigvee \nu(r(0)) \bigvee g(r(\kappa)) \\ &\quad - \bigvee_i [J_i^t(\hat{r}(0)) - \hat{z}_i(-t_f)] \bigvee \nu(\hat{r}(0)) \bigvee g(\hat{r}(\kappa)) + \epsilon. \end{aligned}$$

Using Proposition 2.3, we distinguish the following cases.

Case A: $g(r(\kappa)) \geq \bigvee_i [J_i^t(r(0)) - z_i(-t_f)] \bigvee \nu(r(0))$

$$\begin{aligned} \vartheta(r_0, t_f, z_0) - \vartheta(\hat{r}_0, t_f, \hat{z}_0) &\leq g(r(\kappa)) \\ &\quad - \bigvee_i [J_i^t(\hat{r}(0)) - \hat{z}_i(-t_f)] \bigvee \nu(\hat{r}(0)) \bigvee g(\hat{r}(\kappa)) + \epsilon \\ &\leq g(r(\kappa)) - g(\hat{r}(\kappa)) + \epsilon \leq L_g e^{(t_f + \kappa)L_f} \|r_0 - \hat{r}_0\| + \epsilon, \end{aligned}$$

where the last inequality is due to the fact that g is Lipschitz continuous.

Case B: $\nu(r(0)) \geq \bigvee_i [J_i^t(r(0)) - z_i(-t_f)] \bigvee g(r(\kappa))$

$$\begin{aligned} \vartheta(r_0, t_f, z_0) - \vartheta(\hat{r}_0, t_f, \hat{z}_0) &\leq \nu(r(0)) \\ &\quad - \bigvee_i [J_i^t(\hat{r}(0)) - \hat{z}_i(-t_f)] \bigvee \nu(\hat{r}(0)) \bigvee g(\hat{r}(\kappa)) + \epsilon \\ &\leq \nu(r(0)) - \nu(\hat{r}(0)) + \epsilon \leq L_\nu e^{t_f L_f} \|r_0 - \hat{r}_0\| + \epsilon \end{aligned}$$

Case C: $\bigvee_i [J_i^t(r(0)) - z_i(-t_f)] \geq g(r(\kappa)) \bigvee \nu(r(0))$

Recall (2.20), then under Assumption 2.2, any two trajectories $z(\cdot)$ and $\hat{z}(\cdot)$ reconstructed from J^r with z_0 and $\hat{z}_0$, respectively, are bounded within a given time interval $[-t, 0]$. To see this,

$$\begin{aligned} \|z(-t) - \hat{z}(-t)\| &\leq \|z_0 - \hat{z}_0\| + \int_{-t}^0 \|J^r(r(s), u(s)) - J^r(\hat{r}(s), \hat{u}(s))\| ds \\ &\leq \|z_0 - \hat{z}_0\| + L^r \int_{-t}^0 \|r(s) - \hat{r}(s)\| ds \\ &\leq \|z_0 - \hat{z}_0\| + L^r \int_{-t}^0 \|r_0 - \hat{r}_0\| e^{(t_f + s)L_f} ds \\ &\leq \|z_0 - \hat{z}_0\| + \|r_0 - \hat{r}_0\| L^r e^{t_f L_f} \frac{1 - e^{-t L_f}}{L_f}, \end{aligned}$$

where the third inequity is due to Proposition 2.3, and the last one follows by performing the integration. Next, let $j \in [1, \dots, p]$ be such that

$$J_j^t(r(0)) - z_j(-t_f) = \bigvee_i [J_i^t(r(0)) - z_i(-t_f)].$$

Then it follows, that

$$\begin{aligned}
\vartheta(r_0, t_f, z_0) - \vartheta(\hat{r}_0, t_f, \hat{z}_0) &\leq \bigvee_i \left[J_i^t(r(0)) - z_i(-t_f) \right] \\
&\quad - \bigvee_i \left[J_i^t(\hat{r}(0)) - \hat{z}_i(-t_f) \right] \bigvee \nu(\hat{r}(0)) \bigvee g(\hat{r}(\kappa)) + \epsilon \\
&\leq \bigvee_i \left[J_i^t(r(0)) - z_i(-t_f) \right] \\
&\quad - \bigvee_i \left[J_i^t(\hat{r}(0)) - \hat{z}_i(-t_f) \right] + \epsilon \\
&\leq \left[J_j^t(r(0)) - z_j(-t_f) \right] - \left[J_j^t(\hat{r}(0)) - \hat{z}_j(-t_f) \right] + \epsilon \\
&\leq \left[J_j^t(r(0)) - J_j^t(\hat{r}(0)) \right] - [z_j(-t_f) - \hat{z}_j(-t_f)] + \epsilon
\end{aligned}$$

By Proposition 2.3 and under Assumption 2.3

$$\left[J_j^t(r(0)) - J_j^t(\hat{r}(0)) \right] \leq L^t \|r_0 - \hat{r}_0\| e^{t_f L_f}.$$

Finally, this yields the relationship

$$\begin{aligned}
&\left[J_j^t(r(0)) - J_j^t(\hat{r}(0)) \right] - [z_j(-t_f) - \hat{z}_j(-t_f)] + \epsilon \\
&\leq \|z_0 - \hat{z}_0\| + \|r_0 - \hat{r}_0\| \left[L^t e^{t_f L_f} + L^r \frac{e^{t_f L_f} - 1}{L_f} \right] + \epsilon.
\end{aligned}$$

Thus in every case, on an interval $[0, t_f]$ there exists a set of constants C_r and C_z , such that

$$\vartheta(r_0, t_f, z_0) - \vartheta(\hat{r}_0, t_f, \hat{z}_0) \leq C_r \|r_0 - \hat{r}_0\| + C_z \|z_0 - \hat{z}_0\| + \epsilon$$

The same argument conducted with (r_0, t_f, z) and $(\hat{r}_0, t_f, \hat{z})$ reversed establishes that

$$\vartheta(\hat{r}_0, t_f, \hat{z}_0) - \vartheta(r_0, t_f, z_0) \leq C_r \|r_0 - \hat{r}_0\| + C_z \|z_0 - \hat{z}_0\| + \epsilon.$$

Since ϵ is arbitrary, we conclude that

$$\|\vartheta(\hat{r}_0, t_f, \hat{z}_0) - \vartheta(r_0, t_f, z_0)\| \leq C_r \|r_0 - \hat{r}_0\| + C_z \|z_0 - \hat{z}_0\|,$$

thus concluding the proof. $\square$

The proof for ω is similar to that of ϑ and will therefore be omitted in the interest of space.

Derivation of the optimal thrust angles

Notice that since the applied thrust, T , is always positive, the term

$$(q_4 \cos \alpha + \sin \alpha (q_5 \sin \delta + q_6 \cos \delta))$$

in the Hamiltonian can be minimised separately from T . To this end, we introduce the auxiliary variables

$$\begin{aligned}\chi(\delta) &:= \sqrt{q_5^2 + q_6^2} \cos(\delta - \arctan \frac{q_5}{q_6}), \\ A(\delta) &:= \sqrt{q_4^2 + \chi(\delta)^2},\end{aligned}$$

using the trigonometric identity

$$a \cos x + b \sin x = R \cos(x - \arctan \frac{b}{a}),$$

with $R = \sqrt{a^2 + b^2}$. First optimising over α , and subsequently over δ (notice that this sequential minimisation is exact since $A(\delta) \geq 0$) results in

$$\begin{aligned}\min_{\alpha, \delta \in [-\pi, \pi] \times [-\frac{\pi}{2}, \frac{\pi}{2}]} (q_4 \cos \alpha + \sin \alpha (q_5 \sin \delta + q_6 \cos \delta)) \\ = \min_{\delta \in [-\frac{\pi}{2}, \frac{\pi}{2}]} A(\delta) \min_{\alpha \in [-\pi, \pi]} \cos(\alpha - \arctan \frac{\chi(\delta)}{q_4}).\end{aligned}$$

Thus, the optimal thrust angles are given by

$$\alpha^*(\delta) := \pm\pi + \arctan \frac{\chi(\delta)}{q_4}.$$

Since $\cos(\alpha^*(\delta) - \arctan \frac{\chi(\delta)}{q_4}) = -1$, after applying $\alpha^*(\delta)$, it follows that

$$\delta^* \in \arg \min_{\delta \in [-\frac{\pi}{2}, \frac{\pi}{2}]} -A(\delta) = \arctan \frac{q_5}{q_6} \pm \pi.$$

Subsequently,

$$q_4 \cos \alpha^* + \sin \alpha^* (q_5 \sin \delta^* + q_6 \cos \delta^*) = -\sqrt{q_4^2 + q_5^2 + q_6^2}. \quad (2.50)$$

Substituting (2.50) into (2.35) results in (2.37) which depends in an affine fashion on T .

3

Verification of safety-critical control policies

3.1 Introduction

Learning-based controllers are becoming increasingly relevant for robotics applications [65, 69, 114, 130]. As a result, a steady increase in methods that allow a system to safely learn its dynamics has been noticed. Such methods provide safety guarantees, in terms of the *classical* notion of safety, while allowing a safe exploration of the state-space [17]. Commonly this is done by using a hybrid control approach, whereby a nominal learning-based controller is used to explore the state-space and a safety-critical controller is activated whenever confidence in the safe performance of the system is lost, thus driving the system back into a safe region of the state-space. A common example of such a setup is found in collision avoidance problems, whereby the safety controller is activated to prevent the nominal controller from driving the system too close to an obstacle. Another common example is a pursuit-evasion problem, whereby the safety controller ensures that no evasion strategy exists and thus the pursuer is guaranteed to be able to catch the evader. For the design of the safety controller, both of these problems can be modelled as differential games [104], whereby the disturbance is modelled as an adversarial player.

In [65] the authors introduce safety controllers that estimate the confidence in the underlying model and if the system is shown to start to violate the underlying safety assumption, a safety controller is activated. In a similar fashion, we will derive a confidence metric for when to activate a safety controller, but instead of the confidence metric capturing the accuracy of the system model and possible range of disturbances, we will assert our confidence in the accuracy of the safety controller itself to perform as intended. This chapter focuses on the derivation of the confidence metric and will therefore not test hybrid control setups. Furthermore, as in [65], we restrict ourselves to verifying safety controllers synthesised using Hamilton-Jacobi (HJ) reachability analysis.

The HJ approach has been applied to various real-world applications such as air traffic management [98], payload optimisation for multi-stage launchers [25], aircraft abort landing [8], and in [46, 145] we see how the approach can be used to construct the Pareto front of a multi-objective optimal control problem. All these approaches rely on the fundamental assumption that the computation of the safety controller is accurate and imposes no numerical errors. Thus, the safety controller only takes into account the variations in the dynamics and disturbances modelled prior to deployment. Yet especially high dimensional problems exponentially increase the required memory, computational cost, and numerical dissipation [39] and thus the numerically computed value function – on which the optimal control computation is based – may differ from the actual one. Furthermore, even though slight changes in the dynamics or estimated disturbances might not affect the controller, it is still necessary to recompute the underlying value function used for controller estimation. We, therefore, present a method of predicting the error of these controllers using common regression models. By sampling the error in areas of interest, we are able to predict the marginal likelihood that the computed safety controller will in fact lead to safe system behaviour.

The rest of the chapter is organised into five sections. Section 3.2 contains details regarding the reachability problem, including the characterisation of the backwards reachable tube (BRT) as the sub-zero level set of a value function, which in turn is

the unique viscosity solution of a quasi-variational inequality. Section 3.3 discusses the basic concept of using a Gaussian process (GP) to model the error of the aforementioned value function with Section 3.4 dedicated to a case study involving a pursuit-evasion example. Finally, Section 3.5 provides concluding remarks.

3.2 Problem Setup

As previously discussed, a key concept in reachability analysis is to compute areas of the state-space that ensure safety. In the context of viability theory, such regions are known as capture basins [10] and describe the region from which a system can be driven to a given target while avoiding undesired states (such as obstacles for autonomous cars or overheating of a transformer).

Let $x \in \mathbb{R}^n$ be the system state with Lipschitz continuous dynamics $f : \mathbb{R}^n \times \mathcal{U} \times \mathcal{D} \rightarrow \mathbb{R}^n$ such that

$$\dot{x} = f(x(t), u(t), d(t)), \quad t \in [-t_f, 0], \quad (3.1)$$

The sets $\mathcal{U}$ and $\mathcal{D}$ are compact and denote the set of possible control and disturbance inputs. The solution of the ordinary differential equation in (3.1) is denoted by the trajectory $\zeta : [-t_f, 0] \rightarrow \mathbb{R}^n$, which starts from state x_0 at time $-t_f$ and is governed by the control policy $u(\cdot) : [-t_f, 0] \rightarrow \mathcal{U}$ and disturbance policy $d(\cdot) : [-t_f, 0] \rightarrow \mathcal{D}$:

$$\begin{aligned} \frac{\partial}{\partial t} \zeta(t; x_0, u(\cdot), d(\cdot)) &= f(\zeta(t; x_0, u(\cdot), d(\cdot)), u(t), d(t)) \\ \zeta(-t_f; x_0, u(\cdot), d(\cdot)) &= x_0. \end{aligned}$$

We denote the space of admissible control and disturbance policies by $\mathcal{U}_{ad}$ and $\mathcal{D}_{ad}$ respectively.

To formally discuss the aims of HJ reachability analysis, we will introduce the BRT [39]. The BRT describes the set of states from which the system can be driven into a target set $\mathcal{R} \subset \mathbb{R}^n$ within the time horizon t_f . Furthermore, the system is guaranteed to never be driven into an avoid set $\mathcal{A} \subset \mathbb{R}^n$ prior to reaching $\mathcal{R}$. By determining the BRT, it becomes possible to formally verify that a system will behave as intended and satisfy the reach-avoid problem even under adversarial disturbances.

Let $l : \mathbb{R}^n \rightarrow \mathbb{R}$ and $h : \mathbb{R}^n \rightarrow \mathbb{R}$ denote the Lipschitz continuous signed distance functions to $\mathcal{R}$ and $\mathcal{A}$, respectively. Then we can characterise the reach and avoid sets as:

$$\mathcal{R} = \{x \in \mathbb{R}^n : l(x) \leq 0\} \quad (3.2)$$

$$\mathcal{A} = \{x \in \mathbb{R}^n : h(x) > 0\}. \quad (3.3)$$

Under a robust control lens, we will compute the BRT for the worst-case disturbance and thus ensure that our assumptions of reaching $\mathcal{R}$ and avoiding $\mathcal{A}$ hold for all possible disturbances. To this end, we reformulate our control problem as a reach-avoid differential game.

Let us introduce the nonanticipative strategy, which is a function $\xi : \mathcal{U}_{ad} \rightarrow \mathcal{D}_{ad}$, such that for all $t \in [-t_f, 0]$ and for all $u, \hat{u} \in \mathcal{U}_{ad}$, if $u(\tau) = \hat{u}(\tau)$ for almost every $\tau \in [-t_f, t]$, then $\xi[u](\tau) = \xi[\hat{u}](\tau)$ for almost every $\tau \in [-t_f, t]$ [56, 141]. Furthermore, we use Ξ to denote the class of nonanticipative strategies and we restrict the first player to only nonanticipative strategies. Then it is possible to describe the desired BRT as follows:

$$\begin{aligned} \text{BRT}_{\mathcal{A}, \mathcal{R}}(-t_f) = \{x | \exists \xi(\cdot) \in \Xi, \forall u(\cdot) \in \mathcal{U}_{ad}, [\exists t \in [-t_f, 0] : \zeta(t; x, u(\cdot), \xi(\cdot)) \in \mathcal{R}] \\ \& \quad [\forall \tau \in [-t_f, t], \zeta(\tau; x, u(\cdot), \xi(\cdot)) \notin \mathcal{A}]\} \end{aligned} \quad (3.4)$$

Thus the BRT contains the set of states from which a trajectory can start at time $-t_f$ and reach the target set $\mathcal{R}$ within the time horizon t_f without passing through the set $\mathcal{A}$. The somewhat unconventional choice of the disturbance trying to steer the system state towards the target set stems from the fact that in the case study of Section 3.4, the target set encodes the set of states for which two vehicles are in a collision, so it is in favour to the disturbance to enter this set. In order to determine $\text{BRT}_{\mathcal{A}, \mathcal{R}}(-t_f)$, we introduce a value function that captures the outcome of the differential game.

$$\omega(x, t) = \inf_{\xi(\cdot) \in \Xi} \sup_{u(\cdot) \in \mathcal{U}_{ad}} \max \left\{ \min_{\tau_1 \in [-t, 0]} l(\zeta(\tau_1; x, u(\cdot), \xi(\cdot))), \max_{\tau_2 \in [-t, \tau_1]} h(\zeta(\tau_2; x, u(\cdot), \xi(\cdot))) \right\}, \quad (3.5)$$

where ξ is trying to minimise, and u is trying to maximise the maximum between the minimum over all values attained by l and the maximum value attained by h over the time horizon required to reach $\mathcal{R}$. In [97] the authors show that the value function is the unique continuous viscosity solution of the following quasi-variational inequality:

$$\max \left\{ h(x) - \omega(x, t), \frac{\partial \omega(x, t)}{\partial t} + \min \left[0, \sup_{u \in \mathcal{U}} \inf_{d \in \mathcal{D}} \frac{\partial \omega(x, t)}{\partial x} f(x, u, d) \right] \right\} = 0. \quad (3.6)$$

By characterising $\omega(x, t)$ as a viscosity solution of a quasi-variational inequality, we now only need to optimise over possible control and disturbance inputs, rather than having an infinite-dimensional optimisation problem over control and disturbance policies. This makes the characterisation of the BRT more ascertainable. Once the value function has been obtained, its sub-zero level set characterises the BRT defined in (3.4), i.e., $\text{BRT}_{\mathcal{A}, \mathcal{R}}(-t_f) = \{x \in \mathbb{R}^n | \omega(x, t) \leq 0\}$.

The optimal control policy and worst-case disturbance used for trajectory reconstruction are the optimisers of

$$\sup_{u \in \mathcal{U}} \inf_{d \in \mathcal{D}} \frac{\partial \omega(x, t)}{\partial x} f(x, u, d). \quad (3.7)$$

Notice that the ordering of $\sup_{u \in \mathcal{U}} \inf_{d \in \mathcal{D}}$ results in the optimal disturbance depending on the optimal control action. Thus the *advantage* is given to the disturbance as it is allowed to choose its action on the basis of the chosen control action. By saving the control and disturbance values during the computation of $\omega(x, t)$, we are simultaneously computing the optimal control policy.

Having computed the optimal control policy for the safety controller which we denote by $u_{\text{safety}}(\cdot)$, we can plug it into a hybrid control setup. To this end, let u_{nom} denote the control policy of an arbitrary nominal controller. The overall least-restrictive control law can be defined as

$$u^*(t) := \begin{cases} u_{\text{nom}}(t) & \text{if } \omega(x, t) \leq -\delta \\ u_{\text{safety}}(t) & \text{otherwise,} \end{cases} \quad (3.8)$$

where $\delta > 0$ is a chosen tolerance. Thus if the nominal control policy, u_{nom} , takes the system within a distance δ of the boundary of the BRT, the safety controller is activated. The use of ω as a manoeuvrability envelope as in (3.8) requires Lipschitz continuity of ω which can be shown following similar argumentation as in Section 2.7 and has been previously employed in [65, 69].

3.3 Using Gaussian Processes to verify a control policy

3.3.1 Error of the value function

To illustrate the criticality the numerical error plays in safety-critical control design, we consider the basic pursuit-evasion setup presented in Section 3.4. The aim is to determine the set of states from which a pursuer is guaranteed to be able to catch the evader. We use $\sim$ to denote the numerical solutions. Solving the quasi-variational inequality (3.6) with terminal condition $\omega(x, 0) = \max(l(x), h(x))$, allows us to determine the numerical value function $\tilde{\omega}$ as well as the optimal control and disturbance policies $\tilde{u}$ and $\tilde{d}$, respectively. At time $t = -0.42$, for the relative state $[x_1 = 0.078, x_2 = -0.51, x_3 = 0.20]$, we have $\tilde{\omega} = 0.025 > 0$ and thus the state $[x_1 = 0.078, x_2 = -0.51, x_3 = 0.20]$ does not lie in the BRT, implying it is not possible for the pursuer to catch the evader.

Let us define the value function for a given control and disturbance policy as

$$\omega_{u,d} := \max \left\{ \min_{\tau_1 \in [-t, 0]} l(\zeta(\tau_1; x, u(\cdot), d(\cdot))), \max_{\tau_2 \in [-t, \tau_1]} h(\zeta(\tau_2; x, u(\cdot), d(\cdot))) \right\}, \quad (3.9)$$

where the policy d is derived using a nonanticipative strategy as previously discussed. Then when applying the computed control and disturbance policy, $\tilde{u}$ and $\tilde{d}$, respectively, we notice that $\omega_{\tilde{u}, \tilde{d}} = -0.01$. Thus, even the potentially sub-optimal policies $\tilde{u}$ and $\tilde{d}$ results in the pursuer being able to catch the evader, making the numerical characterisation of the BRT inaccurate. In order to correctly predict that the state $[x_1 = 0.078, x_2 = -0.51, x_3 = 0.20]$ does in fact lie within the BRT, we need to be able to determine the difference between the true value function and the numerical value function.

Since quasi-variational inequalities, such as (3.6), contain multivariate functions of independent variables, their solutions can be defined as the relationship between two or more variables. If the computed relationship is slightly off, we are left with an approximation of the exact solution. Thus, let us consider the error term ϵ , comprised of the exact solution ω as well the numerical solution $\tilde{\omega}$ obtained by

solving (3.6). The error can be considered as the cumulative numerical error as well as the difference to the true value function under changes in the dynamics, environment, or disturbances,

$$\epsilon(x, t) = \tilde{\omega}(x, t) - \omega(x, t). \quad (3.10)$$

Since ω is usually unknown, it is not possible to determine the error ϵ perfectly and we will instead approximate it as follows:

$$\tilde{\epsilon}(x, t) = \tilde{\omega}(x, t) - \omega_{\tilde{u}, \tilde{d}}, \quad (3.11)$$

where $\tilde{u}(\cdot)$ and $\tilde{d}(\cdot)$ are the control and disturbance policies determined by means of (3.7) using the approximated value function $\tilde{\omega}(x, t)$. For a true modelling of ϵ , we would need to take $\inf_{\xi(\cdot) \in \Xi} \sup_{u(\cdot) \in \mathcal{U}_{ad}} \omega_{u, \xi}$ instead of $\omega_{\tilde{u}, \tilde{d}}$. Since the aim is to verify the performance of a control and disturbance policy, this approximation of ϵ is sufficient.

For solving (3.6), we employ level set methods, which have the advantage of not requiring us to parameterise the solution, yet they have the disadvantage of not guaranteeing the conservation of shape and size in an advection field. Upwinding methods (whereby the approximation to the spatial derivatives of the value function is based on the sign of the velocity of the advection field) greatly reduce the numerical dissipation, yet small features will still be dominated and lost as we integrate forward [110]. Areas of the grid that contain many such features, will thus contain larger cumulative numerical error than others. At the same time, changes to the dynamics, environment or disturbances, are likely to only affect the safety-critical control policy in certain areas of the state-space.

Therefore, there is a correlation between the grid and the cumulative error. If we model the error in the context of statistics and probability theory, we are trying to find the probabilistic distribution of the error in our value function. More precisely, we are trying to determine the covariance between the variables in our value function and the error. We are thus faced with a classic regression problem. In light of this, we expand the notation to denote trained solutions to

our regression problem with a $\hat{\cdot}$ superscript. Thus $\hat{\epsilon}$ denotes the predicted error determined by a regression model fitted to finite samples of $\tilde{\epsilon}$ and consequently, $\hat{\omega}(x, t) := \tilde{\omega}(x, t) - \hat{\epsilon}(x, t)$. If $\hat{\epsilon}$ approximates ϵ (defined in (3.10)) perfectly, $\hat{\omega}$ will also perfectly approximate the true solution ω .

3.3.2 Prediction using Gaussian Processes

There are a variety of different regression models that are well studied and we briefly motivate why we opt for a Gaussian process (GP) regression in the next section.

In general, a GPR model is a kernel-based probabilistic model of the form

$$b(x)\beta + g(x), \quad (3.12)$$

where $b(x)$ is a set of basis functions, β is a vector of basis function coefficients and $g(x)$ is a zero mean GP with covariance function, $k(x, \hat{x})$, i.e., $g(x) \sim GP(0, k(x, \hat{x}))$ [119].

The assumption that $g(x) \sim GP(0, k(x, \hat{x}))$, allows us to determine a marginal likelihood. However, there is no reason to believe that the error of the value function follows a Gaussian distribution. Yet experimentation shows that, nevertheless, the error can be sufficiently approximated using a GP and the marginal likelihood offers us a framework for predicting the confidence in the estimated error. The marginal likelihood is based on the standard deviation, σ , and thus if the standard deviation is deemed to be small, we have previously verified a similar state and the confidence in the prediction is high, while in turn a large standard deviation implies that we have little confidence in our prediction. The standard deviation can, therefore, be used in a hybrid control setup to determine when to switch between a nominal controller and the safety controller. We, therefore, modify the least-restrictive control law (3.8) to include the standard deviation

$$u^*(t) := \begin{cases} u_{\text{nom}}(t) & \text{if } (\hat{\omega}(x, t) \leq -\delta) \wedge (\sigma < \sigma_0) \\ u_{\text{safety}}(t) & \text{otherwise,} \end{cases} \quad (3.13)$$

where σ_0 is a threshold set by the designer to determine the predefined confidence that should be maintained at all times.

3.4 Case Study

3.4.1 A Game of Two Identical Vehicles

In order to show how a GP can be used to determine the error of a computed value function, we will introduce a pursuit-evasion example commonly used in HJ reachability analysis.

Let us consider two identical vehicles, for example, two cars. One vehicle, the pursuer, wants to catch the other vehicle, the evader. This is done by coming within a distance r_1 of the evader. The evader, on the other hand, successfully outruns the pursuer if it has put a distance r_2 between itself and the pursuer.

To model this behaviour, we will have one vehicle (the pursuer) take on the best-case behaviour, i.e., it will actively try and catch the other vehicle (the evader). Meanwhile, the evader will play a nonanticipative strategy and try and avoid the pursuer.

Let us introduce the relative dynamics:

$$\begin{aligned}\dot{x}_1 &= -v_e + v_p \cos(2\pi x_3) + dx_2 \\ \dot{x}_2 &= v_p \sin(2\pi x_3) - dx_1 \\ \dot{x}_3 &= \frac{u - d}{2\pi},\end{aligned}$$

where x_1 and x_2 represent the relative distance between the two vehicles and x_3 describes the relative normalised heading. u and d are the angular velocities of the vehicles, where u will be considered our control variable, and d our disturbance variable. Both the control and the disturbance will be bounded by $\bar{u}$ and $\bar{d}$. Finally, v_e and v_p are the constant linear velocities of the evader and pursuer. x_1 and x_2 are scaled to lie between $[-1, 1]$, while x_3 lies between $[0, 1]$. This allows us to compare the regression models without the influence of largely varying scales.

Let us define the target and avoid set as

$$\mathcal{R} = \{x : \|(x_1, x_2)\|_2 \leq r_1\}, \quad (3.14)$$

$$\mathcal{A} = \{x : \|(x_1, x_2)\|_2 \geq r_2\}, \quad (3.15)$$

where r_1 is the distance between the two vehicles that needs to be reached for the pursuer to have caught the evader. In turn, r_2 is the distance at which the evader is deemed to have outrun the pursuer. Naturally, we select $r_1 < r_2$.

To compute the maximal BRT, let us define

$$l(x) := \|(x_1, x_2)\|_2 - r_1, \quad (3.16)$$

as the signed distance to our target (reach) set, and

$$h(x) := \|(x_1, x_2)\|_2 - r_2, \quad (3.17)$$

as the signed distance to our avoid set. Next, we will define the Hamiltonian as the solution of $\sup_{u \in \mathcal{U}} \inf_{d \in \mathcal{D}} \frac{\partial \omega(x, t)}{\partial x} f(x, u, d)$. The spatial derivatives of the value function are denoted by the costate vector p , with elements p_1 , p_2 , and p_3 , which allows us to write the Hamiltonian as

$$H(x, p) = p_1(-v_e + v_p \cos(2\pi x_3)) + p_2(v_p \sin(2\pi x_3)) - \bar{d} \|p_1 x_2 - p_2 x_1 - \frac{p_3}{2\pi}\| + \bar{u} \frac{p_3}{2\pi}. \quad (3.18)$$

Following the derivations of the previous section, this allows us to solve the resulting quasi-variational inequality, producing the final value function shown in Figure 3.1. For each dimension, 21 evenly spaced grid points are used and the quasi-variational inequality (3.6) is solved until $t_f = 1$ using a 3rd order essentially non-oscillatory (ENO) scheme together with a Lax Friedrichs Hamiltonian [110]. For the computation of the value function, we set $v_e = v_p = 0.75$, and $\bar{u} = \bar{d} = 3$. Furthermore, we set $r_1 = 0.25$ and $r_2 = 1$.

3.4.2 Gaussian process error prediction

Next, we fit a variety of regression models in order to approximate the numerical error $\tilde{\epsilon}$. To this end, we draw random samples of (x, t) , for $x \in [-1, 1] \times [-1, 1] \times [0, 1]$ and $t \in [-1, 0]$, and use these as initial conditions to compute the trajectory $\zeta(\tau; x, u(\cdot), d(\cdot), \forall \tau \in [-t, 0])$. The trajectories are then used to evaluate $\omega_{u, d}^{\sim}$, which in turn allows us to compute $\tilde{\epsilon}(x, t)$ as per Equation (3.11). For the comparison of the regression models, we train all models based on 1000 observations of (x, t) . To assess

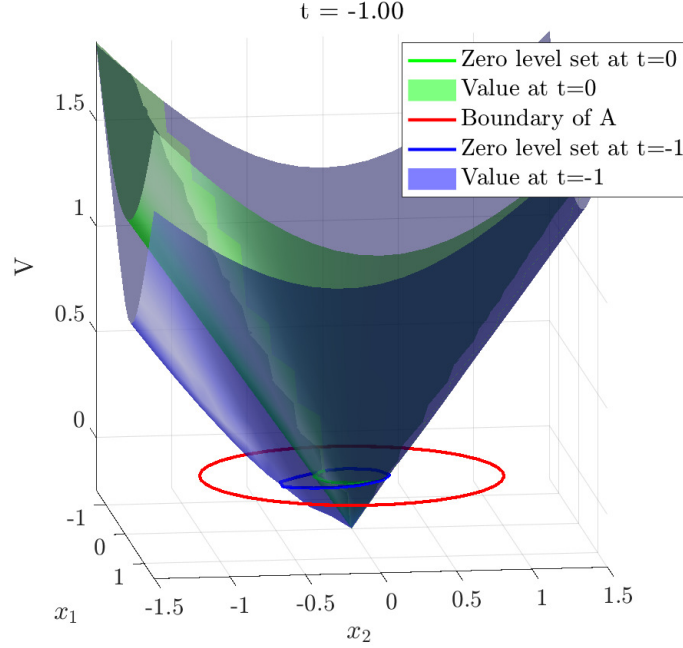


Figure 3.1: Value function of the pursuit-evasion example in the x_1 - x_2 plane sliced along $x_3 = 0.25$. The sub-zero level set (within the dark blue line) indicates the BRT. The boundary of the avoid set is shown by the dark red line.

the performance of the trained models, we use 5-fold cross-validation to minimise the RMSE of $\tilde{\epsilon} - \hat{\epsilon}$, where $\hat{\epsilon}$ is the output of the regression model. In Table 3.1 the RMSE of different regression models is shown. From the comparison of the regression models, one can see that GPR models lead to the smallest root mean squared error.

In Figure 3.2 the prediction capability of the rational quadratic GPR using a constant basis function is shown. For comparison, we also show the prediction capabilities of an identical regression model trained with 100, instead of 1000 observations. These two models will be referred to as the low-fidelity and the high-fidelity model, respectively. The validation set used in Figure 3.2 consists of 100 samples taken at random from (x, t) , as with the training set. For each sample we look up the corresponding value from the computed value function, $\tilde{\omega}$ (green points), calculate the *true* value, $\omega_{u, \tilde{d}}$, as in (3.9) (red points), and compare this to the predicted value, $\hat{\omega}$ (blue points), obtained by subtracting the GPR predicted error, $\hat{\epsilon}$, from the computed value function value. As can be seen, the prediction capabilities improve with a larger training set and perform at a satisfactory level

Regression Type	RMSE
Linear regression	0.043205
Tree	
Fine tree	0.031947
Medium tree	0.034466
Coarse tree	0.037971
Support vector machine	
Linear SVM	0.046966
Quadratic SVM	0.046893
Cubic SVM	0.046067
Gaussian SVM	0.045817
Gaussian process regression	
Squared exponential GPR	0.027390
Matern 5/2 GPR	0.026955
Exponential GPR	0.027353
Rational quadratic GPR	0.027029

Table 3.1: Comparison of commonly used regression models to approximate the numerical error of the value function.

in predicting the *true* value of the value function. Furthermore, even though the error does not necessarily follow a normal distribution, the 95% prediction interval is able to sufficiently capture the possible range the *true* value might take. Thus using the standard deviation as a confidence metric looks promising. Based on the successful prediction capabilities, we will subsequently study how well the trained regression model can improve the control policy.

Since we want to adhere to any memory constraints, we will not improve the fidelity of the value function and for every point on the grid, we simply compute the estimated error, $\hat{\epsilon}$, and subtract it from the value obtained by solving the quasi-variational inequality, $\tilde{\omega}$. The original, $\tilde{\omega}$, (green) and updated value function, $\hat{\omega}$, (blue) using the two GPRs trained previously are shown in Figure 3.3. Using the updated value function we can modify the control and disturbance policy by recomputing the minimisers/maximisers of the Hamiltonian. This results in the control and disturbance policies $\hat{u}$ and $\hat{d}$ respectively.

To evaluate the effectiveness of updating the control and disturbance policies, we calculate the RMSE of $\tilde{\omega}(x, t) - \omega_{\tilde{u}, \tilde{d}}$ as well as $\hat{\omega}(x, t) - \omega_{\hat{u}, \hat{d}}$ using a validation set

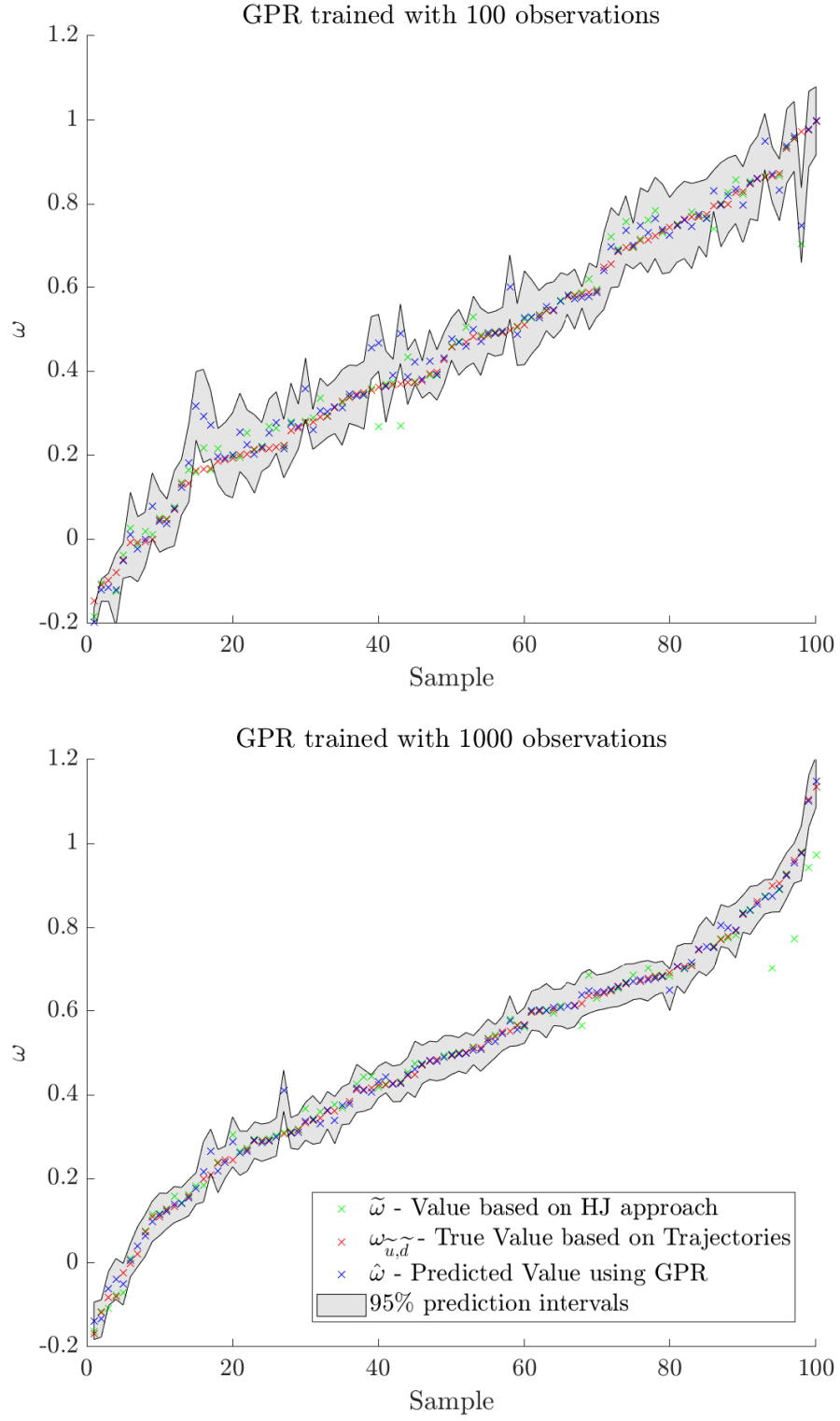


Figure 3.2: The prediction capabilities of the rational quadratic GPR model trained using 100 and 1000 observations respectively.

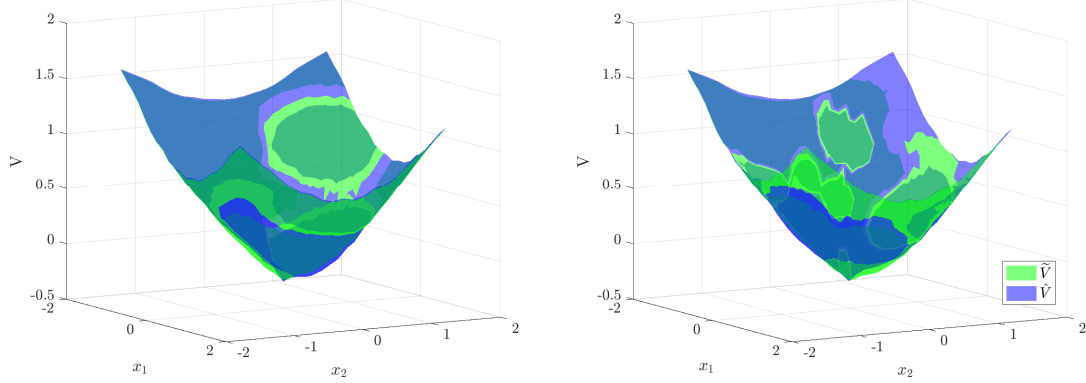


Figure 3.3: The correction capabilities of the rational quadratic GPR model trained using 100 (left) and 1000 (right) observations, respectively. As is expected, the regions of mismatch become more distinguished as we increase the number of observations.

	low-fidelity model	high-fidelity model
$\tilde{\omega}(x, t) - \omega_{\tilde{u}, \tilde{d}}$	$\hat{\omega}(x, t) - \omega_{\hat{u}, \hat{d}}$	$\hat{\omega}(x, t) - \omega_{\hat{u}, \hat{d}}$
0.0481	0.026	0.016

Table 3.2: Comparison of the ability to correct the value function.

consisting of 1000 samples. As can be seen in Table 3.2, both GPR models reduce the error. Thus helping increase the accuracy of the safety-critical controller significantly.

3.4.3 Effects of mismatch in dynamics

Next, we investigate the capabilities of detecting the error of the value function when the system dynamics of the vehicles are changed. To this end, we vary v_p and v_e and for each variation, retrain the GPR model. Conventionally, such changes to the dynamics would require a complete recomputation of the value function. However, as can be seen in Figure 3.4, minor changes to v_p and v_e from the nominal values of $v_e = v_p = 0.75$ do not necessarily lead to a large increase in the RMSE of the GPR model ($\tilde{\epsilon} - \hat{\epsilon}$). Naturally, as is to be expected, very large changes in the dynamics, such as doubling the speed of both the evader and pursuer will require a recomputation of the value function, since the GPR model struggles to correctly predict the error.

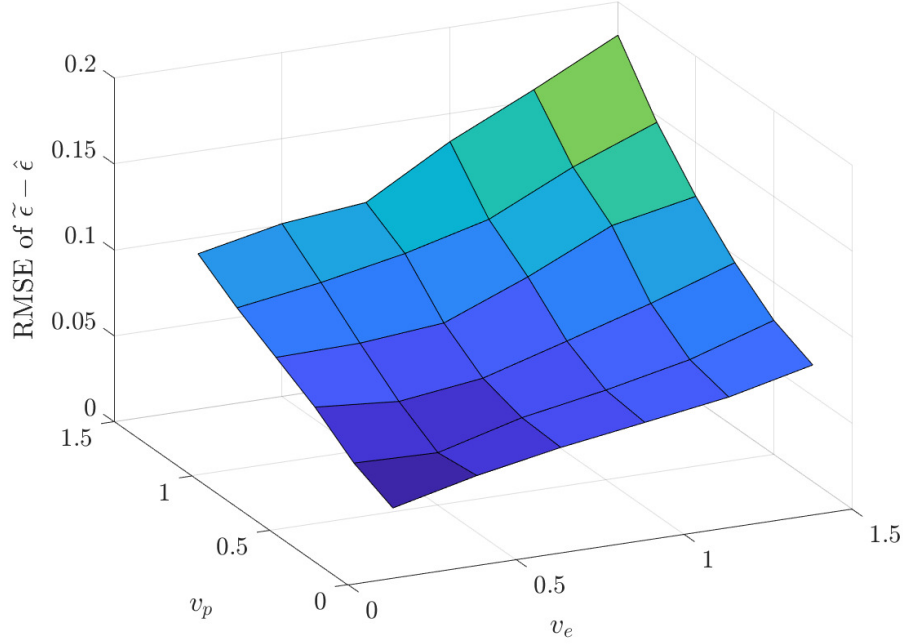


Figure 3.4: The RMSE of a GPR model trained using 1000 samples as we vary v_e and v_p .

3.5 Conclusion

We propose an empirical approach for correcting the prediction capabilities of the value function inherent in HJ reachability analysis. Utilising the confidence metric of the estimated error, we propose an improved manoeuvrability envelope for a least restrictive control law. Furthermore, using the prediction capabilities, we are able to update the value function, thus correcting for numerical errors and changes in the dynamics, environment, or disturbances without re-solving the quasi-variational inequality associated with HJ reachability.

4

Safety-Aware Hybrid Control of Airborne Wind Energy Systems

4.1 Introduction

With the looming threat of the climate crisis becoming ever more detrimental, the need for low-cost and reliable sources of energy becomes increasingly pressing. Wind remains the primary non-hydro renewable and to reach net-zero energy targets of 8008 TWh in 2030, will need to grow annually by 18% between 2021 and 2030 [79]. This has led to a rapid increase in research and development into Airborne Wind Energy (AWE) from both academia and the private sector [43, 128].

In this chapter, we only consider Ground-Gen AWE systems, whereby energy is obtained by continuously performing two phases of flight, a *traction* phase, and a *retraction* phase. During the traction phase, a tethered kite or fixed-wing aircraft is flown in crosswind conditions at altitudes of up to 1000 m above ground. The traction force acting on the aircraft's tether is converted into electricity using a generator and winch located at the base of the tether [41, 59]. For a discussion of alternative AWE systems, we refer to [41, 142]. Since the tether is gradually reeled out during the traction phase, the system will eventually need to be reset. For this, the aircraft is flown back to its starting altitude upwind in a phase commonly referred to as the retraction phase. During the retraction phase, a small amount

of the previously generated energy needs to be spent on reeling in the tether. Nevertheless, AWE systems have been shown to be able to create a high net energy output, with SkySails' recent uninterrupted automated flight of 42 hours producing 62 kW of average power and average power over the pumping cycle of approximately 92 kW [60], while larger systems aim at producing energy in the range of MWs [49].

For AWE systems to become widely adopted, it is necessary to provide guarantees on their safe operation and control. While recent research has addressed the reliability and safety of AWE [124], much of the literature, particularly on the control of AWE, focuses on performance optimisation [55, 57, 88, 93, 153]. The aim of this chapter, in contrast, is to consider the *classical* notion of safety as found in the control theory literature and ensure the safety of an AWE system while maintaining optimal power generation abilities. To this end, we propose a hybrid control law, whereby a safety controller is only activated when a critical system failure is predicted to occur. Incorporating safety by these means, effectively defining an avoidance manoeuvre, has become a common practice in the safety-critical control literature [65], and has been effectively demonstrated in the context of AWE in [117]. In [117] the authors use NDI, utilising a tether force set point during the dynamic inversion in conjunction with a winch control tether force tracking strategy. Using a predictor, such as a support vector machine, trained to detect upset conditions, an avoidance strategy is derived by means of reducing the tether force set point.

In contrast to the empirical avoidance strategy proposed in [117], we present a safety controller derived using reachability analysis, thus guaranteeing the satisfaction of hard constraints encoding safety. The proposed avoidance manoeuvre is derived independently of the chosen winch strategy and incorporates uncertainty in, e.g. wind conditions, through the incorporation of adversarial disturbances in the control design.

We consider an AWE setup similar to that proposed in [115], whereby a fixed-wing aircraft is flown in a figure of eight flight pattern, controlled using NDI. While further extensions to the simple NDI setup can be included to assist safety, such as dealing with model uncertainty by means of incremental nonlinear dynamic inversion [90],

or the avoidance strategies proposed in [117], for the sake of demonstrating the effects of our proposed avoidance manoeuvres, such extensions are omitted. To set the control problem, we consider an NDI controller that uses pseudo control inputs to choose the bank angle and angle of attack so as to track a desired course and path angle for optimal tracking of a figure eight flight path. Wind gusts and overly aggressive steering of the aircraft can lead to high stresses on the tether, causing it to rupture. As shown in Figure 4.1, with the underlying NDI controller, the tether ruptures before the aircraft can complete one entire cycle of flight. Using Hamilton-Jacobi (HJ) reachability analysis, we will synthesise a safety controller as well as switching conditions, such that in a hybrid control setup, minimal penalties on power generation are incurred, while ensuring a safe flight without a tether rupture.

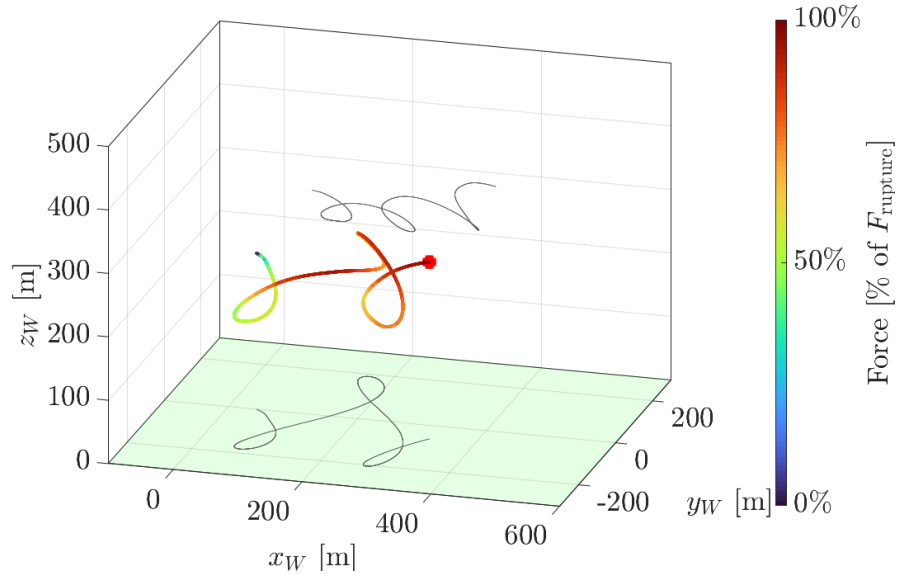


Figure 4.1: Tether rupture during the traction phase. The red dot marks the point during flight at which a tether rupture occurs due to the maximum tether force being exceeded.

HJ reachability analysis has become a well-adopted formal verification method, whereby the optimal controller is derived through computation of the backwards reachable set (BRS). Its advantages include compatibility with nonlinear system dynamics, formal treatment of nonlinear bounds, and recently also with multi-objective optimisation problems [13, 147]. However, since HJ reachability analysis

requires solving a quasi-variational inequality over a gridded state-space, it is subject to the curse of dimensionality. As such, its applications have been limited to low-dimensional systems [39]. To this end, we will derive a novel low-dimensional formulation of the system dynamics of AWE systems and formulate the safety-critical control problem as a differential game of two players. Furthermore, the aforementioned gridding of the state space necessitates a coordinate transformation that ensures only relevant states are considered for controller synthesis, minimising numerical overhead. Thus the key contributions of this chapter are:

1. The development of a simplified low dimensional AWE model that is accurate enough for safety-critical control purposes (Section 4.3);
2. The synthesis and deployment of an avoidance control strategy based on the developed abstractions and HJ reachability analysis, a non-trivial application for which we have made the code and simulation environment available (Section 4.4). This sets the framework for the development of a hybrid control law that allows switching between any performance-oriented controller and the developed safety controller;
3. The computation of the optimal control/disturbance actions that optimise the Hamiltonian of the underlying optimal control problem, which is case-dependent and non-trivial in the setting (Section 4.4.2);
4. The validation of the performance of the controller by applying it to a high-fidelity model, thus rendering our controller as an add-on to existing tools to guarantee safety (Section 4.5).

The remainder of the chapter is organised as follows: Section 4.2 discusses the AWE model used for simulation purposes as well as the guidance strategy employed by the primary controller. Section 4.3 introduces the simplifications, abstractions, and the new reference frame used to derive a low dimensional model of the AWE system suitable for safety-control synthesis, with the safety controller being derived and introduced in Section 4.4. Finally, in Section 4.5 we validate the avoidance control

strategy in its hybrid control setup using the high fidelity AWE model and discuss the effects on power generation with Section 4.6 providing concluding remarks.

4.2 Modelling

The modelling of AWE systems has been well studied and a variety of definitions for the equations of motion have been derived. Yet commonly AWE systems require a high number of states to accurately capture both the dynamics of the kite or aircraft as well as the tether and the associated winch controller. Since the goal of this chapter is to derive a safety-critical controller for the AWE system using HJ reachability analysis, we seek the development of a simplified yet accurate enough, from a control point of view, model.

In the following section, we begin by modelling the AWE system based predominantly on the work of [115] with extensions taken from [96, 116] and [61]. This model will be used for simulation purposes in Section 4.5 and is presented to provide the necessary notation and understanding needed for the abstractions and synthesis, the primary contributions of this work, in Sections 4.3 and 4.4.

4.2.1 Reference Frames

For the discussion of the dynamics of AWE systems, it is useful to introduce a variety of reference frames. The utilised reference frames (with the exception of the Γ frame, Section 4.3.3) are common for AWE systems and have been extensively discussed in [115] and [80]. For the transformation from one reference frame to another, we adopt the transformation matrices $\mathbf{M}_{(\cdot)(\cdot)}$, where the first subscript indicates the destination frame and the second subscript indicates the origin frame. The transformation matrices for the reference frames are listed in the appendix. In this chapter, we utilise bold notation for vectors and matrices and regular notation for scalars. This will ease notation when referring to the magnitudes of vectors. Furthermore, we utilise subscripts to denote vector elements, i.e., $(v_k)_{\tau,y}$ denotes the y component of the kinematic velocity in the τ frame.

Wind Frame

The wind frame (W) is a rotation of the commonly used North-East-Down (NED) (or O) frame. The x-axis of the W frame is aligned with the mean wind direction (denoted by ϖ). We assume that the mean wind direction is such that the z-axis of the W frame points upwards and the y-axis forms a right-hand reference frame (Figure 4.2).

Tangential Frame

The tangential frame (τ) is centred at the position of the aircraft, such that the z-axis points towards the origin of the NED frame. To this end, the x-axis points towards the north direction, while the y-axis completes a right-hand reference frame. Since the τ frame moves together with the aircraft, its location is determined using the longitude (λ) and latitude (ϕ) as well as the distance to the origin (h_τ). The tangential frame is shown in purple in Figure 4.2.

Aerodynamic Frame

The aircraft's kinematic velocity is denoted by $\mathbf{v}_k$ and is composed of the aerodynamic velocity $\mathbf{v}_a$ as well as the wind velocity $\mathbf{v}_w$, i.e., $\mathbf{v}_k = \mathbf{v}_a + \mathbf{v}_w$. This relationship holds true regardless of reference frame. For a discussion of the aerodynamic velocity, we define the aerodynamic frame (A) such that the velocity vector $\mathbf{v}_a$ is aligned with the x-axis. For the derivation with respect to the NED frame (shown in blue in Figure 4.2), we first rotate the NED frame by the course angle χ_a and path angle γ_a , resulting in the intermediate frame $\bar{A}$ (shown in orange in Figure 4.2), before tilting the frame by the bank angle μ_a . The A frame is shown in green in Figure 4.2.

Body-Fixed Frame

The body-fixed frame (B) is used to calculate the aerodynamic forces acting on the aircraft. It can be obtained from the aerodynamic frame (shown in green in Figure 4.2) using the angle of attack α_a as well as the sideslip angle β_a (the sideslip angle is neglected in this work). The B frame is shown in red in Figure 4.2.

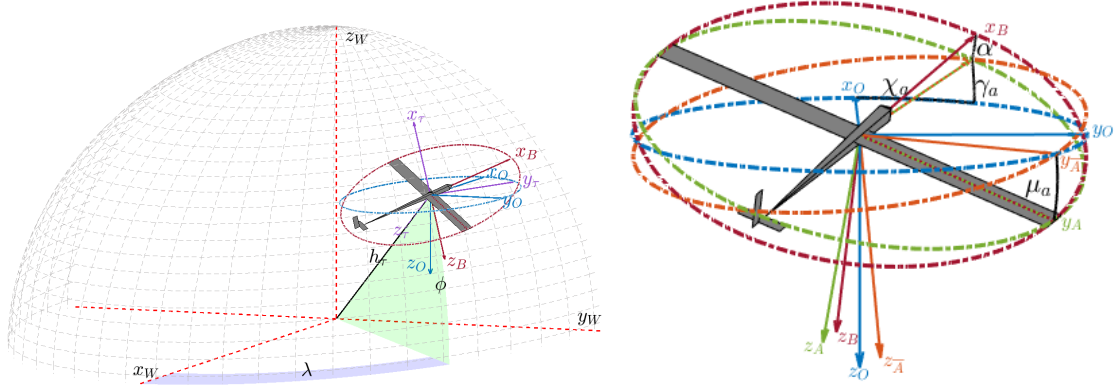


Figure 4.2: Visualisation of the NED frame (blue), intermediate aerodynamic frame $\bar{A}$ (orange), the aerodynamic frame A (green), the body frame B (red) and the tangential frame (purple).

4.2.2 Aircraft Equations of Motion and Ground Station Model

Since we are considering an avoidance manoeuvre for the path-following controller, for the purpose of controller synthesis, we will assume perfect tracking of the attitude and rate controllers [90, 115]. This simplification, common during the design of the path-following controller, allows us to analyse the performance of the avoidance manoeuvre independent of the effects of the inner control loop. To this end, we can neglect the yaw, pitch, and roll rates as well as the sideslip angle β_a , and define our control inputs as the bank angle μ_a as well as the angle of attack α_a .

To provide a realistic conclusion on overall feasibility, we will include actuator dynamics and delays in the final simulation to represent tracking delays of the inner loop. Further validation before deployment can be performed by integrating the path-following controller and avoidance manoeuvre into a full 6 DOF simulation, however, this is beyond the scope of this work.

The position of the aircraft is defined in the W frame by $[\lambda, \phi, h_r]$ and it moves with velocity v_a along the x-axis of the $\bar{A}$ frame. To connect the $\bar{A}$ frame with the NED frame, we add the course angle χ_a and path angle γ_a as additional states. To influence the heading of the aircraft, we utilise the bank angle μ_a and angle of attack α_a as inputs to the system and assume that the sideslip angle β_a is negligible, i.e., $\beta_a = 0$.

Therefore our initial set of states used are $[\lambda, \phi, h_\tau, v_a, \chi_a, \gamma_a] \in \mathbb{R}^6$. The position propagation can be calculated from the kinematic velocity in the τ frame as follows:

$$\dot{\lambda} = \frac{(v_k)_{\tau,y}}{\cos(\phi)h_\tau}, \quad \dot{\phi} = \frac{(v_k)_{\tau,x}}{h_\tau}, \quad \dot{h}_\tau = -(v_k)_{\tau,z}, \quad (4.1)$$

where $(v_k)_{\tau,x}$, $(v_k)_{\tau,y}$ and $(v_k)_{\tau,z}$ denote the x, y and z components of the kinematic velocity in the τ frame, respectively. $(\mathbf{v}_k)_\tau$ is calculated by transforming the kinematic velocity from the O frame to the W frame, and then to the τ frame using the appropriate transformation matrix, $\mathbf{M}_{WO}$ and $\mathbf{M}_{\tau W}$, respectively, found in the appendix. $(\mathbf{v}_k)_O$ is derived from the aerodynamic and wind velocity $(\mathbf{v}_k)_O = \mathbf{M}_{OA} [v_a, 0, 0]_A^T + (\mathbf{v}_w)_O$.

By assuming the wind field stays stationary, we are able to calculate the derivative of the kinematic velocity using the gravitational force $(\mathbf{F}_g)$, aerodynamic force $(\mathbf{F}_a)$, as well as the force the tether exerts on the aircraft $(\mathbf{F}_t)$. For a complete derivation, we refer to [115].

$$\begin{bmatrix} \dot{v}_a \\ \dot{\chi}_a \\ \dot{\gamma}_a \end{bmatrix} = \frac{1}{m_a} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \frac{1}{v_a \cos \gamma_a} & 0 \\ 0 & 0 & -\frac{1}{v_a} \end{bmatrix} \left(\mathbf{M}_{AO} [(\mathbf{F}_g)_O + (\mathbf{F}_t)_O] + \mathbf{M}_{AA} \mathbf{M}_{AB} (\mathbf{F}_a)_B \right), \quad (4.2)$$

where m_a denotes the mass of the aircraft. Note that the matrix $\mathbf{M}_{AO}$ depends on the course and path angles χ_a and γ_a , $\mathbf{M}_{AA}$ depends on the bank angle μ_a and $\mathbf{M}_{AB}$ depends on the angle of attack α_a and sideslip angle β_a .

Ground Station Model

The ground station serves as the power generator for the AWE system. The winch controller within the ground station is responsible for determining an appropriate torque so as to reel out or reel in the tether during flight. Various reeling strategies exist in the literature with the most common focusing on aligning the reel-out speed to one-third of the wind speed [89] or less when considering the retraction phase [92]. This is done in practice by either setting the torque proportional to the reel-out velocity squared [18, 137, 157] or by using reference tracking [55].

Since the winch will be modelled as an adversarial disturbance for controller synthesis, the reeling strategy can be chosen so as to determine tighter bounds

on the possible range of disturbances, by, e.g., restricting the tether to only be reeled out using a positive torque. To demonstrate the efficacy of the avoidance strategy derived in the sequel, however, we do not wish to limit the disturbance range by these means and instead account for a wider range of winching strategies, i.e., including the possibility of reeling in the tether during traction. To this end, a tether set point tracking strategy [118] is employed.

Gravitational and Aerodynamic Forces

The gravitational force can simply be expressed in the O frame as $(\mathbf{F}_g)_O = [0, 0, m_a g]^T \in \mathbb{R}^3$, where g is the gravitational constant. Using the aerodynamic coefficients, as derived in [96], we can then calculate the aerodynamic force of the aircraft as $(\mathbf{F}_a)_B = \frac{1}{2} \rho S_{\text{ref}} v_a^2 [C_x, C_y, C_z]^T$, where S_{ref} is the aerodynamic reference area corresponding to the projected surface area of the aircraft wing, and ρ is the air density.

Tether Forces

For the calculation of the tether forces $(\mathbf{F}_t)$ that act on the aircraft, we use models of varying complexity. For the controllers, we rely on a straight tether approximation, whereas for simulation purposes, we use a tether model similar to the model derived in [61]. We begin by presenting a brief overview of the lumped-mass tether model, as it provides the foundation for the simplified model used for controller synthesis, and is also used for the full simulation in Section 4.5

Let us first consider a fixed number of lumped masses connected by n spring-damper elements. The length of each segment is denoted by l_s and the spring and damping constants for each segment are denoted by k and c , respectively. Each tether segment is modelled as a point mass with position $\mathbf{p}_i$ and velocity $\mathbf{v}_i$, as shown on the left of Figure 4.3. The equations of motion of the tether segment are given by $\dot{\mathbf{p}}_i = \mathbf{v}_i$, and $\dot{\mathbf{v}}_i = m_t \mathbf{F}_i$, where m_t is the mass of an individual segment and $\mathbf{F}_i$ is the tether segment force. The tether segment force for segment i is given by $\mathbf{F}_i = \mathbf{F}_{s,i+1} - \mathbf{F}_{s,i} - \mathbf{F}_{g,s} + \mathbf{F}_{a,i}$, where $\mathbf{F}_{g,s}$ is the gravitational force acting on the tether segment, $\mathbf{F}_{a,i}$ is the aerodynamic drag acting on the i -th

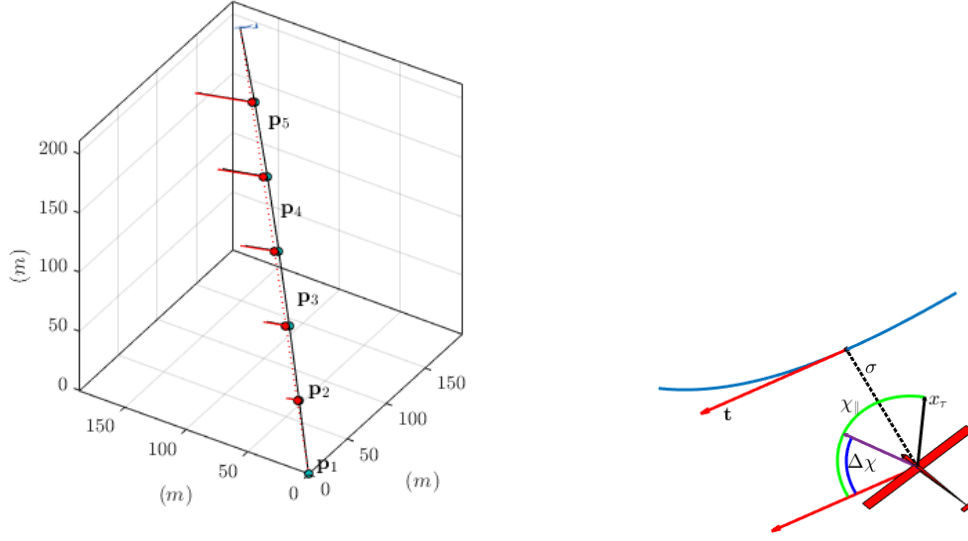


Figure 4.3: (left) Visualisation of the tether modelled as a straight tether (red) and as a fixed number of lumped masses (blue). The velocities of the point masses are shown by the red and blue vectors respectively.

(right) Visualisation of the guidance strategy to produce the optimal course heading (purple arrow).

segment, and $\mathbf{F}_{s,i}$ is the tensile force of the i -th segment. The tensile force is calculated according to Hooke's law

$$\mathbf{F}_{s,i} = \left(k(\|\mathbf{s}_i\|_2 - l_s) + c\left(\frac{\mathbf{s}_i}{\|\mathbf{s}_i\|_2} \mathbf{s}_{v,i}\right) \right) \frac{\mathbf{s}_i}{\|\mathbf{s}_i\|_2}, \quad (4.3)$$

where $\mathbf{s}_i = \mathbf{p}_i - \mathbf{p}_{i-1}$ and $\mathbf{s}_{v,i} = \mathbf{v}_i - \mathbf{v}_{i-1}$. Thus the maximum tether force is exerted on the final segment attached to the aircraft, i.e., $\mathbf{F}_t = \mathbf{F}_{s,n+1}$, with $\mathbf{s}_{n+1} = \mathbf{p}_{\text{aircraft}} - \mathbf{p}_n$ and $\mathbf{s}_{v,n+1} = \mathbf{v}_{\text{aircraft}} - \mathbf{v}_n$ and $\mathbf{p}_{\text{aircraft}}$ and $\mathbf{v}_{\text{aircraft}}$ are the position and velocity, respectively, of the aircraft in Cartesian coordinates. For the tether drag calculation, we refer to [61]. Since the tether can only exert a pulling force on the aircraft, the spring and damping constants are set to zero whenever $\|\mathbf{s}_i\|_2 - l_s < 0$.

This complex tether model requires $6n$ additional states, where n is the number of segments used, and is therefore only suited for simulation purposes. For the controller synthesis we use a simplified tether model that relies on a straight tether approximation and is introduced in Section 4.3.1.

Wind Field Model

To capture the varying wind speeds at different altitudes we employ the wind shear model provided by the MATLAB Aerospace Toolbox [101]. In addition to the wind shear, we also model atmospheric turbulence in the form of a continuous Dryden wind turbulence model as specified in Military Handbook MIL-HDBK-1797B [140], utilising the continuous Dryden model block provided by the MATLAB Aerospace Toolbox [101]. Finally, the wind velocity acting on the aircraft is computed by adding the turbulence to the wind shear $(\mathbf{v}_W)_O = (\mathbf{v}_{\text{turb}})_O + (\mathbf{v}_{\text{shear}})_O$.

4.2.3 Optimal Flight Path and Guidance

The fundamental idea behind Ground-Gen AWE is to transfer the force from the aerodynamic lift of the aircraft or kite to the connected tether [47]. Similar to the blades of a wind turbine, this can be done by moving the aircraft perpendicular to the mean wind direction. To this end, it has become common practice in the AWE community to adopt a figure eight flight pattern. Various methods have become well-established in the AWE control community to track such a flight pattern. In [55, 59] it is shown that two target points can suffice to generate the alternating left and right turns. In [54], the turn rate law is introduced, which is demonstrated in the SkySails control structure [53]. For the desired flight path of the safety controller, the choice of guidance strategy only affects the choice of the initial target set. However, the target set needs to be defined in relation to the chosen state-space.

Most crucial, however, is that the flight path of the safety avoidance manoeuvre does not overly upset the primary controller flight path, as this would impede on power performance. We thus adopt a similar guidance strategy as employed by the chosen primary controller [118], which is based on guidance law presented in [80].

For the chosen flight path, we use a Lissajous curve, $\mathbf{\Gamma}$, lying on a sphere. The two-dimensional curve can be described by its longitude λ_{Γ} and latitude ϕ_{Γ} . We parameterise $\mathbf{\Gamma}$ on $\mathbb{S}^2$ using the arc length s . In Cartesian coordinates the

curve is then given by

$$\mathbf{\Gamma}_P(s) = \begin{bmatrix} \cos \lambda_\Gamma(s) \cos \phi_\Gamma(s) \\ \sin \lambda_\Gamma(s) \cos \phi_\Gamma(s) \\ \sin \phi_\Gamma(s) \end{bmatrix} h_\tau. \quad (4.4)$$

For the specific Lissajous curve, we opt for the Lemniscate of Booth [118], a commonly used figure eight curve, using the height/width ratio of $a_{\text{Booth}}/b_{\text{Booth}} = 120/200$. The derivation of the longitude and latitude of the Lemniscate of Booth can be found in [115]. We define the reference curve to lie centred on the horizon. However, during flight, we rotate the curve by ψ_0 to assist with power generation. Thus the tracking curve used during flight is given by $\mathbf{\Gamma}(s, \psi_0) = \mathbf{M}_{\text{WP}} \mathbf{\Gamma}_P(s)$. Finally, the tangent of the curve is given by $\mathbf{t}(s) = \frac{d\mathbf{\Gamma}}{ds} = \frac{\partial \mathbf{\Gamma}}{\partial \lambda_\Gamma} \frac{\partial \lambda_\Gamma}{\partial s} + \frac{\partial \mathbf{\Gamma}}{\partial \phi_\Gamma} \frac{\partial \phi_\Gamma}{\partial s}$.

As shown in Figure 4.6, the shortest path from the position of the aircraft (denoted by $\mathbf{K}$) to the closet point on the tracking curve (denoted by $\mathbf{C}$), is given by the geodesic vector $\boldsymbol{\sigma}$. For perfect path tracking, we need to minimise $\|\boldsymbol{\sigma}\|_2$. Taking the shortest path, however, would result in the aircraft intercepting the curve perpendicularly. In practice, this is not desirable and thus we need to introduce a commanded flight direction that results in the distance to the tracking curve being minimised, while also ensuring that when the aircraft intercepts the tracking curve, its kinematic velocity is aligned with the tangent of the curve, $\mathbf{t}$. The commanded flight path can be defined on $\mathbb{S}^2$, independently of the distance to the origin by simply scaling the parameters a_{Booth} and b_{Booth} by $1/h_\tau$. The optimal course angle $(\chi)_\tau$ should point the aircraft along the geodesic when far from the curve, however, point perpendicular to the geodesic to align with $\mathbf{t}(s)$ when close to the curve. To remain on the tracking sphere, the optimal flight path angle is set to zero.

To this end, let us define the optimal path and course angles as $(\gamma_{\text{cmd}})_\tau = 0$ and $(\chi_{\text{cmd}})_\tau = (\chi_{\parallel})_\tau + (\Delta\chi)_\tau$. The first component of the command course angle is inspired by [59], i.e., $(\chi_{\parallel})_\tau = \arctan \frac{t_y}{t_x}$, where t_x and t_y denote the x and y components of the curve tangent. The second component of the command course angle, $\Delta\chi_\tau$, is chosen as in [118] to decrease as we approach the tracking curve. This tracking strategy is visualised on the right of Figure 4.3 and will be used to define the target set during the reachability calculation in Section 4.4.

4.3 Model Abstraction and Safety Considerations

For the derivation of the safety controller presented in Section 4.4, the previously introduced model does not suffice, since it requires too many states for synthesis. We thus begin by introducing a simplification of the tether model that allows us to introduce adversarial winch control into the AWE dynamics. Together with a new reference frame and the subsequent safety control model, this section provides the first of our four key contributions.

4.3.1 Simplified Tether Model

As mentioned in the previous sections, a full particle tether model requires too many states, prohibiting its adoption for the safety controller. Initial attempts at using simplified tether approximations, such as those derived in [36, 52, 55, 57], were found to be insufficient for the safety-control synthesis of fixed-wing aircraft as the assumption of the aircraft maintaining aerodynamic equilibrium is not necessarily valid during the avoidance manoeuvre. Thus an alternative approach to simplifying the complex tether model is introduced.

From field tests conducted with an aircraft in [80], for a small aircraft the tether forces far exceed the gravitational forces and a straight tether approximation is reasonable. However, if larger systems with larger tethers are employed, the tether sag will need to be taken into account. For the AWE system considered in this chapter, for the purpose of controller synthesis, we assume that the tether sag remains negligible and, therefore, opt to approximate the position and velocity of the tether segments. To this end, we assume that the position and velocity of the tether segments are distributed evenly along a straight line between the origin and the position of the aircraft. Thus the tether particle's position and velocity can be computed as the x , y , and z projections of the straight tether

up to the i -th particle, i.e.,

$$\begin{aligned}\mathbf{p}_i &= \frac{i h_\tau}{n+1} \begin{bmatrix} \cos(\phi) \cos(\lambda) \\ \cos(\phi) \sin(\lambda) \\ \sin(\phi) \end{bmatrix}, \\ \mathbf{v}_i &= \frac{i \dot{h}_\tau}{n+1} \begin{bmatrix} \cos(\phi) \cos(\lambda) \\ \cos(\phi) \sin(\lambda) \\ \sin(\phi) \end{bmatrix} - \frac{i h_\tau}{n+1} \begin{bmatrix} \sin(\phi) \dot{\phi} \cos(\lambda) + \cos(\phi) \sin(\lambda) \dot{\lambda} \\ \sin(\phi) \dot{\phi} \sin(\lambda) - \cos(\phi) \cos(\lambda) \dot{\lambda} \\ -\cos(\phi) \dot{\phi} \end{bmatrix}. \quad (4.5)\end{aligned}$$

On the left of Figure 4.3 a comparison of the straight tether and the full tether model is shown. Recall that the tether force acting on the aircraft is given by the spring force of the $(n+1)$ -th segment, i.e., $\mathbf{F}_t = \mathbf{F}_{s,n+1}$. The dominating factor in Hooke's law is given by the term $\|\mathbf{s}_{n+1}\|_2 - l_s$, which is the difference between the segment length and the distance between the final tether point mass and the aircraft. Having $\|\mathbf{s}_{n+1}\|_2 - l_s > 0$ implies the distance is greater than the length of the tether and thus the tether is under tension. A similar approach of modelling the tether by considering the effect of the tether elongation as the dominant term was recently also presented in [137]. However, in contrast to [137], for the proposed fixed-wing aircraft the damping term is not considered negligible. Modelling the elongation accurately is imperative for calculating the tether force, as the slightest deviations result in vastly inaccurate force calculations. To this end, let us introduce $\Delta_t = \|\mathbf{s}_{n+1}\|_2 - l_s$ as an additional state. By (4.3), the tether force acting on the aircraft can be calculated as

$$\mathbf{F}_t = \left(k \Delta_t + c \left(\frac{\mathbf{s}_{n+1}}{\|\mathbf{s}_{n+1}\|_2} \mathbf{s}_{v,n+1} \right) \right) \frac{\mathbf{s}_{n+1}}{\|\mathbf{s}_{n+1}\|_2}, \quad (4.6)$$

where the position and velocity of the tether point masses are computed using Equation (4.5). Since we wish to consider an upper bound on the tether force, we utilise an upper bound for the damping and stiffness coefficients based on the minimum expected tether segment length. This reduces complexity while adding robustness to uncertainty in the measured Δ_t .

The derivative of Δ_t is given by

$$\dot{\Delta}_t = \frac{\mathbf{s}_{v,n+1}^T \mathbf{s}_{n+1}}{\|\mathbf{s}_{n+1}\|_2} - \frac{\dot{l}_s}{n+1}. \quad (4.7)$$

To keep the number of system states minimal, we need to decouple the winch system from the aircraft dynamics. To this end, we model the effects of the winch as a

scalar disturbance, $d_{\Delta_t} := \dot{\Delta}_t$. By accounting for a worst-case winch behaviour, the system can be considered robust to arbitrary winch control. To compute the expected range of values that the derivative of Δ_t might take, we can consider the winch control strategy and derive bounds on the maximum/minimum reeling speed. However, to translate the bounds on the reeling speed to bounds on the derivative of Δ_t , we require also bounding the term $\mathbf{s}_{v,n+1}$. This would result in a coupling between the wind disturbance and the disturbances desired to model adversarial effects on Δ_t . Since combining the worst-case $\mathbf{s}_{v,n+1}$ with the worst-case $\dot{l}_s$, also leads to overly conservative bounds on $\dot{\Delta}_t$, an alternative empirical approach was adopted whereby the bounds are computed by means of simulation. To this end, we measure $\dot{\Delta}_t$ during simulated flight, as shown in Figure 4.4. Since $\dot{\Delta}_t$ never exceeds $0.0015 \frac{m}{s}$, we constrain the disturbance, d_{Δ_t} , to lie within the interval $[-d_{\Delta_t, \max}, d_{\Delta_t, \max}]$ with $d_{\Delta_t, \max} = 0.005 \frac{m}{s}$.

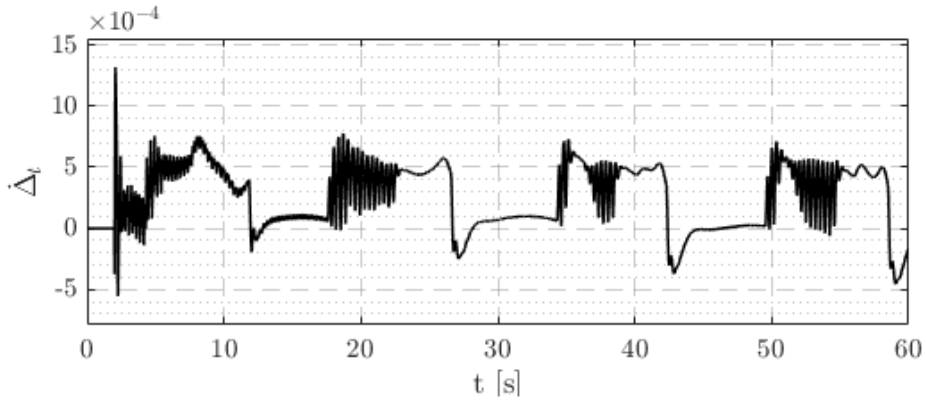


Figure 4.4: $\dot{\Delta}_t$ during the traction phase of flight, using the primary path-following controller.

Remark 4.1. *It is worth pointing out that determining the bounds on $\dot{\Delta}_t$ utilising true flight data in conjunction with methods such as the scenario approach [32] could potentially yield probabilistic guarantees for safety. This is, however, beyond the scope of this work.*

4.3.2 Adversarial Wind Turbulence

Since we model the wind turbulence as a stochastic process, we need to ensure that the safety controller can account for worst-case wind gusts, i.e., wind gusts that drive the system away from its target trajectory, and/or lead to a tether rupture. To this end, we model the wind turbulence as an adversarial disturbance input to the system. Thus the wind velocity for controller synthesis is given by $(\mathbf{v}_w)_O = (\mathbf{v}_{\text{shear}})_O + (\mathbf{d}_{\text{turb}})_O$, where $\mathbf{d}_{\text{turb}} \in \mathbb{R}^3$ is an additional disturbance vector. Similarly, any uncertainty in the wind shear can be included in this additive disturbance and will therefore not be treated separately.

In the same manner that we computed the bounds for d_{Δ_t} , we simulate the flight of the aircraft using the primary path-following controller and analyse the behaviour of the longitudinal, u_g , lateral, v_g , and vertical, w_g , turbulence velocity aligned along the horizontal relative mean wind vector, as shown in Figure 4.5. From the behaviour u_g , v_g and w_g , we can safely choose each element of $\mathbf{d}_{\text{turb}}$ to be bounded by $\pm 4 \frac{m}{s}$. To achieve these bounds in practice, we can use data-driven approaches with an appropriate estimator [18, 58, 129].

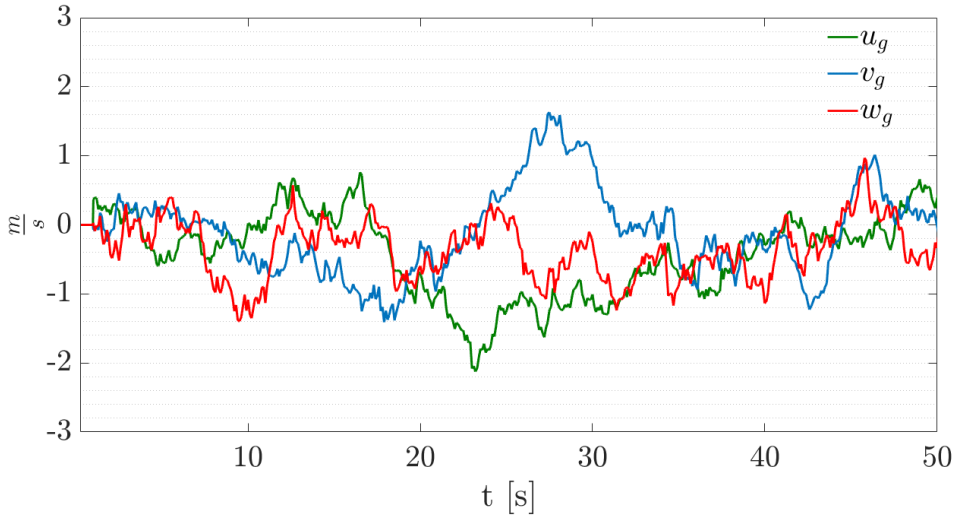


Figure 4.5: The behaviour of the wind gusts during 50 seconds of flight.

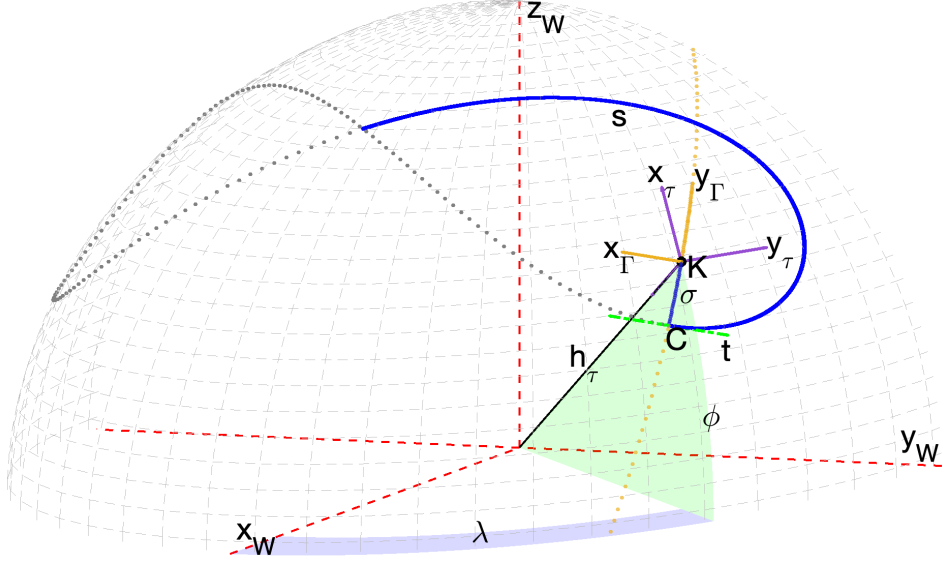


Figure 4.6: Visualisation of the Γ frame. s is used to denote the position along the reference curve. The geodesic vector, σ , describes the distance to the reference curve.

4.3.3 Γ Frame and Safety Control Model

As visualised in Figure 4.6, the geodesic vector pointing from $\mathbf{C}$ along the geodesic towards $\mathbf{K}$, is always orthogonal to $\mathbf{t}$. The derivative of Γ together with the direction of the geodesic vector can, therefore, be used as basis vectors to construct another reference frame that will become useful during gridding, as discussed in Section 4.4. This new reference frame will be referred to as the Γ frame. Any point on $\mathbb{S}^2$ defined using the longitude, λ , and latitude, ϕ , can also be defined using the Γ frame, where the first coordinate, s , describes an arbitrary point along the curve Γ , and the second coordinate $\sigma = \|\boldsymbol{\sigma}\|_2$ describes the geodesic distance in meters perpendicular to $\mathbf{t}(s)$. The calculation of the geodesic for a sphere is derived from [109, p. 216] and relies on solving the geodesic equations for a sphere.

$$\begin{aligned} \frac{\partial \lambda^2}{\partial t^2} - 2 \tan(\phi) \frac{\partial \lambda}{\partial t} \frac{\partial \phi}{\partial t} &= 0, \\ \frac{\partial \phi^2}{\partial t^2} + \sin(\phi) \cos(\phi) \left(\frac{\partial \lambda}{\partial t} \right)^2 &= 0. \end{aligned}$$

Transforming the equations to a 4-dimensional ordinary differential equation yields

$$\begin{aligned} x_1 &= x_2, \\ x_2 &= 2 \tan(x_3) x_2 x_4, \\ x_3 &= x_4, \\ x_4 &= -\sin(x_3) \cos(x_3) x_2^2, \end{aligned}$$

with $x_1 = \lambda$, $x_2 = \frac{\partial \lambda}{\partial t}$, $x_3 = \phi$ and $x_4 = \frac{\partial \phi}{\partial t}$.

The derivatives of s can be obtained by projecting the kinematic velocity of the aircraft onto $\mathbf{t}$. Similarly, the derivative of σ is obtained by projecting the kinematic velocity onto a perpendicular rotation of $\mathbf{t}$, denoted by $\mathbf{t}_\perp$.

$$\dot{s} = \frac{(\mathbf{t})_\tau (\mathbf{v}_k)_\tau}{\|\mathbf{t}\|_2 l_\Gamma}, \quad \dot{\sigma} = \frac{(\mathbf{t}_\perp)_\tau (\mathbf{v}_k)_\tau}{\|\mathbf{t}_\perp\|_2}, \quad (4.8)$$

where l_Γ denotes the arc length of the Lissajous curve Γ [11]. Using the Γ frame allows us to replace the longitude and latitude resulting in the final set of states used for controller synthesis

$$\mathbf{x} := [s, \sigma, h_\tau, v_a, \chi_a, \gamma_a, \Delta_t]^T \in \mathbb{R}^7. \quad (4.9)$$

The transformation between the NED frame and the Γ frame is not bijective, thus we need to take the velocity vectors into account to derive a deterministic transformation. To this end, we choose the transformation from the NED frame to the Γ frame, such that the error between $(\mathbf{v}_k)$ and $\mathbf{t}$ is minimised. Finally, we can summarise the equations of motion for the simplified aircraft model as $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}, \mathbf{d})$. The control inputs, $\mathbf{u}$, and disturbances, $\mathbf{d}$, are defined as

$$\mathbf{u} := \begin{bmatrix} \alpha_a \\ \mu_a \end{bmatrix} \in U \subset \mathbb{R}^2, \quad \mathbf{d} := \begin{bmatrix} d_{\Delta_t} \\ \mathbf{d}_{\text{turb}} \end{bmatrix} \in D \subset \mathbb{R}^4. \quad (4.10)$$

The aircraft dynamics are composed of $\hat{\mathbf{f}}$, affected only by the control inputs, and $\mathbf{f}_C(\mathbf{x}, \mathbf{d})$, affected only by the disturbances and incorporating all remaining additive terms, i.e., $\mathbf{f}(\mathbf{x}, \mathbf{u}, \mathbf{d}) = \hat{\mathbf{f}}(\mathbf{x}, \mathbf{u}) + \mathbf{f}_C(\mathbf{x}, \mathbf{d})$. If the disturbance does not appear in the dynamics in an affine manner [86, p. 133], the reachability problem remains solvable, however, a numerical minimiser of the Hamiltonian might become

necessary (e.g., by gridding the domain of the control inputs). Since this usually impedes on computational complexity, utilising the structure to develop analytic expressions for the optimisers is beneficial.

Decomposing the dynamics serves both the presentation of the final system dynamics, as well as the derivation of the Hamiltonian in the subsequent section. Since the control inputs affect only the aircraft velocity, course, and path angles, with a slight abuse of notation, we omit the four remaining state derivatives that are zero, allowing us to summarise $\hat{\mathbf{f}}(\mathbf{x}, \mathbf{u})$ as

$$\hat{\mathbf{f}}(\mathbf{x}, \mathbf{u}) = \frac{1}{m_a} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \frac{1}{v_a \cos \gamma_a} & 0 \\ 0 & 0 & -\frac{1}{v_a} \end{bmatrix} \mathbf{M}_{\bar{A}A} \mathbf{M}_{AB} (\mathbf{F}_a)_B. \quad (4.11)$$

The term $\mathbf{f}_C(\mathbf{x}, \mathbf{d})$ is then given by

$$\mathbf{f}_C(\mathbf{x}, \mathbf{d}) = \underbrace{\left[\frac{(\mathbf{t})_\tau (\mathbf{v}_k)_\tau}{\|\mathbf{t}\|_2 l_\Gamma}, \frac{(\mathbf{t}_\perp)_\tau (\mathbf{v}_k)_\tau}{\|\mathbf{t}_\perp\|_2}, -(v_k)_{\tau,z}, \frac{(F_t)_{\bar{A},x}}{m_a}, \frac{(F_t)_{\bar{A},y}}{m_a v_a \cos \gamma_a}, -\frac{(F_t)_{\bar{A},z}}{m_a v_a}, 0 \right]^T}_{=\mathbf{f}_{C1}(\mathbf{x}, \mathbf{d}_{\text{turb}})} \quad (4.12)$$

$$+ \underbrace{\left[0, 0, 0, 0, 0, 0, d_{\Delta_t} \right]^T}_{=\mathbf{f}_{C2}(\mathbf{x}, d_{\Delta_t})} \quad (4.13)$$

$$+ \underbrace{\left[0, 0, 0, \frac{(F_g)_{\bar{A},x}}{m_a}, \frac{(F_g)_{\bar{A},y}}{m_a v_a \cos \gamma_a}, -\frac{(F_g)_{\bar{A},z}}{m_a v_a}, 0 \right]^T}_{=\mathbf{f}_{C3}(\mathbf{x})}, \quad (4.14)$$

where we adopt the same notation as before to denote the x, y, and z components of the forces in the $\bar{A}$ frame and simplify the notation by omitting the dependence of $\mathbf{d}_{\text{turb}}$ and $\mathbf{x}$ in the tether and gravitational forces.

4.4 HJ Reachability Analysis and Controller Synthesis

4.4.1 Problem Statement

Having formulated the system dynamics, we are able to introduce the necessary concepts of HJ reachability analysis used for controller synthesis. The synthesised controller and subsequent hybrid control laws provide the second key contribution of this chapter.

Let $\mathcal{R}$ be the reach set, the set of states that should be reached in a given time, and let $\mathcal{A}$ be the set of avoid states, the set of states that lead to a critical system failure (i.e., tether rupture). Then we can define the backwards reachable set (BRS) as the set of states from which it is possible to reach the set $\mathcal{R}$ at the end of a given time interval with duration t_f while guaranteeing never to enter the set $\mathcal{A}$ until then. Mathematically, let $\mathbf{x} \in \mathbb{R}^7$ be the system state defined in (4.9) evolving according to the ordinary differential equation

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t), \mathbf{d}(t)), \quad t \in [-t_f, 0], \mathbf{u} \in \mathcal{U}, \mathbf{d} \in \mathcal{D}. \quad (4.15)$$

Note that we treat time as negative consistent with [13, 146]; this only simplifies some of the notation and implies that we start our system at time $-t_f$. The dynamics, $\mathbf{f}$, are assumed to be bounded and Lipschitz continuous in $\mathbf{x}$ and uniformly continuous in $\mathbf{u}$ and $\mathbf{d}$. Then, given the Lebesgue-measurable functions $\mathbf{u}(\cdot) \in \mathcal{U}$ and $\mathbf{d}(\cdot) \in \mathcal{D}$, the control and disturbance inputs, respectively, there exists a unique trajectory, ζ , solving (4.15), i.e.,

$$\begin{aligned} \frac{\partial}{\partial t} \zeta(t; \mathbf{x}_0, \mathbf{u}(\cdot), \mathbf{d}(\cdot)) &= \mathbf{f}(\zeta(t; \mathbf{x}_0, \mathbf{u}(\cdot), \mathbf{d}(\cdot)), \mathbf{u}(t), \mathbf{d}(t)) \quad \forall t \in [-t_f, 0] \\ \zeta(-t_f; \mathbf{x}_0, \mathbf{u}(\cdot), \mathbf{d}(\cdot)) &= \mathbf{x}_0 \end{aligned}$$

To capture the worst-case disturbance, we model the underlying control problem as a differential game of two players. Following [97], we restrict the first player to play a nonanticipative strategy [56, 141], which is a function $\xi : \mathcal{D} \rightarrow \mathcal{U}$, such that for all $t \in [-t_f, 0]$ and for all $\mathbf{d}, \hat{\mathbf{d}} \in \mathcal{D}$, if $\mathbf{d}(\tau) = \hat{\mathbf{d}}(\tau)$ for almost every $\tau \in [-t_f, t]$, then $\xi[\mathbf{d}](\tau) = \xi[\hat{\mathbf{d}}](\tau)$ for almost every $\tau \in [-t_f, t]$. Furthermore, we use Ξ to denote the class of nonanticipative strategies. Finally, we can define the BRS as

$$\begin{aligned} \text{BRS}_{\mathcal{A}, \mathcal{R}}(-t_f) &= \left\{ \mathbf{x} \in \mathbb{R}^7 \mid \exists \xi(\cdot) \in \Xi, \forall \mathbf{d}(\cdot) \in \mathcal{D}, \right. \\ &\quad \left. [\zeta(0; \mathbf{x}, \xi(\cdot), \mathbf{d}(\cdot)) \in \mathcal{R}] \ \& \ [\forall \tau \in [-t_f, 0], \zeta(\tau; \mathbf{x}, \xi(\cdot), \mathbf{d}(\cdot)) \notin \mathcal{A}] \right\}. \end{aligned} \quad (4.16)$$

In words, $\text{BRS}_{\mathcal{A}, \mathcal{R}}(-t_f)$ is the set of states from which trajectories can start at $-t_f$, and there exists a choice for a nonanticipative strategy ξ , such that for any

disturbance strategy $\mathbf{d}$, the system state can reach $\mathcal{R}$ at the end of the horizon while avoiding $\mathcal{A}$ until then.

By defining the avoid states as the set of states that imply a tether rupture, and the reach set as the set of states that follow the optimal guidance strategy, the BRS is able to provide information about an impending critical failure. Furthermore, by computing the BRS, we are able to simultaneously find the optimal trajectory and control policy that allows for optimal flight while avoiding critical states.

Recall that the maximum force acting on the tether is given by the final segment attached to the aircraft. Thus, by setting the maximum allowed tether force as F_{rupture} , we can define the Lipschitz continuous function $h(\mathbf{x}) = \|\mathbf{F}_t\|_2 - F_{\text{rupture}}$. Then we can define the avoid set related to the super-zero level set of $h(\cdot)$ as $\mathcal{A} = \{\mathbf{x} \in \mathbb{R}^7 | h(\mathbf{x}) > 0\}$.

Rather than defining our target set $\mathcal{R}$ as being the set of states that lie on the optimal tracking curve, we define our target set as all states that are aligned with the commanded flight direction. To this end, we define the Lipschitz continuous function $l(\cdot)$ by taking the signed distance functions of the course and path errors in the NED frame. Since the primary objective of the safety manoeuvre is not optimal path-following, but rather the avoidance of a tether rupture, the target set should simply favour the commanded course and path angles, rather than strictly enforcing equality. This is done by introducing a tolerance $\epsilon > 0$. The resulting signed distance function is $l(\mathbf{x}) = \max\{|\gamma_{\text{cmd},O} - (\gamma)_O|, |\chi_{\text{cmd},O} - (\chi)_O|\} - \epsilon$, where $(\gamma)_O$ and $(\chi)_O$ are the 5th and 6th states of our system transformed to the NED frame. The reach set is then defined by the sub-zero level set of $l(\cdot)$, i.e., $\mathcal{R} = \{\mathbf{x} \in \mathbb{R}^7 | l(\mathbf{x}) \leq 0\}$.

Using the definition of $\mathcal{A}$ and $\mathcal{R}$, as in [97], it can be shown that $\text{BRS}_{\mathcal{A},\mathcal{R}}(-t_f) = \{\mathbf{x} \in \mathbb{R}^7 | \omega(\mathbf{x}, t) \leq 0\}$, where

$$\omega(\mathbf{x}, t) = \inf_{\xi(\cdot) \in \Xi} \sup_{\mathbf{d}(\cdot) \in \mathcal{D}} \max \left\{ l(\zeta(0; \mathbf{x}, \xi(\cdot), \mathbf{d}(\cdot))), \max_{\tau \in [-t, 0]} h(\zeta(\tau; \mathbf{x}, \xi(\cdot), \mathbf{d}(\cdot))) \right\}. \quad (4.17)$$

Furthermore, as in [97], the value function in (4.17) is the unique continuous viscosity solution of the following quasi-variational inequality

$$\max \left\{ h(\mathbf{x}) - \omega(\mathbf{x}, t), \frac{\partial \omega(\mathbf{x}, t)}{\partial t} + \sup_{\mathbf{d} \in D} \inf_{\mathbf{u} \in U} \frac{\partial \omega(\mathbf{x}, t)}{\partial \mathbf{x}} f(\mathbf{x}, \mathbf{u}, \mathbf{d}) \right\} = 0, \quad (4.18)$$

with terminal condition $\omega(\mathbf{x}, 0) = \max\{h(\mathbf{x}), l(\mathbf{x})\}$ and where D and U are the compact sets of possible disturbance and control inputs, respectively, as defined in (4.10). For numerical reasons, it has become common practice to solve (4.18) using level set methods in combination with upwinding schemes such as ENO schemes [110, p. 29]. Alternative methods for solving (4.18), such as using Deep Learning [12] have been proposed, however, they lack many of the performance and safety guarantees we require.

To apply level set methods to (4.18), we are required to grid the state-space, thus exponentially scaling the memory requirements with each additional state. This necessitates the need for the low dimensional model of the AWE system derived earlier in Section 4.3. Furthermore, for each grid node, we are required to evaluate the dynamics and calculate the derivative of the value function. Therefore, to minimise computational overhead, we need to ensure we only consider grid nodes that are relevant during flight. To use conventional level set methods, however, we are required to use an evenly spaced grid. As can be seen in Figure 4.7, gridding the position of the aircraft in the NED frame is inefficient compared to the new Γ frame introduced in Section 4.3.3.

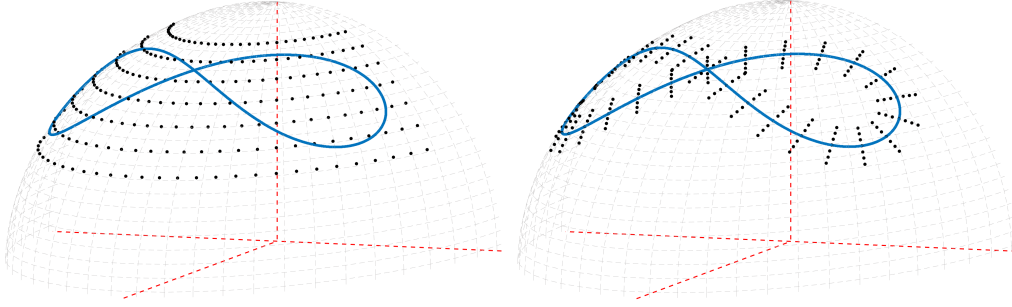


Figure 4.7: Comparison of a 31×7 grid in the NED frame vs the Γ frame. The Γ frame successfully captures only relevant nodes close to the optimal tracking curve.

4.4.2 Optimal Control and Disturbance Inputs

To solve (4.18), we need to find the optimal control and disturbance inputs that are the minimisers and maximisers, respectively, of $\sup_{\mathbf{d} \in D} \inf_{\mathbf{u} \in U} \frac{\partial \omega(\mathbf{x}, t)}{\partial \mathbf{x}} \mathbf{f}(\mathbf{x}, \mathbf{u}, \mathbf{d})$. Notice that the order of $\sup_{\mathbf{d} \in D} \inf_{\mathbf{u} \in U}$ has been swapped compared to the differential

game of the previous chapter. Due to the high inertia of the winch, we assume that the response time of the winch controller is slower compared to the flight controller and we also assume the wind gusts to change at a slower frequency compared to the choice of flight control actions. As such, in a differential game setup, it makes sense to allow the controller to choose its control actions after the disturbance.

Let us define the Hamiltonian of the system as

$$H(\mathbf{x}, \mathbf{q}) = \max_{\mathbf{d} \in D} \min_{\mathbf{u} \in U} \mathbf{q}^T \mathbf{f}(\mathbf{x}, \mathbf{u}, \mathbf{d}), \quad (4.19)$$

where $\mathbf{q} = [q_1, \dots, q_7] \in \mathbb{R}^7$ is the costate vector and since both D and U are compact, we are able to define the Hamiltonian using the max and min over D and U , instead of the sup and inf, respectively, i.e., the optimisers are achieved. To determine the optimisers of the Hamiltonian, we utilise the separation of the dynamics presented in Section 4.3.3. This allows us to write the Hamiltonian as

$$H(\mathbf{x}, \mathbf{q}) = \min_{\mathbf{u} \in U} \mathbf{q}^T \hat{\mathbf{f}}(\mathbf{x}, \mathbf{u}) + \max_{\mathbf{d} \in D} \mathbf{q}^T \mathbf{f}_C(\mathbf{x}, \mathbf{d}), \quad (4.20)$$

separating the control from the disturbance inputs. We begin by finding the control inputs that minimise the Hamiltonian

$$\begin{aligned} \mathbf{u}^* &\in \arg \min_{\mathbf{u} \in U} \mathbf{q}^T \hat{\mathbf{f}}(\mathbf{x}, \mathbf{u}) \\ &= \arg \min_{\mathbf{u} \in U} \frac{1}{m_a} \left[q_4 \quad \frac{q_5}{v_a \cos \gamma_a} \quad -\frac{q_6}{v_a} \right] \mathbf{M}_{\bar{A}A}(\mu) \mathbf{M}_{AB}(\alpha, \beta) (\mathbf{F}_a(\alpha, v_a))_B. \end{aligned} \quad (4.21)$$

Since the derived avoidance manoeuvre needs to adhere to control constraints placed on the path-following controller, notably limiting the range of bank angles, no analytic expression for (4.21) could be found. However, since we only need to know the optimal control inputs for a finite number of v_a, γ, q_4, q_5 , and q_6 values, we grid the input space and evaluate (4.21) for discrete values of μ and α to find the near-optimal control inputs, i.e.,

$$\begin{aligned} \mathbf{u}^* &\in \arg \min_{\substack{\alpha \in [\alpha_1, \dots, \alpha_n], \\ \mu \in [\mu_1, \dots, \mu_m]}} v_a q_4 (a(\alpha) \cos(\alpha) + b(\alpha) \sin(\alpha)) \\ &\quad + \left(\frac{q_5}{\cos \gamma_a} \sin(\mu) + q_6 \cos(\mu) \right) (b(\alpha) \cos(\alpha) - a(\alpha) \sin(\alpha)), \end{aligned} \quad (4.22)$$

where $a(\alpha)$ and $b(\alpha)$ are second order polynomials and $[\alpha_0, \dots, \alpha_n]$ and $[\mu_1, \dots, \mu_m]$ are discrete values of α and μ , respectively. Notice that u^* is a state-feedback controller as it depends on the state implicitly through the costate vector q .

Determining the control inputs that minimise the Hamiltonian by means of (4.22), we proceed to find the worst-case disturbances as the maximiser of the Hamiltonian, i.e., $\mathbf{d}^* \in \arg \max_{\mathbf{d} \in D} \mathbf{q}^T \mathbf{f}_C(\mathbf{x}, \mathbf{d}) = \arg \max_{d_{\Delta_t}, \mathbf{d}_{\text{turb}}} \mathbf{q}^T (\mathbf{f}_{C1}(\mathbf{x}, \mathbf{d}_{\text{turb}}) + \mathbf{f}_{C2}(\mathbf{x}, d_{\Delta_t}))$. Due to this separable structure, we begin by computing the worst-case disturbance, d_{Δ_t} , which captures adverser winch control

$$d_{\Delta_t}^* \in \arg \max_{d_{\Delta_t} \in [-d_{\Delta_t, \max}, d_{\Delta_t, \max}]} \mathbf{q}^T \mathbf{f}_{C2}(\mathbf{x}, d_{\Delta_t}), \quad (4.23)$$

where $d_{\Delta_t, \max} = 0.005$ is considered the bound of $\dot{\Delta}_t$ as derived in Section 4.3.1. Solving (4.23) yields $d_{\Delta_t}^* = -\text{sign}(q_7)d_{\Delta_t, \max}$.

For the computation of the disturbance vector $\mathbf{d}_{\text{turb}}$, the worst-case wind turbulence, we need to consider the effects the wind turbulence has on the position and heading of the aircraft. The wind turbulence naturally influences the wind velocity and subsequently the kinematic velocity of the aircraft. We thus begin by rewriting the kinematic velocity in the τ frame, whereby we replace the wind turbulence with the disturbance vector $\mathbf{d}_{\text{turb}}$, i.e.,

$$(\mathbf{v}_k)_\tau = \underbrace{\mathbf{M}_{\tau W} \mathbf{M}_{W O} \left(\mathbf{M}_{O \bar{A}} \begin{bmatrix} v_a \\ 0 \\ 0 \end{bmatrix}_{\bar{A}} + (\mathbf{v}_{\text{shear}})_O \right)}_{(\mathbf{v}_{k,0})_\tau} + \underbrace{\mathbf{M}_{\tau W} (\mathbf{d}_{\text{turb}})_W}_{(\mathbf{d}_{\text{turb}})_\tau}. \quad (4.24)$$

This allows us to rewrite the position propagation in the NED frame as

$$\dot{\lambda} = \underbrace{\frac{(v_{k,0})_{\tau,y}}{\cos(\phi)h_\tau}}_{\dot{\lambda}_0} + \frac{(d_{\text{turb}})_{\tau,y}}{\cos(\phi)h_\tau}, \quad \dot{\phi} = \underbrace{\frac{(v_{k,0})_{\tau,x}}{h_\tau}}_{\dot{\phi}_0} + \frac{(d_{\text{turb}})_{\tau,x}}{h_\tau}, \quad \dot{h}_\tau = \underbrace{-(v_{k,0})_{\tau,z}}_{\dot{h}_{\tau,0}} - (d_{\text{turb}})_{\tau,z}, \quad (4.25)$$

where $(v_{k,0})_{\tau,x}$, $(v_{k,0})_{\tau,y}$ and $(v_{k,0})_{\tau,z}$ denote the x, y, and z components of the base kinematic velocity in the τ frame and $(d_{\text{turb}})_{\tau,x}$, $(d_{\text{turb}})_{\tau,y}$ and $(d_{\text{turb}})_{\tau,z}$ denote the x, y, and z components of the turbulence disturbance vector in the τ frame. Recall that the tether force (4.6) is also affected by the kinematic velocity of the

aircraft through the difference between the velocity of the aircraft and the final tether segment, given by $\mathbf{s}_{v,n+1} = \mathbf{v}_{\text{aircraft}} - \mathbf{v}_n$. Thus the turbulence will only affect the tether force if the turbulence acting on $\mathbf{v}_{\text{aircraft}}$ is notably different from that acting on $\mathbf{v}_n$. Since we already incorporate uncertainty in the tether force by means of d_{Δ_t} which is chosen conservatively, we neglect the minimal impact of the wind turbulence on the tether force.

Thus the worst-case turbulence disturbance can be calculated by

$$\mathbf{d}_{\text{turb}}^* \in \arg \max_{\substack{d_{\text{turb},x} \in D_{\text{turb}}, \\ d_{\text{turb},y} \in D_{\text{turb}}, \\ d_{\text{turb},z} \in D_{\text{turb}}}} [q_1, q_2, q_3] [\dot{\lambda}, \dot{\phi}, \dot{h}_\tau]^T, \quad (4.26)$$

where $D_{\text{turb}} = [-d_{\text{turb,max}}, d_{\text{turb,max}}]$, with $d_{\text{turb,max}} = 4\frac{m}{s}$, the possible range of turbulence velocities. The results in (4.25) utilise the disturbance in the τ frame, while we require the final result to be in the wind frame. To this end, we transform $(\mathbf{d}_{\text{turb}})_\tau$ back into the wind frame and drop the additive term $(\mathbf{v}_{k,0})_\tau$, resulting in the final worst-case turbulence disturbances that maximise the Hamiltonian

$$\begin{aligned} d_{\text{turb},x}^* &= -d_{\text{turb,max}} \cdot \text{sign} \left[\sin(\phi) \cos(\lambda) \left(q_1 \frac{(t)_{\tau,x}}{\|\mathbf{t}\|_2 l_\Gamma} + q_2 \frac{(t_\perp)_{\tau,x}}{\|\mathbf{t}_\perp\|_2} \right) \right. \\ &\quad \left. + \sin(\lambda) \left(q_1 \frac{(t)_{\tau,y}}{\|\mathbf{t}\|_2 l_\Gamma} + q_2 \frac{(t_\perp)_{\tau,y}}{\|\mathbf{t}_\perp\|_2} \right) + \cos(\phi) \cos(\lambda) \left(q_1 \frac{(t)_{\tau,z}}{\|\mathbf{t}\|_2 l_\Gamma} + q_2 \frac{(t_\perp)_{\tau,z}}{\|\mathbf{t}_\perp\|_2} - q_3 \right) \right] \\ d_{\text{turb},y}^* &= d_{\text{turb,max}} \cdot \text{sign} \left[-\sin(\phi) \sin(\lambda) \left(q_1 \frac{(t)_{\tau,x}}{\|\mathbf{t}\|_2 l_\Gamma} + q_2 \frac{(t_\perp)_{\tau,x}}{\|\mathbf{t}_\perp\|_2} \right) \right. \\ &\quad \left. + \cos(\lambda) \left(q_1 \frac{(t)_{\tau,y}}{\|\mathbf{t}\|_2 l_\Gamma} + q_2 \frac{(t_\perp)_{\tau,y}}{\|\mathbf{t}_\perp\|_2} \right) - \cos(\phi) \sin(\lambda) \left(q_1 \frac{(t)_{\tau,z}}{\|\mathbf{t}\|_2 l_\Gamma} + q_2 \frac{(t_\perp)_{\tau,z}}{\|\mathbf{t}_\perp\|_2} - q_3 \right) \right] \\ d_{\text{turb},z}^* &= d_{\text{turb,max}} \cdot \text{sign} \left[\cos(\phi) \left(q_1 \frac{(t)_{\tau,x}}{\|\mathbf{t}\|_2 l_\Gamma} + q_2 \frac{(t_\perp)_{\tau,x}}{\|\mathbf{t}_\perp\|_2} \right) \right. \\ &\quad \left. - \sin(\phi) \left(q_1 \frac{(t)_{\tau,z}}{\|\mathbf{t}\|_2 l_\Gamma} + q_2 \frac{(t_\perp)_{\tau,z}}{\|\mathbf{t}_\perp\|_2} - q_3 \right) \right] \end{aligned}$$

With both the worst-case disturbance and optimal control inputs computed, the Hamiltonian can be calculated. To solve (4.18) we employ the Level Set Method toolbox of [103] and begin by initialising the value function at $t = 0$ with $\omega(\mathbf{x}, 0) = \max\{h(\mathbf{x}), l(\mathbf{x})\}$. As we solve the quasi-variational inequality (4.18), we compute the optimal control inputs that form our control policy $\mathbf{u}(\mathbf{x}, t)$. The evolution of the value function over the time period $[-0.1s, 0s]$ is shown in Figure 4.8. Any

state in the sub-zero level set belongs to the BRS, i.e., the set of points that can reach $\mathcal{R}$ while avoiding $\mathcal{A}$ within t units of time.

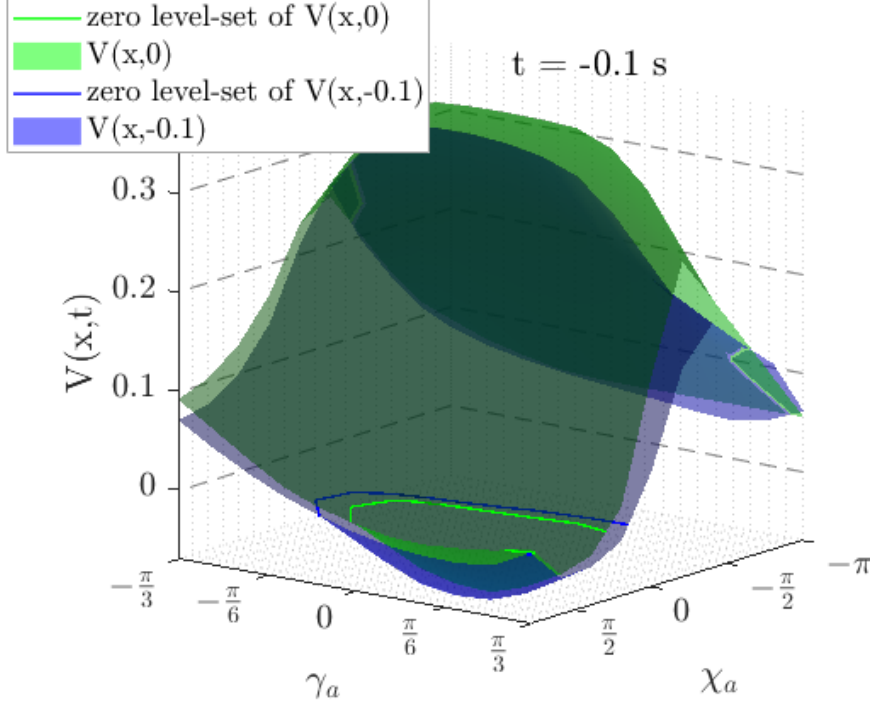


Figure 4.8: Visualization of the value function used for safety controller synthesis projected along $[s = 4\pi/3, \sigma = 20m, h_\tau = 250m, v_a = 31m/s, \Delta_t = 0.3mm]$.

4.4.3 Hybrid Control Setup

Let us refer to the safety and the NDI control laws as u_{safety} and u_{NDI} respectively. The aim is to combine these control laws in such a way as to mitigate the negative effects on power generation through conservative and sub-optimal manoeuvres while maintaining safety properties. In the context of flight manoeuvres, logic-based switching has become a common practice as it has many advantages over traditional adaptive control techniques, especially when considering systems with a high degree of model uncertainty [86, pp. 75, 130] [82, 135]. Utilising logic-based switching to incorporate HJ reachability into a least restrictive control law has been demonstrated in the past in [65, 69]. However, unlike previous implementations of the HJ reachability least restrictive control laws, incorporating switching into more complex systems, such as flight control, requires careful consideration of

the switching behaviour. Badly chosen switching rules can lead to chattering and destabilisation of the system. Thus, in summary, we derive a switching law based on the following criteria:

1. Allow for fast switching to the avoidance manoeuvre to facilitate a rapid response.
2. Prevent the system from being destabilised due to chattering behaviour.
3. Account for uncertainty in the tether force estimation.
4. Minimise negative effects on power generation.

A common method to prevent the system from being destabilised is to rely upon slow switching conditions, such as dwell time switching or hysteresis switching. As the avoidance manoeuvre generated by the safety controller will not have a clear minimum execution time, it becomes difficult to determine the appropriate average dwell time and potential chatter-bound [76]. To this end, the switching law derived in the sequel is designed based on hysteresis switching and sliding modes [86, pp. 12–14]. In the context of systems with large uncertainty, the switching is conventionally performed using supervisory control, whereby an estimator is used in combination with a monitoring signal generator, and finally, a switching logic is used to determine an appropriate controller [86, p. 133]. The integration of state estimation into the control process is outside the scope of this work, thus we restrict ourselves to deriving an appropriate switching logic.

To create the necessary switching surfaces, we will consider the current tether force as well as a first-order approximation of the predicted tether force. Since the actuation delays are not accounted for in the safety control synthesis, belated switching to the safety controller will potentially not give the safety controller ample time to take the necessary action to avoid a critical system failure. To this end, we choose the switching surface S_1 to lie slightly below the maximum tether rupture force.

$$S_1(\mathbf{F}_t) = (\|\mathbf{F}_t\|_2 \geq F_{\text{rupture}} - 30N) \vee (\|\mathbf{F}_t\|_2 + \frac{\partial \|\mathbf{F}_t\|_2}{\partial t} t_f \geq F_{\text{rupture}} - 50N), \quad (4.27)$$

where $t_f = 0.1s$ is the time horizon used for the BRS computation and $\overline{\|\mathbf{F}_t\|_2}$ is the moving average of the tether force acting on the aircraft. The numerical values in (4.27) are tuned experimentally, while due to uncertainty in the linear extrapolation of the tether force, the threshold for the predicted force is chosen lower. As a switch from u_{NDI} to u_{safety} is considered time-sensitive, we utilise sliding modes, to prevent abrupt manoeuvres while maintaining speed.

Using solely sliding modes, however, may lead to chattering. To this end, we draw on the aforementioned hysteresis switching logic and define a second switching surface, S_2 , that is offset from the first and utilised to switch from u_{safety} to u_{NDI} .

$$S_2(\mathbf{F}_t) = \neg S_1(\|\mathbf{F}_t\|_2) \wedge \left[(\|\mathbf{F}_t\|_2 \leq F_{\text{rupture}} - 40N) \vee \left(\frac{\partial \overline{\|\mathbf{F}_t\|_2}}{\partial t} \leq 0 \right) \right]. \quad (4.28)$$

A summary of the switching logic is illustrated in Figure 4.9.

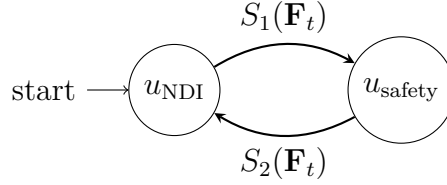


Figure 4.9: State automaton of the hybrid control setup.

4.5 Simulation

4.5.1 Simulation Setup

To verify the effectiveness of the derived avoidance strategy, we consider the simplified dynamics derived in Section 4.3 simulated using the worst-case disturbances. As can be seen in Figure 4.10, when the avoidance strategy is engaged, the controller derived by means of HJ reachability is successfully able to decrease the tether force. To verify the feasibility of the chosen strategy, however, it is necessary to demonstrate its effectiveness in conjunction with the guidance and regular path-following controllers. Furthermore, integrating the chosen avoidance manoeuvre into a larger simulation framework serves as validation that the chosen robustness margins are adequate at handling unmodelled adversarial effects, such as actuator

delays. To this end, we integrate the avoidance strategy into an extension of the MATLAB Simulink framework presented in [115] and [49].

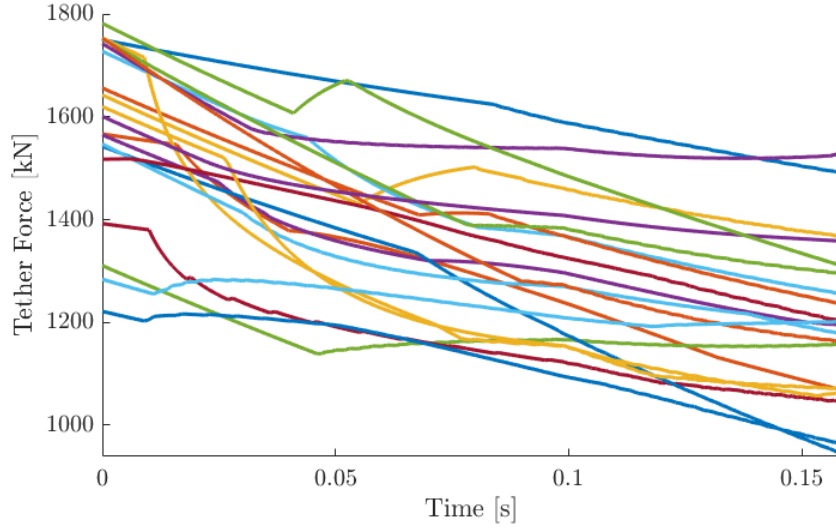


Figure 4.10: The tether force acting on the aircraft during the avoidance manoeuvre under worst-case disturbances. Each colour represents a random initial position, velocity, and tether force and the evolution of the tether force as the avoidance manoeuvre is executed.

Together with the subsequent discussion, the validation of the presented hybrid control setup constitutes our fourth key contribution of this chapter. Some of the relevant simulation parameters are presented in Table 4.1 and we have made the simulation code available ¹.

Parameter	Value	Description
F_{ref}	1600 N	Force tracked by the winch controller
F_{rupture}	1870 N	Force at which a tether rupture occurs
$ W_{20} $	$9 \frac{m}{s}$	Measured wind speed at 20 feet
ϖ	π	Wind direction
n_{tether}	5	Tether segments used during simulation

Table 4.1: Simulation Parameters

To acquire the safety control inputs, u_{safety} , we solve (4.22) using the numerically computed $\frac{\partial \omega(\mathbf{x}, t_f)}{\partial \mathbf{x}}$ as our costate vector. Since calculating the derivatives of the value function is computationally intensive, it is impractical to compute u_{safety}

¹www.vertovec.info/code/AWE-simulation

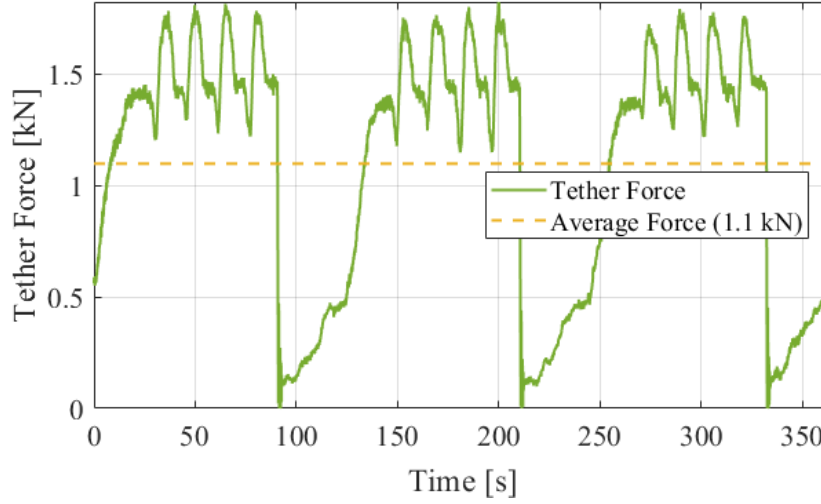


Figure 4.11: The tensile force of the tether acting on the aircraft during flight.

during the simulation. To this end, we calculate the optimal safety control inputs offline using a high-fidelity grid and save them on each grid point, using the saved values as a lookup table to be used during simulation.

Both control inputs, u_{safety} and u_{NDI} are passed to the control switch, which determines which control input to use based on the switching law derived in Section 4.4.3. The final control u_{cmd} is passed to the actuator, which simulates the actuator delays. The bank angle μ_a and angle of attack α_a are then passed to the aircraft model. To complete the pumping cycle and execute the retraction phase, we utilise the framework derived in [115].

4.5.2 Simulation Results

To evaluate the performance of the safety controller, we run the simulation for multiple pumping cycles to determine if, at any point, the maximum allowed tether force is exceeded. The tether force of three pumping cycles, as well as the switching behaviour of the safety controller, is shown in Figure 4.11, 4.12 and 4.13. Notice how the tether force is able to remain high throughout the pumping cycle and only minimal interventions by the safety controller (blue segments in Figure 4.13) are necessary to prevent a tether rupture.

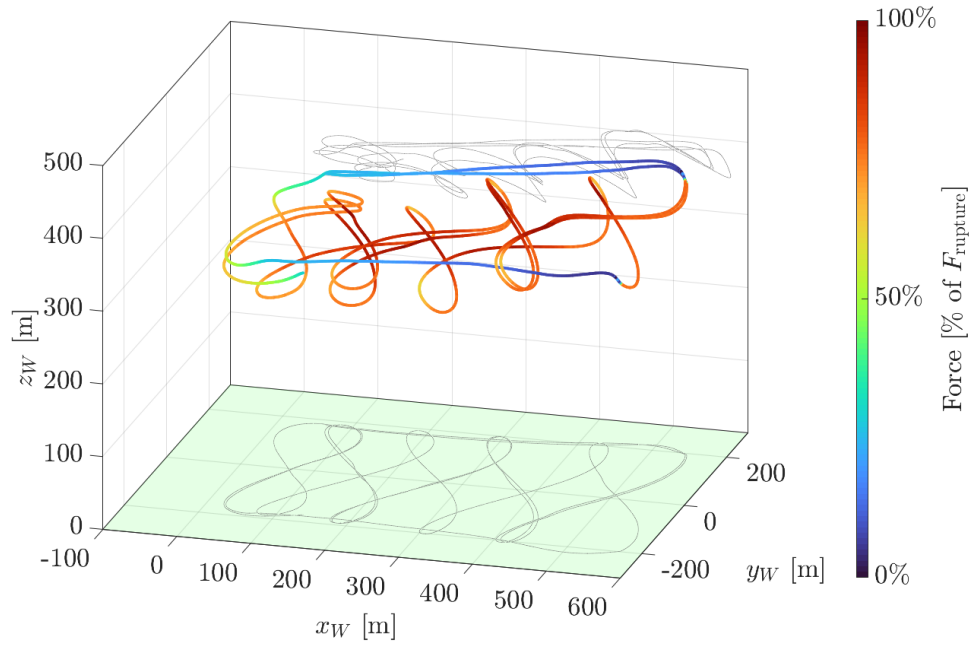


Figure 4.12: The tether force in % of the maximum allowed tether force before a rupture occurs.

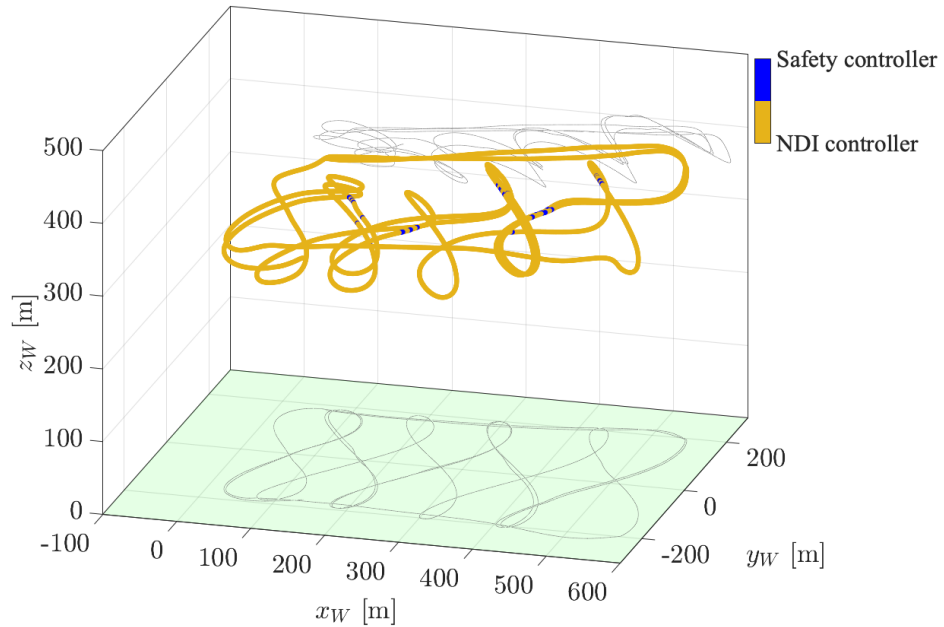


Figure 4.13: The safety controller only needs to be activated at particular times. For the remainder of the flight, a more power-optimal controller, such as the NDI controller can be employed.

The mechanical power that is able to be harnessed by the ground station can be calculated by multiplying the tether force at the ground station by the reel-out speed, i.e., $P_{\text{mech}} = \dot{l}_s \|\mathbf{F}_w\|_2$. In Figure 4.14 and 4.15, the power generation capabilities of the AWE system are shown. The average power generated over 3 cycles is 3.346 kW with a peak power output of 19.927 kW. Without the safety controller and neglecting the tether's rupture the average generated power over 3 cycles was 3.354 kW with a peak power output of 21.024 kW. Thus the safety controller is able to enforce safety with only minimal effects on power generation.

To further put the power generation into context, we derive a theoretical upper limit of the power that could be generated based on the work of [44]. We begin by calculating the maximum power harvesting factor, ζ , based on the drag and lift coefficients presented in [118]. The power harvesting factor is a common metric used by both AWE applications as well as conventional wind turbines (not to be confused with the power coefficient commonly used for wind turbines)

$$\zeta = \frac{4}{27} \frac{C_L^3}{(C_D + C_{D\text{tether}})^2}. \quad (4.29)$$

The power harvesting factor is unique for a given AWE setup and is used in [47] and [89] to calculate the theoretical limit P_{max} . However, in order to calculate a tighter, more realistic upper bounds of the power that a given controller could achieve, we calculate an efficiency factor e that can be multiplied by the theoretical upper bound of the power generation, P_{max} . This theoretical upper bound is given by Theorem 1 in [44], $\tilde{P} = P_{\text{max}} \underbrace{\cos^3(\gamma_0)}_e$, where γ_0 is the optimal angle between the aerodynamic force and the wind.

Based on the observation that the average aerodynamic force needs to balance the gravitational force acting on the aircraft as well as the force exerted on the aircraft by the tether (assuming that on average the system is not accelerating), we can state the following equality based on a 2-dimensional simplification

$$\|\mathbf{F}_a\|_2 \begin{bmatrix} \cos \gamma_0 \\ \sin \gamma_0 \end{bmatrix} - \|\mathbf{F}_t\|_2 \begin{bmatrix} \cos \psi \\ \sin \psi \end{bmatrix} + \begin{bmatrix} \|\mathbf{F}_{\text{drag}}\|_2 \\ \|\mathbf{F}_g\|_2 \end{bmatrix} = 0, \quad (4.30)$$

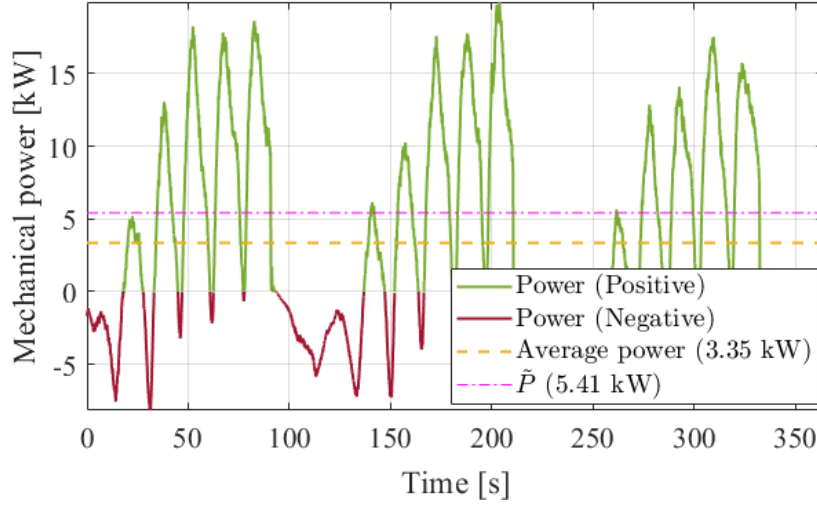


Figure 4.14: The power P_{mech} that is generated at the ground station during flight.

where ψ is the angle between the tether and the ground and $\|\mathbf{F}_{\text{drag}}\|_2$ is the negated x component of the aerodynamic force, $(\mathbf{F}_a)_A$. On average, ψ is equal to the rotation of the tracking curve, i.e., $\psi = \psi_0$. As in [44], we multiply both sides by $[\sin \psi, -\cos \psi]^T$, which cancels out the effect of the tether, allowing us to solve for γ_0 , which in turn leads to the theoretical efficiency factor

$$e = \cos^3 \left(\psi_0 + \sin^{-1} \left(\frac{\|\mathbf{F}_{\text{drag}}\|_2}{\|\mathbf{F}_a\|_2} \sin \psi_0 + \frac{\|\mathbf{F}_g\|_2}{\|\mathbf{F}_a\|_2} \cos \psi_0 \right) \right). \quad (4.31)$$

Finally, we calculate the theoretical maximum power that the hybrid control setup could obtain as $\tilde{P} = e A_{\text{eff}} \zeta \frac{1}{2} \rho \|\mathbf{v}_w\|_2^3$. When evaluated and averaged, this equates to a theoretical average power of 5.41 kW. Thus the hybrid control setup comes close to the upper bound that could be achieved.

4.6 Conclusion

We introduce a novel avoidance manoeuvre for the path-following controller of an AWE system through the introduction of a low-dimensional simplification of the system dynamics and application of HJ reachability analysis. Utilising the presented switching law, the avoidance manoeuvre can be used in conjunction with an arbitrary primary flight controller for safe and power-optimal flight. We demonstrate the efficacy of the hybrid control setup through a full-scale simulation with a wind

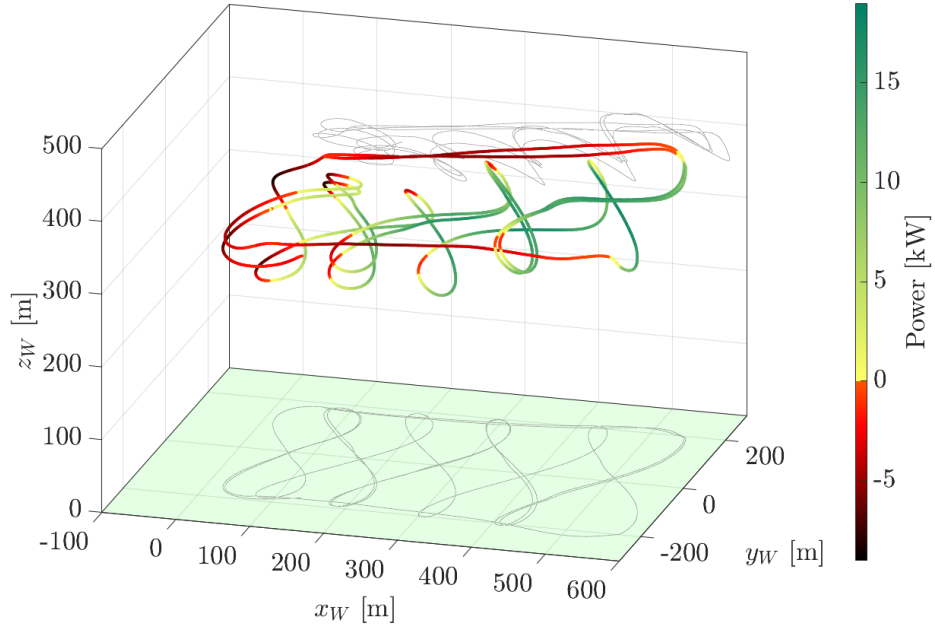


Figure 4.15: Generated power in relation to the aircraft's flight path. Negative (resp. positive) power indicates a reeling in (resp. out) by the winch thus power is exerted (resp. harvested).

turbulence model, actuator delays, and nonlinear tether dynamics, validating that the chosen simplifications and worst-case considerations in the controller synthesis are sufficient for ensuring safe flight and avoiding a critical tether rupture. In the subsequent chapter, we will depart from HJ reachability analysis to address the second distinct notion of safety considered in this thesis.

4.7 Appendix

Coordinate transformation matrices

For brevity, we only list the transformation matrices in one direction, as the inverse transformation is obtained by simply transposing the appropriate matrix since all

transformation matrices are orthogonal [106, p. 87].

$$\begin{aligned}
\mathbf{M}_{AB} &= \begin{bmatrix} \cos \alpha_a \cos \beta_a & \sin \beta_a & \sin \alpha_a \cos \beta_a \\ -\cos \alpha_a \sin \beta_a & \cos \beta_a & -\sin \alpha_a \sin \beta_a \\ -\sin \alpha_a & 0 & \cos \alpha_a \end{bmatrix}, & \mathbf{M}_{\bar{A}A} &= \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \mu_a & -\sin \mu_a \\ 0 & \sin \mu_a & \cos \mu_a \end{bmatrix}, \\
\mathbf{M}_{\tau W} &= \begin{bmatrix} -\sin \phi \cos \lambda & -\sin \phi \sin \lambda & \cos \phi \\ -\sin \lambda & \cos \lambda & 0 \\ -\cos \phi \cos \lambda & -\cos \phi \sin \lambda & -\sin \phi \end{bmatrix}, & \mathbf{M}_{OW} &= \begin{bmatrix} \cos \varpi & \sin \varpi & 0 \\ \sin \varpi & -\cos \varpi & 0 \\ 0 & 0 & -1 \end{bmatrix}, \\
\mathbf{M}_{\bar{A}O} &= \begin{bmatrix} \cos \chi_a \cos \gamma_a & \sin \chi_a \cos \gamma_a & -\sin \gamma_a \\ -\sin \chi_a & \cos \chi_a & 0 \\ \cos \chi_a \sin \gamma_a & \sin \chi_a \sin \gamma_a & \cos \gamma_a \end{bmatrix}, & \mathbf{M}_{WP} &= \begin{bmatrix} \cos(\psi_0) & 0 & -\sin(\psi_0) \\ 0 & 1 & 0 \\ \sin(\psi_0) & 0 & \cos(\psi_0) \end{bmatrix}.
\end{aligned}$$

5

State Aggregation for Distributed Value Iteration in Dynamic Programming

5.1 Introduction

Value iteration is a well-established method for solving dynamic programming problems, yet it exhibits scalability issues for applications with a large state-space. State aggregation can be used to reduce the number of states that need to be considered and has a long history in scientific computing with applications ranging from the improvement of Galerkin methods [37], to solving large-scale optimisation models [120], and dynamic programming [15]. In [138] it is shown how the aggregation approach can be used in conjunction with value iteration, however, for problems with large decision spaces and cost and transition probabilities that potentially evolve over time, conventional aggregation will still be hampered by scalability issues. In light of this, we propose a multi-agent value iteration algorithm that utilises aggregation to minimise communication requirements between agents and allows for solving dynamic programming problems where each agent has only partial knowledge of the state transition costs and probabilities.

Value iteration has been expanded to a multi-agent framework for the class of problems that contain a joint decision space [20], yet in this approach no restrictions are imposed as far as the agents' knowledge of the underlying transition probabilities

and costs are concerned. Such restrictions have been considered for small-scale problems as in [112], however, require a centralised critic for estimating the expected team benefit in a non-cooperative setting.

The proposed framework differs from many multi-agent reinforcement learning approaches that utilise sharing a weighted average of agents' local estimated value. Rather than considering a set of global states shared by all the agents in conjunction with a local reward observed by individual agents as in [48, 71], we consider a Markov Decision Process (MDP) partitioned among agents with the objective of determining in a distributed manner the global value function without disclosing information on the transition probabilities and costs among agents. This allows us to address the second distinct notion of safety considered in this thesis: limiting the sharing of private information.

This chapter performs the following main contributions: (i) We propose a distributed value iteration algorithm for multi-agent dynamic programming, preventing transition probabilities and state costs from constituting global information among agents; (ii) We provide a rigorous mathematical analysis on the convergence and optimality properties of the proposed algorithm, merging tools from multi-agent consensus and dynamic programming principles; and, (iii) We demonstrate our scheme on a non-trivial traffic routing problem.

The rest of the chapter is organised as follows. Section 5.2 provides some mathematical preliminaries and introduces the distributed multi-agent value iteration algorithm. Section 5.3 provides the main statements and associated mathematical analysis supporting the proposed algorithm. In Section 5.4 we introduce a traffic routing problem and show how our proposed approach is able to solve the problem in a distributed manner. Finally, Section 5.5 provides concluding remarks and directions for future work.

5.2 Problem Setup

5.2.1 Multi-agent Markov Decision Processes

We first introduce a standard Markov Decision Process (MDP) that will be expanded to a multi-agent setting in view of the proposed distributed algorithm presented in the sequel. We consider a finite set of n states denoted by $\mathcal{X}$ and let $g(i, u, j)$ be the cost-to-go from state $i \in \mathcal{X}$ to state $j \in \mathcal{X}$, using the input $u \in \mathcal{U}(i)$, where $\mathcal{U}(i)$ is a finite set of actions/decisions available at state i . Furthermore, let $p(i, u, j)$ be the transition probability to transition from state i to state j using the input u , with $\sum_{j=1}^n p(i, u, j) = 1$. The objective is to solve a global discounted infinite horizon dynamic programming problem. The associated Bellman equation is given by $J^*(i) = \min_{u \in \mathcal{U}(i)} \sum_{j=1}^n p(i, u, j)[g(i, u, j) + \alpha J^*(j)]$, where $\alpha \in [0, 1)$ is a discount factor, and J^* denotes the optimal value function that satisfies the Bellman identity. Solving the Bellman equation using conventional value iteration or policy iteration requires knowledge of all transition probabilities and associated costs.

We consider the setting where a set of q agents collaborate to solve the aforementioned infinite horizon dynamic programming problem in a distributed manner, with only partial knowledge of the state transition probabilities and costs while minimising communication between agents. As such, we partition the state-space $\mathcal{X}$ into q subsets, $I_\ell \subset \mathcal{X}$, such that for all $m, \ell \in \{1, \dots, q\}$ with $m \neq \ell$, $I_\ell \cap I_m = \emptyset$. Each agent knows only the state transition probabilities and cost for transitions originating within its state subset I_ℓ , i.e., for each agent ℓ , $g_\ell : I_\ell \times \mathcal{U} \times \mathcal{X} \rightarrow \mathbb{R}$, $p_\ell : I_\ell \times \mathcal{U} \times \mathcal{X} \rightarrow [0, 1]$. Using conventional value iteration would require each agent to share its knowledge of the transition probabilities and costs, resulting in significant communication overhead. In the subsequent section, we introduce an alternative method relying on state aggregation that allows solving the discounted infinite horizon dynamic programming problem under consideration in a distributed manner where some tentative information is exchanged only with neighbouring agents.

5.2.2 Distributed Value Iteration

We start by aggregating the value function (expected optimal cost-to-go) to construct for each agent $\ell = 1, \dots, q$, the aggregate value, defined as $r_{\ell,\ell} = \sum_{i \in I_\ell} d_{\ell i} J(i)$, where J constitutes an approximation of the optimal value function of the Bellman equation, and $d_{\ell i}$ is the so-called disaggregation probability (encoding the contribution of each agent's value function to the aggregate one), satisfying for all $\ell = 1, \dots, q$, $\sum_{i \in I_\ell} d_{\ell i} = 1$ and for all $i \notin I_\ell$, $d_{\ell i} = 0$. The aggregate values for each agent, $r_{\ell,\ell}$, can be combined into a common vector denoted by r_ℓ ; tentative values for this vector will be communicated across all agents. Thus we define $r_\ell = [r_{\ell,1}, \dots, r_{\ell,\ell}, \dots, r_{\ell,q}]^T$.

In the sequel, we will define an iterative scheme, according to which each agent $\ell = 1, \dots, q$ will update their estimate for the vector r_ℓ ; we denote this at iteration k , by r_ℓ^k , where its m -th element is indicated by $r_{\ell,m}^k$, $m = 1, \dots, q$. Each value $r_{\ell,m}^k$ for $m \neq \ell$ corresponds to agent ℓ 's last received value of the aggregated term $r_{m,m}$, while $r_{\ell,\ell}^k$ will denote agent ℓ 's own aggregate value at iteration k . The vector r_ℓ can be initialised to some tentative value and will be updated as agent ℓ receives updated values from its neighbours.

Next, we introduce the aggregation probability satisfying for all $\ell = 1, \dots, q$,

$$\phi_{j\ell} = \begin{cases} 1, & \text{if } j \in I_\ell, \\ 0, & \text{otherwise.} \end{cases}$$

Such a formulation of the aggregation probability is known as hard aggregation in the literature [21, p. 311].

To solve the discounted infinite horizon dynamic programming problem, we propose a distributed algorithm for which the pseudocode is given in Algorithm 1 and 2. The proposed algorithm involves an agent-to-agent communication protocol. At each algorithm iteration $k \geq 0$ we consider the directed communication graph $(V, E(k))$, where the node set $V = \{1, \dots, q\}$ includes the agents and the set $E(k)$, the directed edges (m, l) , indicating that at iteration k agent ℓ can receive information from agent m . Let $E_B(k) = \bigcup_{i=k}^{(k+B)} E(i)$ for some integer B . We make the fairly standard assumption in the existing literature on the communication structure between agents [48, 107]:

Assumption 5.1. *[Connectivity and Communication] There exists a positive integer B such that $(V, E_B(k))$ is fully connected for each iteration k .*

Assumption 5.1 implies that for any agent pair (ℓ, m) there is a direct link at least once every B iterations. This prevents agents from having to share information with all other agents at all iterations as well as for the presence of a central authority.

Algorithm 1 Distributed value iteration

```

1: Initialisation
2: Set  $r_\ell^0 \leftarrow [0, \dots, 0]$  for all  $\ell = 1, \dots, q$ 
3: Set  $r_{\ell,\ell}^{\text{prev}} \leftarrow 0$  for all  $\ell = 1, \dots, q$ 
4: Set  $V_\ell^0(i) \leftarrow 0$  for all  $i \in I_\ell$  and  $\ell = 1, \dots, q$ 
5:  $k = 0$ 
6: repeat
7:   for each Agent  $\ell \in 1, \dots, q$  do
8:      $V_\ell^{k+1}, r_{\ell,\ell}^{k+1} \leftarrow \text{Agent-Update}(V_\ell^k, r_\ell^k)$ 
9:      $N_\ell(k) = \{m \in \{1, \dots, q\} : (\ell, m) \in E(k)\}$ 
10:    if  $\|r_{\ell,\ell}^{k+1} - r_{\ell,\ell}^{\text{prev}}\| > C_{\text{threshold}}$ ,
11:      or if  $(\ell, m) \notin \bigcup_{i=k-B+1}^k E(i)$  then
12:        Send  $r_{\ell,\ell}^{k+1}$  to all agents  $m \in N_\ell(k)$ 
13:      end if
14:      if receiving information from an agent  $m$  then
15:         $r_{\ell,m}^{k+1} \leftarrow r_{m,m}^{k+1}$ 
16:      else
17:         $r_{\ell,m}^{k+1} \leftarrow r_{\ell,m}^k$ 
18:      end if
19:    end for
20: until  $\|r_\ell^{k+1} - r_\ell^k\| \leq \text{tolerance}$  for all  $\ell = 1, \dots, q$ 

```

Algorithm 2 Agent-Update

```

1: function AGENT-UPDATE( $V_\ell^k, r_\ell^k$ )
2:   for  $i \in I_\ell$  do
3:      $V_\ell^{k+1}(i) \leftarrow \min_{u \in U(i)} \sum_{j=1}^n p_\ell(i, u, j) [g_\ell(i, u, j)$ 
4:        $+ \alpha(\phi_{j\ell} V_\ell^k(j) + \sum_{\substack{m=1 \\ m \neq \ell}}^q \phi_{jm} r_{\ell,m}^k)]$ 
5:      $V_\ell^k(i) \leftarrow V_\ell^{k+1}(i)$ 
6:   end for
7:    $r_{\ell,\ell}^{k+1} \leftarrow \sum_{j \in I_\ell} d_{\ell j} V_\ell^{k+1}(j)$ 
8:   return  $V_\ell^{k+1}, r_{\ell,\ell}^{k+1}$ 
9: end function

```

Initially in Algorithm 1, each agent ℓ , $\ell = 1, \dots, q$, starts with some tentative value of the aggregate vector r_ℓ^0 , and local cost function V_ℓ^0 . The initialisation

choice is arbitrary. Utilising the aggregate vector r_ℓ^k at iteration $k \geq 0$, each agent ℓ locally and in parallel executes Algorithm 2 so as to construct the updated local value function $V_\ell^{k+1} : I_\ell \rightarrow \mathbb{R}$ which will, in turn, yield an updated aggregated cost $r_{\ell,\ell}^{k+1} \in \mathbb{R}$ (Algorithm 1, Steps 8).

To minimise communication overhead, an updated aggregated cost $r_{\ell,\ell}^{k+1}$ is only shared with other agents in the set $N_\ell(k)$ (the neighbours of ℓ at iteration k) when the aggregated cost has changed significantly, i.e., $\|r_{\ell,\ell}^{k+1} - r_{\ell,\ell}^{\text{prev}}\| > C_{\text{threshold}}$, where $r_{\ell,\ell}^{\text{prev}}$ is the aggregated cost previously broadcast to other agents and $C_{\text{threshold}}$ is a communication threshold (Algorithm 1, Steps 10 - 13), or when agent ℓ and a neighbouring agent m have not communicated in the last B iterations (see Step 11; this ensures that there exists a bounded intercommunication time as required by Assumption 5.1). For further discussion on the communication threshold to limit the transmission of insignificant data, we refer to [68].

Until convergence is reached, each agent will repeat the execution of Algorithm 2 and the subsequent communication step. Note that at iteration k , the vector r_ℓ^k may contain aggregate values from other agents received at prior time steps (Algorithm 1, Step 17). It will be shown in the proof of Theorem 5.1 that convergence will nevertheless be reached.

We now turn to the update of V_ℓ^k and $r_{\ell,\ell}^k$, performed when calling Algorithm 2. Each agent uses the local value function, V_ℓ^k , to represent the cost-to-go function at each state $i \in I_\ell$, and the aggregated cost, r_ℓ^k , to approximate the cost for states $j \in \mathcal{X} \setminus I_\ell$. The local value function is iteratively updated for each state $i \in I_\ell$ (Algorithm 2, Steps 2 - 6). This update follows from standard value iteration; choosing the control action at state i that minimises the expected cost-to-go (Algorithm 2, Step 4) and using a Gauss-Seidel update on the local value function to improve convergence and minimise memory usage (Algorithm 2, Step 5). We perform an update only to the local value function since we restrict ourselves to states for which sufficient knowledge of the transition probabilities and costs is available. After the local value function has been updated, the updated local aggregated cost, $r_{\ell,\ell}^{k+1}$, is computed (Algorithm 2, Step 7).

Notice that Algorithms 1 and 2 prevent disclosing the transition probabilities to all agents; in contrast, each agent ℓ , has access only to the probabilities and costs associated with transitions originating from their partition I_ℓ , $\ell = 1, \dots, q$.

5.3 Algorithm analysis

The following theorem is the main result of this section.

Theorem 5.1. *Consider Assumption 5.1. Algorithm 1 converges asymptotically to a common value for r among agents, i.e., there exists $\bar{r}$ such that for all $\ell = 1, \dots, q$, $\lim_{k \rightarrow \infty} \|r_\ell^k - \bar{r}\| = 0$. Moreover, for all $\ell = 1, \dots, q$, for all $i \in \mathcal{X}$, $V_\ell^k(i)$ converges to some $\bar{V}_\ell^*(i)$.*

Proof. We show convergence when agents transmit $r_{\ell,\ell}^k$, $\ell = 1, \dots, q$, irrespective of whether this deviates more than $C_{\text{threshold}}$ from their previously transmitted cost. This is without loss of generality as in this case, the resulting sequence of transmitted costs would form a subsequence of $\{r_{\ell,\ell}^k\}_{k=0}^\infty$, hence it will be convergent as we will show that $\{r_{\ell,\ell}^k\}_{k=0}^\infty$ converges.

Fix any $k \geq B$, and any $\ell \in \{1, \dots, q\}$. Let

$$\begin{aligned}\epsilon_{\ell,m}^k &= r_{\ell,m}^{k+1} - r_{\ell,m}^k, \\ \epsilon_{\ell,\infty}^k &= \max_{m=1,\dots,q} |r_{\ell,m}^{k+1} - r_{\ell,m}^k|, \\ \epsilon_\infty^k &= \max_{\ell=1,\dots,q} \sum_{n=k-B}^k \epsilon_{\ell,\infty}^n,\end{aligned}$$

i.e., ϵ_∞^k is the maximum among agents of the cumulative incremental update over the most recent $B + 1$ iterations; recall that due to Assumption 5.1 this is the window within which agent ℓ communicates with other agents at least once. Consider the update to the value $V_\ell^k(i)$, for an arbitrary iteration k and agent $\ell = 1, \dots, q$, which

we define as

$$\begin{aligned}
\delta_\ell^k(i) &= V_\ell^{k+1}(i) - V_\ell^k(i) \\
&= \min_{u \in U(i)} \sum_{j=1}^n p_\ell(i, u, j) \left(g_\ell(i, u, j) + \alpha [\phi_{j\ell} V_\ell^k(j) + \sum_{\substack{m=1 \\ m \neq \ell}}^q \phi_{jm} r_{\ell,m}^k] \right) - V_\ell^k(i) \\
&= \min_{u \in U(i)} \left(\sum_{j=1}^n p_\ell(i, u, j) \left(g_\ell(i, u, j) + \alpha [\phi_{j\ell} V_\ell^{k-1}(j) + \sum_{\substack{m=1 \\ m \neq \ell}}^q \phi_{jm} r_{\ell,m}^{k-1}] \right) \right. \\
&\quad \left. + \alpha \sum_{j=1}^n p_\ell(i, u, j) \left(\phi_{j\ell} \delta_\ell^{k-1}(j) + \sum_{\substack{m=1 \\ m \neq \ell}}^q \phi_{jm} \epsilon_{\ell,m}^{k-1} \right) \right) - V_\ell^k(i) \\
&\leq V_\ell^k(i) + \max_{u \in U(i)} \alpha \sum_{j=1}^n p_\ell(i, u, j) \left(\phi_{j\ell} \delta_\ell^{k-1}(j) + \sum_{\substack{m=1 \\ m \neq \ell}}^q \phi_{jm} \epsilon_{\ell,m}^{k-1} \right) - V_\ell^k(i) \\
&= \max_{u \in U(i)} \alpha \sum_{j=1}^n p_\ell(i, u, j) \left(\phi_{j\ell} \delta_\ell^{k-1}(j) + \sum_{\substack{m=1 \\ m \neq \ell}}^q \phi_{jm} \epsilon_{\ell,m}^{k-1} \right),
\end{aligned}$$

where the first equality follows from the definition of V_ℓ^k , and the second one from a rearrangement after substituting $V_\ell^k(j) = V_\ell^{k-1}(j) + \delta_\ell^{k-1}(j)$ and $r_{\ell,m}^k = r_{\ell,m}^{k-1} + \epsilon_{\ell,m}^{k-1}$. The inequality follows from the definition of V_ℓ^k and by considering the maximum over $u \in U(i)$.

For each $\ell = 1, \dots, q$, we now define the maximum update of $|\delta_\ell^k|$, over all states $i \in I_\ell$, as $\bar{\delta}_\ell^k = \max_{i \in I_\ell} |\delta_\ell^k(i)|$. It then follows that for all $i \in I_\ell$,

$$\begin{aligned}
\delta_\ell^k(i) &\leq \max_{u \in U(i)} \alpha \sum_{j=1}^n p_\ell(i, u, j) \left(\phi_{j\ell} \bar{\delta}_\ell^{k-1} + \sum_{\substack{m=1 \\ m \neq \ell}}^q \phi_{jm} \epsilon_{\ell,m}^{k-1} \right) \\
&\leq \max_{u \in U(i)} \alpha \sum_{j=1}^n p_\ell(i, u, j) \left(\sum_{m=1}^q \phi_{jm} \max\{\bar{\delta}_\ell^{k-1}, \epsilon_{\ell,m}^{k-1}\} \right) \\
&= \alpha \max\{\bar{\delta}_\ell^{k-1}, \epsilon_{\ell,\infty}^{k-1}\}, \tag{5.1}
\end{aligned}$$

where the second inequality follows from the definition of $\bar{\delta}_\ell^k$, and the equality follows from the fact that $\sum_{j=1}^n p(i, u, j) = 1$, and $\sum_{m=1}^q \phi_{jm} = 1$. Define $\bar{\delta}_\infty^k := \max_{\ell=1, \dots, q} \sum_{n=k-B}^k \bar{\delta}_\ell^n$, resulting in

$$\bar{\delta}_\infty^k \leq \alpha \max_{\ell=1, \dots, q} \sum_{n=k-B}^k \max\{\bar{\delta}_\ell^{n-1}, \epsilon_{\ell,\infty}^{n-1}\} = \alpha \max\{\bar{\delta}_\infty^{k-1}, \epsilon_\infty^{k-1}\}, \tag{5.2}$$

where the inequality is since (5.1) holds for all $i \in I_\ell$, and the exchange of the summation and the maximisation operator is since all quantities are non-negative.

At the same time,

$$\begin{aligned}
\sum_{n=k-B}^k \max_{m=1,\dots,q} |r_{\ell,m}^{n+1} - r_{\ell,m}^n| &\leq \sum_{n=k-B}^k \max_{m=1,\dots,q} |r_{m,m}^{n+1} - r_{m,m}^n| \\
&= \sum_{n=k-B}^k \max_{m=1,\dots,q} \left| \sum_{i \in I_m} d_{mi} (V_m^n(i) + \delta_m^n(i)) - r_{m,m}^n \right| \\
&= \sum_{n=k-B}^k \max_{m=1,\dots,q} \left| \sum_{i \in I_m} d_{mi} \delta_m^n(i) \right| \\
&= \sum_{n=k-B}^k \max_{m=1,\dots,q} \bar{\delta}_m^n = \max_{m=1,\dots,q} \sum_{n=k-B}^k \bar{\delta}_m^n = \bar{\delta}_\infty^k, \quad (5.3)
\end{aligned}$$

where the inequality is since $\sum_{n=k-B}^k |r_{\ell,m}^{n+1} - r_{\ell,m}^n|$ is bounded by the value this would become if agents ℓ and m communicated at every iteration $n = k - B + 1, \dots, k$ (see Steps 14-18, Algorithm 1). The first equality follows from the fact that $r_{m,m}^{n+1} = \sum_{i \in I_m} d_{mi} V_m^{n+1}(i)$ (see Step 7, Algorithm 2), and since $V_m^{n+1}(i) = V_m^n(i) + \delta_m^n(i)$, while the last one is due to the definition of $\bar{\delta}_\infty^k$.

By (5.3) we can upper-bound $\epsilon_\infty^k = \max_{\ell=1,\dots,q} \sum_{n=k-B}^k \max_{m=1,\dots,q} |r_{\ell,m}^{n+1} - r_{\ell,m}^n|$ as

$$\epsilon_\infty^k \leq \max_{\ell=1,\dots,q} \bar{\delta}_\infty^k \leq \alpha \max\{\bar{\delta}_\infty^{k-1}, \epsilon_\infty^{k-1}\}, \quad (5.4)$$

where the last inequality follows from (5.2), and the fact that the bound in (5.2) is independent of ℓ .

By (5.2) and (5.4) we then have that $\max\{\bar{\delta}_\infty^k, \epsilon_\infty^k\} \leq \alpha \max\{\bar{\delta}_\infty^{k-1}, \epsilon_\infty^{k-1}\}$, which implies that $\max\{\bar{\delta}_\infty^k, \epsilon_\infty^k\}$ is contractive. As a result, $\{\max\{\bar{\delta}_\infty^k, \epsilon_\infty^k\}\}_{k \geq 0}$, and hence also $\{\epsilon_\infty^k\}_{k \geq 0}$ and $\{\bar{\delta}_\infty^k\}_{k \geq 0}$ will be converging to zero. Since $\{\epsilon_\infty^k\}_{k \geq 0}$ converges, then each term in the summation in the definition of ϵ_∞^k will also converge, implying that $\{|r_{\ell,m}^{k+1} - r_{\ell,m}^k|\}_{k \geq 0}$ is convergent. This implies that for all $\ell = 1, \dots, q$ there exists $\bar{r}$ such that $\lim_{k \rightarrow \infty} \|r_\ell^k - \bar{r}\| = 0$, while convergence of $\{\bar{\delta}_\infty^k\}_{k \geq 0}$ (due to (5.2)), and the definition of $\delta_\ell^k(i)$ implies that for all $\ell = 1, \dots, q$, for all $i \in \mathcal{X}$, $\{V_\ell^k(i)\}_{k \geq 0}$ would be convergent, thus concluding the proof. $\square$

Theorem 5.1 implies consensus among agents to a common $\bar{r}$, and also establishes convergence of $\{V_\ell^k(i)\}_{k \geq 0}$ to some $V_\ell^*(i)$. Moreover, the convergence rate is linear

since $\max\{\bar{\delta}_\infty^k, \epsilon_\infty^k\}$ is shown to be contractive in the proof of Theorem 5.1. The exact convergence rate for $\{|r_{\ell,m}^{k+1} - r_{\ell,m}^k|\}_{k \geq 0}$ will depend on B .

Next, we consider error bounds between the limiting $V_\ell^*(i)$ and the optimal $J^*(i)$, satisfying the Bellman equation. For the subsequent result we assume that $B = 0$, i.e., agents communicate with all other agents at all iterations.

Theorem 5.2. *Consider Assumption 5.1 with $B = 0$ and Algorithm 1. For all $\ell = 1, \dots, q$, $i \in \mathcal{X}$, we have that the limit point $V_\ell^*(i)$ of $\{V_\ell^k(i)\}_{k \geq 0}$ satisfies*

$$|V_\ell^*(i) - J^*(i)| \leq \alpha \frac{\delta}{1 - \alpha}, \quad (5.5)$$

where $\delta := \max_{\ell \in \{1, \dots, q\}} \max_{i, j \in I_\ell} |J^*(i) - J^*(j)|$ and J^* is the solution of the Bellman equation.

Proof. The proof is inspired by [138]. Fix $\ell = 1, \dots, q$ and $i \in \mathcal{X}$. We only show that $V_\ell^*(i) \leq J^*(i) + \alpha \frac{\delta}{1 - \alpha}$, as the other side in (5.5) follows symmetric arguments. To this end, for each $i \in I_\ell$, define $\bar{V}_\ell(i) := J^*(i) + \alpha \frac{\delta}{1 - \alpha}$. It follows then that the aggregate of $\bar{V}_\ell(i)$ can be constructed as

$$\begin{aligned} \bar{r}_{\ell, \ell} &:= \sum_{j \in I_\ell} d_{\ell j} \bar{V}_\ell(j) = \sum_{j \in I_\ell} d_{\ell j} (J^*(j) + \alpha \frac{\delta}{1 - \alpha}) \\ &\leq \sum_{j \in I_\ell} d_{\ell j} (\min_{i \in I_\ell} J^*(i) + \delta + \alpha \frac{\delta}{1 - \alpha}) \\ &= \min_{i \in I_\ell} J^*(i) + \delta + \alpha \frac{\delta}{1 - \alpha} = \min_{i \in I_\ell} J^*(i) + \frac{\delta}{1 - \alpha}, \end{aligned} \quad (5.6)$$

where the inequality is since $J^*(j) \leq \min_{i \in I_\ell} J^*(i) + \delta$, while the second last equality is due to $\sum_{j \in I_\ell} d_{\ell j} = 1$.

Denote the Bellman operator induced by the Bellman equation as $\mathcal{T}$, such that we can compactly write the Bellman equation as $J^*(i) = (\mathcal{T}J^*)(i)$, where effectively by $(\mathcal{T}J^*)(i)$ we imply the right-hand side of the Bellman equation which depends on i and on $J^*(j)$ for all $j = 1, \dots, n$. For all $i = 1, \dots, n$, we now consider the

application of the Bellman operator to $\bar{V}_\ell$, i.e.,

$$\begin{aligned}
(\mathcal{T}\bar{V}_\ell)(i) &= \min_{u \in U(i)} \sum_{j=1}^n p_\ell(i, u, j) \left(g_\ell(i, u, j) + \alpha [\phi_{j\ell} \bar{V}_\ell(j) + \sum_{\substack{m=1 \\ m \neq \ell}}^q \phi_{jm} \bar{r}_{\ell,m}] \right) \\
&\leq \min_{u \in U(i)} \sum_{j=1}^n p_\ell(i, u, j) \left(g_\ell(i, u, j) + \alpha [\phi_{j\ell} (J^*(j) + \alpha \frac{\delta}{1-\alpha}) \right. \\
&\quad \left. + \sum_{\substack{m=1 \\ m \neq \ell}}^q \phi_{jm} (\min_{j \in I_m} J^*(j) + \frac{\delta}{1-\alpha})] \right) \\
&\leq \min_{u \in U(i)} \sum_{j=1}^n p_\ell(i, u, j) \left(g_\ell(i, u, j) + \alpha \sum_{m=1}^q \phi_{jm} (J^*(j) + \frac{\delta}{1-\alpha}) \right) \\
&\leq \min_{u \in U(i)} \sum_{j=1}^n p_\ell(i, u, j) \left(g_\ell(i, u, j) + \alpha J^*(j) \right) + \alpha \frac{\delta}{1-\alpha} \\
&= J^*(i) + \alpha \frac{\delta}{1-\alpha} = \bar{V}_\ell(i),
\end{aligned} \tag{5.7}$$

where the first inequality is due to the definition of $\bar{V}_\ell$ and utilises $\bar{r}_{\ell,m} = \bar{r}_{m,m}$, which holds since $B = 0$. The second inequality follows, since $\min_{j \in I_m} J^*(j) \leq J^*(j)$ for all $j \in \mathcal{X}$, and $\alpha \frac{\delta}{1-\alpha} \leq \frac{\delta}{1-\alpha}$, that in turn allows us to combine the terms multiplied with $\phi_{j\ell}$ into the last summation. The last inequality follows from $\sum_{m=1}^q \phi_{jm} = 1$, while the second last equality is due to the Bellman equation.

By (5.7) we have that $(\mathcal{T}\bar{V}_\ell)(i) \leq \bar{V}_\ell(i)$, for all $i = 1, \dots, n$, which implies that $\{\bar{V}_\ell^k(i)\}_{k \geq 0}$ is a non-increasing sequence. Moreover, $\mathcal{T}$ is contractive and as such it will converge to its (unique) fixed point; a direct consequence of Theorem 5.1 is that this fixed point is $V_\ell^*(i)$ (as the latter was constructed by successive applications of the Bellman operator). Therefore, for all $i = 1, \dots, n$, $V_\ell^*(i) = \lim_{k \rightarrow \infty} (\mathcal{T}^k \bar{V}_\ell)(i) \leq \bar{V}_\ell(i)$, thus establishing $V_\ell(i) \leq J^*(i) + \alpha \frac{\delta}{1-\alpha}$, concluding the proof. $\square$

It follows from Theorem 5.2, that if the aggregation sets $I_1, \dots, I_q$ are chosen such that the cost function J^* is expected to vary moderately between states within an aggregation set, then the maximum error compared to $V_1, \dots, V_q$ will be moderate as well. One method to aggregate states with similar costs is to use feature-based aggregation, whereby the aggregation is performed on a set of representative features instead of representative states. This form of aggregation has been thoroughly investigated; we refer to [21, p. 322], [19, p. 68] and references

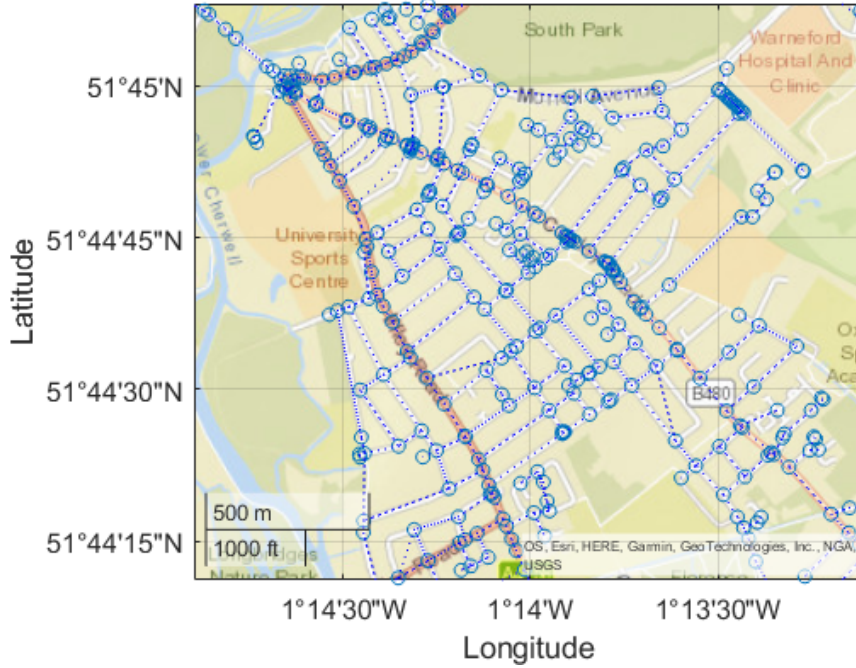


Figure 5.1: Graph representation of the East-Oxford road network. Junctions are represented by vertices and edges represent roads.

therein. For applications with a high discount factor, i.e., $\alpha \ll 1$, the maximum error will also be moderate.

5.4 Application to traffic routing

5.4.1 Simulation Set-up

We demonstrate the proposed algorithm in a traffic routing case study. To this end, we begin by modelling the traffic network as a graph, with vertices representing junctions and edges representing roads connecting junctions. An illustration of such a graph representation for the Oxford road network is shown in Figure 5.1.

We consider a low-energy antenna with a limited range and computation power to be situated at the centre of each network partition, gathering the current speed and location of nearby vehicles. The transition cost of an edge is the expected travel time along that edge and is computed based on the data received by cars, i.e.,

$$g_\ell(i, u, j) = \frac{\text{length of edge}}{\text{average speed of cars on edge}}, \quad (5.8)$$

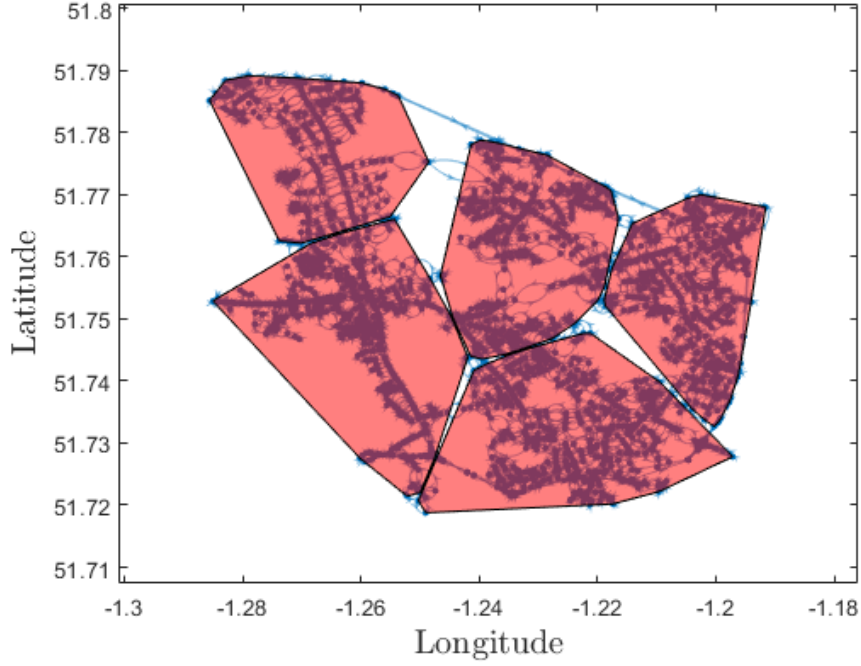


Figure 5.2: The complete Oxford road network subdivided into 5 subgraphs representing the aggregation sets.

where u is used to determine the selected edge between node i and j . Due to the nature of the low-energy application, we utilise the proposed algorithm to limit the amount of data that needs to be sent over longer distances.

As is common in transit node routing, vehicles are trying to reach a common access node, such as a freeway used for long-distance routing. The goal is to find the fastest path to the access point, thus the cost at each vertex is the discounted expected travel time to the access point. In our example related to the Oxford traffic network, the access point will be London Road, leading long-distance travellers toward London.

5.4.2 Simulation results

The Oxford road network is partitioned into 5 subgraphs using K-means clustering of the vertices based on their Euclidean distance, as shown in Figure 5.2.

The cost-to-go is calculated as in (5.8), where the average car speed is randomly generated from a uniform distribution to lie between 25% and 100% of the speed

limit of the relevant road. For each vertex, the outgoing edges are enumerated and the transition probability between two vertices is set to either 1 or 0, depending on whether an edge directly connects the vertices and the input selects that edge.

The disaggregation probabilities are zero for all nodes that have no edge connected to a vertex in another aggregation set I_ℓ . The remaining vertices are given a normalised non-zero disaggregation probability. The discount factor, $\alpha = 0.9$, is chosen close enough to 1 so as to reflect the desire to reach the final node and avoid loops, yet smaller than 1, so as to weight costs further off as less important due to the constantly evolving traffic situation.

Solving the aggregate problem and comparing it to what is considered to be the *true solution* J^* obtained via conventional value iteration, we notice that the normalised average error of the expected cost-to-go, i.e., $\frac{1}{n} \sum_{\ell=1, \dots, q} \sum_{i \in I_\ell} \frac{|\phi_{i\ell} V_\ell(i) - J^*(i)|}{|J^*(i)|}$, is 0.94%, and the normalised maximum error i.e., $\max_{\ell=1, \dots, q} \max_{i \in I_\ell} \frac{|\phi_{i\ell} V_\ell(i) - J^*(i)|}{|J^*(i)|}$, is 190.83%.

The value function is shown in Figure 5.3 and Figure 5.4. The evolution over time of the aggregated costs is shown in Figure 5.5, where the coloured dots represent when the change to an agent's local value function compared to the last broadcast is greater than the chosen communication threshold (0.1 in this example). At this point, the agent will broadcast its updated aggregated cost. If since the last broadcast, no agents value function has changed significantly, the agents signal each other that convergence is reached and the algorithm terminates.

For comparison, we show how, on average, the normalised error increases as we increase the number of agents (see Table 5.1). This is to be expected, as agents will rely more heavily on the aggregated values the smaller their respective partitions become.

Number of Agents	4	8	12	16
Normalised average error	0.67%	1.63%	2.84%	4.46%

Table 5.1: Normalised average error with respect to the number of partitions/agents.

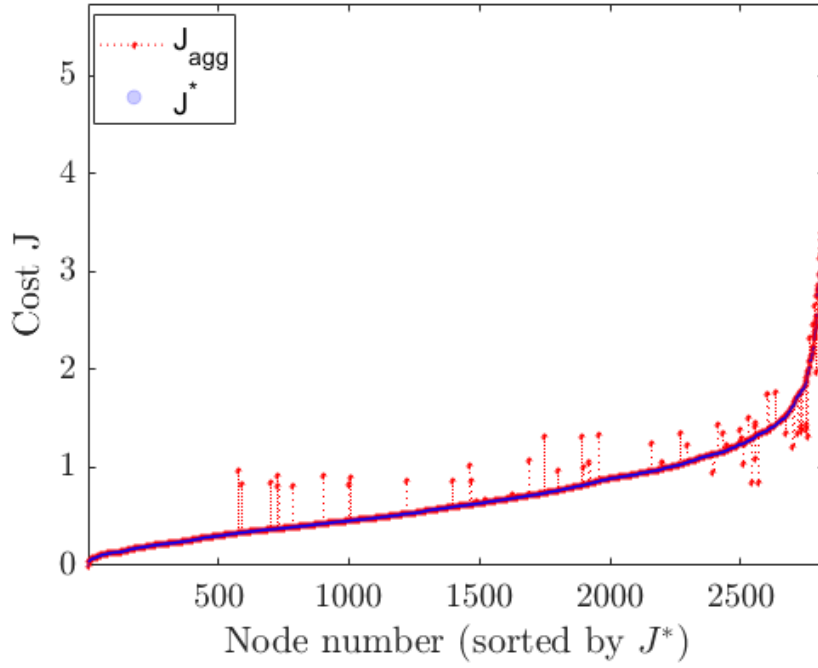


Figure 5.3: The node cost sorted by size. The blue line indicates the *true cost* computed using conventional value iteration, the red markers represent the cost obtained by the proposed distributed algorithm.

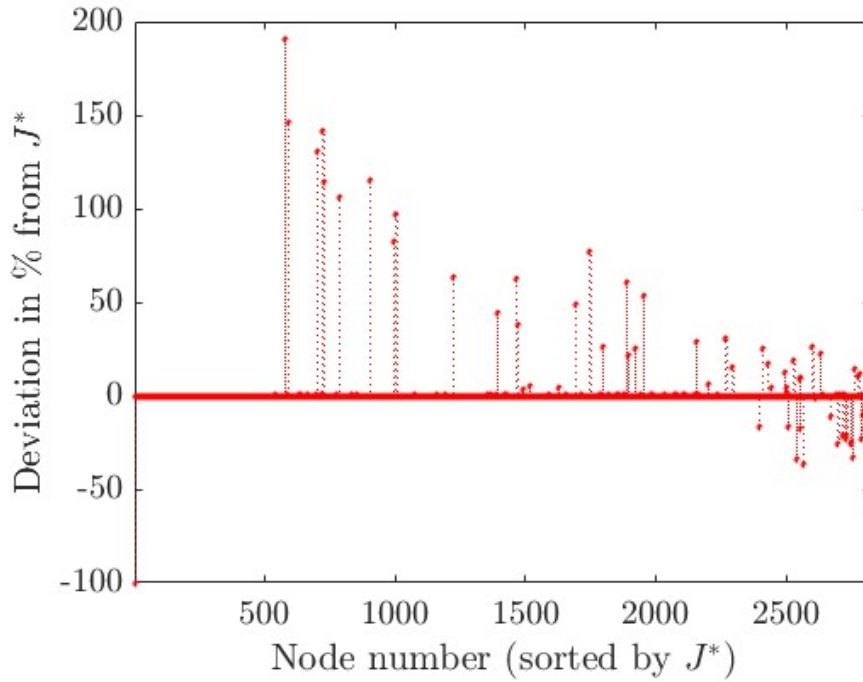


Figure 5.4: The deviation of the cost obtained by the proposed distributed algorithm with respect to the *true cost*. The node numbers correspond to those of Figure 5.3.

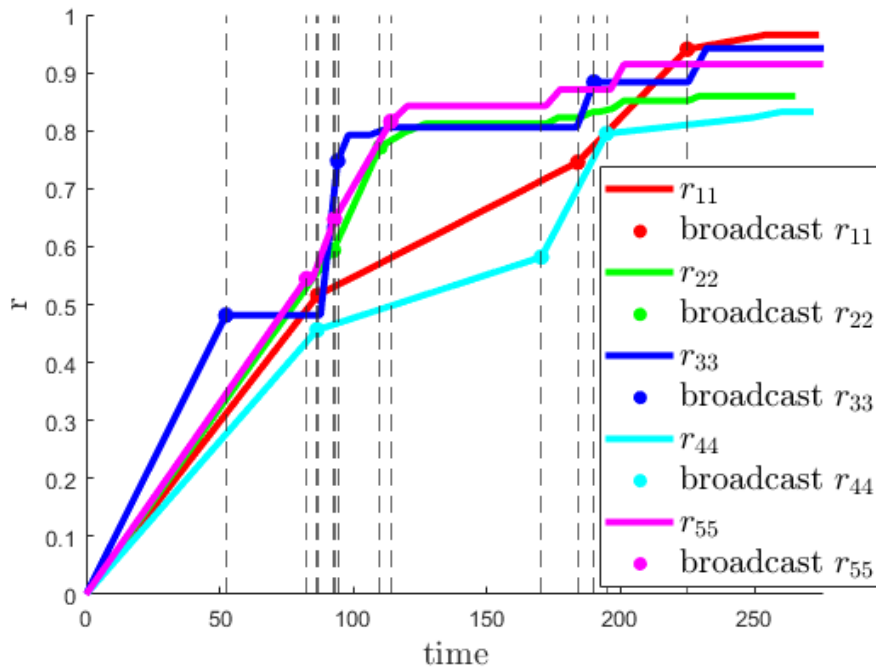


Figure 5.5: The aggregated cost r evolving over time with minimal communication between agents.

To reproduce the numerical results the associated code has been made available ¹ with the ability to upload any OpenStreetMap file, which is then converted to a graph and subsequently a discounted Markov decision process.

5.5 Conclusion

We presented a multi-agent extension of aggregated value iteration which was shown to be able to solve large-scale dynamic programming problems in a fully distributed manner. The presented methodology finds application in problems where each agent has only partial knowledge of the transition probabilities and costs. To this end, we demonstrated its efficacy in a distributed traffic routing problem.

¹www.vertovec.info/code/distributed-traffic-routing

6

Finite sample learning of moving targets

6.1 Introduction

The use of probabilistic and randomised methods to analyse and design uncertain systems has long been a key research area within the control community. Early attempts at dealing with uncertainty were focused on stochastic approaches with later research focusing on *worst-case* settings. Recent research has addressed the interplay of stochastic (soft) and worst-case (hard) performance bounds by utilising algorithms based on uncertainty randomisation, known as *randomised algorithms* [136]. These paradigms involve applying tools from statistical learning theory based on the VC theory to control [4, 136, 149]. In general, these developments can be cast as binary classification problems with the main focus being the provision of finite-sample complexity bounds. VC theoretic techniques do not require the underlying control problem to be written as a convex optimisation program; however, they do require the so called VC dimension to be finite, while its computation is in general a difficult task for generic optimisation problems. Under a convexity assumption, the so called scenario approach has offered a theoretically sound and efficient methodology to provide *a priori* probabilistic feasibility guarantees for uncertain optimisation programs, with uncertainty represented by means of scenarios and without resorting to VC theory [27–29, 32, 34]. These developments

have been recently extended to the non-convex case, however, they typically involve *a posteriori* characterisations [31, 67]. Applications and sample complexity bounds of the aforementioned methodologies to control synthesis problems have been demonstrated in [35, 45, 136], while notable extensions involve trading feasibility to performance [30, 121, 122], applications in game theory [62], and sequential methods [136]. Connections between the scenario approach and statistical learning theory based on the notion of compression have been provided in [33, 99].

A common feature is that the aforementioned approaches can be considered in the context of learning an unknown labelling mechanism, whereby we independently draw m samples from a domain $X \subseteq \mathbb{R}^n$, according to some possibly unknown probability distribution $\mathbb{P}$. Each sample is assigned a $\{0, 1\}$ -valued label according to an unknown *target* labelling function, f . The learning problem involves characterising sample complexity bounds for m , such that we can generate a hypothesis h that, with a prescribed confidence $1 - \delta$, provides the same labelling with the target function when it comes to a new sample x up to a predefined accuracy level ϵ , i.e.,

$$\mathbb{P}^m \{(x_1, \dots, x_m) \in X^m : \mathbb{P}\{x \in X : h(x) \neq f(x)\} \leq \epsilon\} \geq 1 - \delta, \quad (6.1)$$

where $\mathbb{P}^m$ is the product probability measure. An algorithm that generates a hypothesis satisfying the above statement is said to be probably approximately correct (PAC) to accuracy ϵ if the left side of (6.1) approaches 1 as $m \rightarrow \infty$ [149, pg. 56]. We will refer to the labelling mechanism as being probably approximately correct (PAC) learnable to accuracy ϵ if there exists an algorithm that is PAC to accuracy ϵ .

The framework of the proposed tracking problem can be seen in a variety of applications, most notably safety. In the context of safety, the PAC learnable labelling mechanism can be considered to be a fault or safety indicator, which, on the basis of a performance or state measurement, predicts a fault or unsafe condition. Learning the labelling mechanism provides a method to robustify towards uncertainty that could otherwise compromise safety and will thus be the focus for dealing with the third distinct notion of safety considered in this thesis. In Section 6.4 we provide an example of such a safety labelling mechanism

to demonstrate numerically our theoretical developments and further illustrate the application to safety-critical systems.

In this chapter, we will study a similar problem as above of finding a hypothesis satisfying (6.1), however, with the notable difference that we consider the unknown labelling function to change after each drawn sample. In light of this labelling mechanism changing in a structured manner (this structure will be specified in the sequel), we will consider both the construction of the hypothesis as well as the minimum number of samples that are necessary, so as to, with a certain confidence, provide probabilistic bounds on the event of the hypothesis disagreeing with the subsequently received label.

We will begin by providing the mathematical formulation for the tracking problem in Section 6.2, while in Section 6.3 we provide our main contributions:

1. We provide *a-priori* results on the minimum number of samples needed to generate a sufficiently accurate hypothesis.
2. We provide a constructive method of generating a hypothesis from a finite set of samples using a Mixed Integer Linear Program (MILP).

We demonstrate numerically our theoretical results in Section 6.4 on a case study that involves autonomous emergency braking, and discuss further practical improvements to the Mixed Integer Linear Program (MILP) including a sample discarding approach to improve computational feasibility. Finally, Section 6.5 provides some concluding remarks.

6.2 Learning moving targets

6.2.1 Problem statement

We consider the problem of learning a labelling mechanism that is changing in a structured manner (this structure will be specified in the sequel). To this end, we follow a sample-based approach, where each sample x is generated independently from a compact domain $X \subseteq \mathbb{R}^n$, endowed with a σ -algebra $\mathcal{X}$. Let $\mathbb{P}$ denote the fixed

(potentially unknown) probability measure over $\mathcal{X}$. We refer to $(x_1, \dots, x_m) \in X^m$ as an m -multisample, where its elements $x_i \in X$ are independently and identically distributed (i.i.d.) according to $\mathbb{P}$. For each $i = 1, \dots, m$, let $f_i(\cdot) : X \rightarrow \{0, 1\}$ be a $\{0, 1\}$ -valued labelling function. Such a function could be constructed as the indicator function over a given set (see Section 6.3 for a particular construction), i.e., for each $i = 1, \dots, m$, let $B_i \in \mathcal{X}$ be such that

$$f_i(x) = \mathbb{1}_{B_i}(x) = \begin{cases} 1 & \text{if } x \in B_i \\ 0 & \text{otherwise} \end{cases}. \quad (6.2)$$

We refer to these labelling functions as *target functions*. In our setting, each sample x_i is labelled according to a target function f_i , $i = 1, \dots, m$, giving rise to the *labelled m -multisample* $\{(x_1, f_1(x_1)), \dots, (x_m, f_m(x_m))\}$.

Notice that each sample is labelled by means of a different target function. This captures the case where the set B_i is moving/changing with time, hence once a sample from the m -multisample is obtained its label is provided by the corresponding target function. It should be emphasised that the sets B_i (and hence also the target functions f_i), $i = 1, \dots, m$, are unknown and we only have access to the labels of specific samples, namely $\{f_i(x_i)\}_{i=1}^m$.

A natural question that we seek to answer is whether we could construct a labelling mechanism h_m that correctly (with certain probability) predicts the label of the next sample x that would be labelled by means of the unknown target function f_{m+1} (the indicator function over the set B_{m+1} , which is however, not available). In other words, we seek to provide guarantees that $h_m(x) = f_{m+1}(x)$, where h_m is referred to as a *hypothesis* and constitutes an approximation/prediction of f_{m+1} . Notice that we introduce the subscript m to our hypothesis to highlight that this is constructed on the basis of the labelled m -multisample.

We refer to this problem, pictorially illustrated in Figure 6.1, as a tracking problem, as we seek to track a moving labelling mechanism. Denote by $\mathcal{H}$ a class of labelling functions, with all target and hypotheses functions belonging to the same class, i.e., $f_1, \dots, f_m, f_{m+1}, h_m \in \mathcal{H}$. We assume throughout that $\mathcal{H}$ has a finite VC dimension; we refer to Section 6.3.2 and the numerical example of Section 6.4

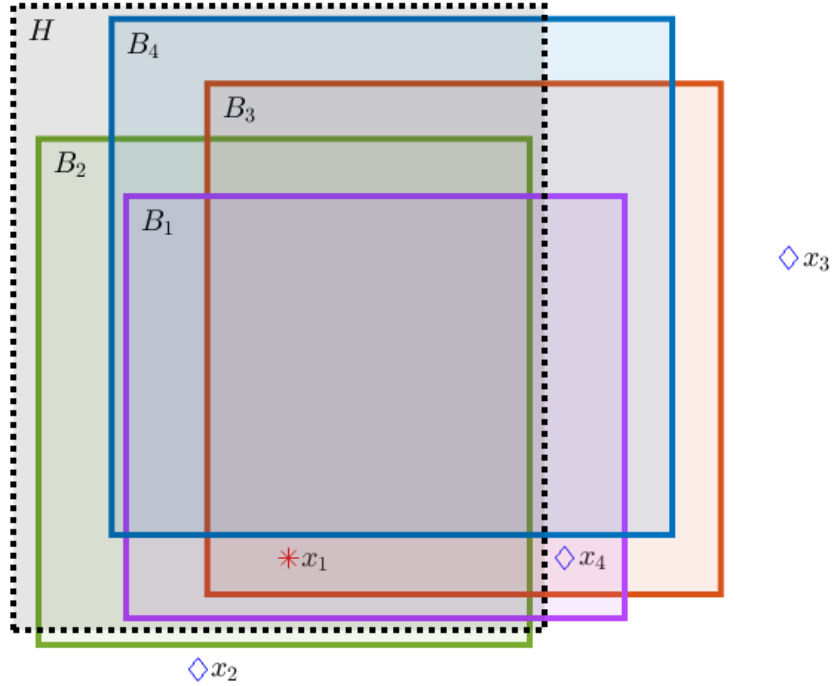


Figure 6.1: For each i , the set B_i changes and we receive a single sample, x_i , labelled either by 1 (red star) or 0 (blue diamond), to indicate if the drawn sample lies within the set B_i . We seek to find a hypothesis, h_m on the basis of m observed samples labelled according to the procedure that involves a moving labelling mechanism, such that with certain confidence we can predict the (unknown) label, f_{m+1} , of the subsequent sample.

for a constructive case where the definition of the class is explicit. We can then formalise the tracking problem as summarised below.

Problem 6.1 (Tracking Problem). *Let $\epsilon, \delta \in (0, 1)$ be any fixed accuracy and confidence level, respectively. Determine $m_0(\epsilon, \delta)$ such that for any number of labelled samples $m \geq m_0(\epsilon, \delta)$, namely, $\{(x_1, f_1(x_1)), \dots, (x_m, f_m(x_m))\}$, we can construct a hypothesis $h_m \in \mathcal{H}$ such that*

$$\mathbb{P}^m\{(x_1, \dots, x_m) \in X^m : \mathbb{P}\{x \in X : h_m(x) \neq f_{m+1}(x)\} \leq \epsilon_0 + \epsilon\} \geq 1 - \delta, \quad (6.3)$$

where $\epsilon_0 \in (0, 1)$.

In words, with confidence at least $1 - \delta$, the probability that the constructed hypothesis h_m produces a label for a new sample x that does not agree with the target function f_{m+1} is at most $\epsilon_0 + \epsilon$. Note that unlike more standard PAC statements, the accuracy is deteriorated by ϵ_0 ; this is not user chosen, but rather

depends on how the target function is moving. We specify this in the next section, and show that its presence is the price to pay for providing such statements for moving targets, while $\epsilon_0 = 0$ for the specific case of a constant target.

Notice that the statement we seek to provide is within the realm of probably approximately correct (PAC) learning. The main difference here compared to traditional PAC statements is that the target function is changing. In the next section we address this problem and show that it is indeed possible to provide such $\epsilon - \delta$ guarantees, provided that the target functions change in a particular manner (see Section 6.2.2).

6.2.2 Mathematical preliminaries and assumptions

To simplify notation, for any labelling functions f, h we define their probabilistic and empirical disagreement, respectively, as

$$\text{er}(f, h) := \mathbb{P}\{x \in X : h(x) \neq f(x)\}, \quad (6.4)$$

$$\hat{\text{er}}_m(f, h) := \frac{1}{m} \sum_{i=1}^m |f(x_i) - h(x_i)|, \quad (6.5)$$

where the empirical disagreement is computed on an m -multisample $x_1, \dots, x_m$, hence we introduce the subscript m in the definition of $\hat{\text{er}}$ to emphasise this dependence. Notice that in (6.5), $|f(x_i) - h(x_i)| = 1$ if f, h disagree on x_i , and zero otherwise. Under these definitions, for $\epsilon, \delta \in (0, 1)$, the statement of (6.3) can be equivalently written as $\mathbb{P}^m\{(x_1, \dots, x_m) \in X^m : \text{er}(f_{m+1}, h_m) \leq \epsilon_0 + \epsilon\} \geq 1 - \delta$.

We first provide some preliminary results that will be invoked in the subsequent developments. Proposition 6.1 below is a direct consequence of Hoeffding's inequality (e.g., see [77], [136]).

Proposition 6.1. *Let $p_1, \dots, p_m \in [0, 1]$, and consider independent Bernoulli random variables $Y_1, \dots, Y_m$ such that $\mathbb{P}\{Y_i = 1\} = p_i$ and $\mathbb{P}\{Y_i = 0\} = 1 - p_i$, for all $i = 1, \dots, m$. For any $\tau > 0$ we then have that*

$$\mathbb{P}^m\left\{\sum_{i=1}^m Y_i - \sum_{i=1}^m p_i \geq \tau\right\} \leq e^{-\frac{2\tau^2}{m}}. \quad (6.6)$$

The following result is a PAC-type bound that holds for any target function $f \in \mathcal{H}$. Its proof involves adapting the definition of the so called probability of one-sided constrained failure ([4, Definition 3]) to our notation, and then directly invoking the result of [4, Theorem 7].

Theorem 6.1. *Fix $\epsilon, \delta \in (0, 1)$ and $\rho \in [0, 1]$. Fix any $f \in \mathcal{H}$, and denote by d the VC dimension of $\mathcal{H}$. For any*

$$m \geq \frac{5(\rho + \epsilon)}{\epsilon^2} \left(\ln \frac{4}{\delta} + d \ln \frac{40(\rho + \epsilon)}{\epsilon^2} \right) \quad (6.7)$$

we have that

$$\begin{aligned} \mathbb{P}^m \{ (x_1, \dots, x_m) \in X^m : \exists h \in \mathcal{H} \text{ such that} \\ \hat{\text{er}}_m(f, h) \leq \rho \text{ and } \text{er}(f, h) > \rho + \epsilon \} \leq \delta. \end{aligned} \quad (6.8)$$

In words, Theorem 6.1 implies that if (as typically is the case) δ is chosen to be small, then the probability that there exists a hypothesis such that its empirical error $\hat{\text{er}}_m(f, h)$ with the target function is at most ρ but the actual error $\text{er}(f, h)$ is higher than a value higher than ρ , namely, $\rho + \epsilon$, is low (at most δ).

For the subsequent developments we consider target functions that exhibit the following structure on the way the labeling is changing, i.e., the target is moving.

Assumption 6.1. *Let $f_1, \dots, f_m, f_{m+1} \in \mathcal{H}$, and consider $\underline{\mu}, \bar{\mu} \in (0, 1)$ with $\underline{\mu} \leq \bar{\mu}$. We assume that*

$$m\underline{\mu} \leq \sum_{i=1}^m \text{er}(f_i, f_{m+1}) \leq m\bar{\mu}. \quad (6.9)$$

Assumption 6.1 implies that the cumulative error between each of the target functions $f_1, \dots, f_m$ and the subsequent one f_{m+1} , whose labelling we want to predict, lives in a bounded interval. In other words, the target sets are changing but we impose a restriction (both upper and lower limits) on the probability that the labelling they produce changes. We refer to $\underline{\mu}, \bar{\mu}$ as the minimum and maximum, respectively, target change.

6.3 Finite sample probabilistic certificates

6.3.1 Main result

Problem 6.1 requires obtaining finite sample complexity bounds such that a hypothesis h_m constructed on the basis of a labelled m -multisample tracks (probabilistically) the moving target function. Such a result does not hold for any hypothesis in $\mathcal{H}$; we show in this section that this is the case for hypotheses in minimal empirical disagreement on the m -multisample. We formalise the set of such hypotheses in the definition below.

Definition 6.1 (minimal disagreement). *Consider a labelled m -multisample $\{(x_1, f_1(x_1)), \dots, (x_m, f_m(x_m))\}$. We refer to the set*

$$M_m := \arg \min_{h \in \mathcal{H}} \frac{1}{m} \sum_{i=1}^m |f_i(x_i) - h(x_i)| \quad (6.10)$$

as the set of hypotheses in $\mathcal{H}$ that minimise the empirical error with the labelled m -multisample. We then say that any $h \in M_m$ is in minimal disagreement with $f_1, \dots, f_m$.

Let h_m be any minimal disagreement hypothesis from the set M_m . We show that for this particular hypothesis choice, we can provide an answer to Problem 6.1. We formalise this in the next theorem, which is the main result of this section.

Theorem 6.2. *Fix $\epsilon, \delta \in (0, 1)$. Denote by d the VC dimension of $\mathcal{H}$, and consider Assumption 6.1. If we choose $m \geq m_0(\epsilon, \delta)$, where*

$$m_0(\epsilon, \delta) = \max \left\{ \frac{1}{2\mu^2} \ln \frac{2}{\delta}, \frac{5(4\bar{\mu} + \epsilon)}{\epsilon^2} \left(\ln \frac{8}{\delta} + d \ln \frac{40(4\bar{\mu} + \epsilon)}{\epsilon^2} \right) \right\}, \quad (6.11)$$

we then have that for any $h_m \in M_m$,

$$\mathbb{P}^m \{ (x_1, \dots, x_m) \in X^m : \text{er}(f_{m+1}, h_m) \leq 4\bar{\mu} + \epsilon \} \geq 1 - \delta. \quad (6.12)$$

Proof. Fix any $\epsilon, \delta \in (0, 1)$, and let $\mu = \sum_{i=1}^m \text{er}(f_i, f_{m+1})$. We define the following events:

$$\begin{aligned} E &= \{ (x_1, \dots, x_m) \in X^m : \text{er}(f_{m+1}, h_m) > 4\bar{\mu} + \epsilon \}, \\ A &= \{ (x_1, \dots, x_m) \in X^m : \sum_{i=1}^m |f_i(x_i) - f_{m+1}(x_i)| > 2\mu \}. \end{aligned} \quad (6.13)$$

A is an approximation set as it includes the m -multisamples for which the empirical cumulative disagreement $\sum_{i=1}^m |f_i(x_i) - f_{m+1}(x_i)|$ is at least twice as big as the actual disagreement μ . E plays the role of the error set, as by its definition, $\mathbb{P}^m\{E\} \leq \delta$ is the complementary statement to that of (6.12).

We can bound $\mathbb{P}^m\{E\}$ as

$$\mathbb{P}^m\{E\} = \mathbb{P}^m\{E \cap A\} + \mathbb{P}^m\{E \cap \bar{A}\} \leq \mathbb{P}^m\{A\} + \mathbb{P}^m\{E \cap \bar{A}\}, \quad (6.14)$$

where $\bar{A}$ denotes the complement of A . The inequality is since $\mathbb{P}^m\{E \cap A\} \leq \mathbb{P}^m\{A\}$. To show (6.12), we can equivalently establish that $\mathbb{P}^m\{E\} \leq \delta$. To achieve this, it suffices to show that $\mathbb{P}^m\{A\} \leq \delta/2$ and $\mathbb{P}^m\{E \cap \bar{A}\} \leq \delta/2$ ¹.

Case $\mathbb{P}^m\{A\} \leq \delta/2$: For each $i = 1, \dots, m$, set $Y_i = |f_i(x_i) - f_{m+1}(x_i)|$ and $p_i = \text{er}(f_i, f_{m+1})$, and notice that $Y_1, \dots, Y_m$ are independent Bernoulli random variables. Under this variables assignment, and selecting $\tau = \mu$, $\mathbb{P}^m\{A\}$ coincides with the left-hand side of (6.6). We then have that

$$\mathbb{P}^m\{A\} \leq e^{-\frac{2\mu^2}{m}} \leq e^{-2m\underline{\mu}^2}, \quad (6.15)$$

where the first inequality is due to Proposition 6.1, and the second one is since $\mu \geq m\underline{\mu}$ by Assumption 6.1.

By inspection of (6.15), to ensure that $\mathbb{P}^m\{A\} \leq \delta/2$, it suffices to show that $e^{-2m\underline{\mu}^2} \leq \delta/2$. By taking the logarithm making m the argument, we conclude that

$$m \geq \frac{1}{2\underline{\mu}^2} \ln \frac{2}{\delta} \implies \mathbb{P}^m\{A\} \leq \frac{\delta}{2}. \quad (6.16)$$

Case $\mathbb{P}^m\{E \cap \bar{A}\} \leq \delta/2$: We have that

$$\begin{aligned} \mathbb{P}^m\{E \cap \bar{A}\} &= \mathbb{P}^m\{(x_1, \dots, x_m) \in X^m : \text{er}(f_{m+1}, h_m) > 4\bar{\mu} + \epsilon \\ &\text{and } \frac{1}{m} \sum_{i=1}^m |f_i(x_i) - f_{m+1}(x_i)| \leq 2\bar{\mu}\}, \end{aligned} \quad (6.17)$$

where the first statement in the right-hand side of (6.17) is event E , and the second one is $\bar{A}$. To see the latter, notice that $\bar{A}$ requires $\frac{1}{m} \sum_{i=1}^m |f_i(x_i) - f_{m+1}(x_i)| \leq 2\frac{\mu}{m}$, and $\mu \leq m\bar{\mu}$ due to Assumption 6.1.

¹Splitting the confidence equally between these two terms is not necessary; further optimising the split would have minor effect on the final sample size bound as the confidence appears inside the logarithm. As such we do not pursue this here to simplify the analysis.

We have assumed that for any m -multisample, h_m is chosen from M_m . By (6.10), since $h_m \in M_m$, $\sum_{i=1}^m |f_i(x_i) - h_m(x_i)| \leq \sum_{i=1}^m |f_i(x_i) - h(x_i)|$ for any $h \in \mathcal{H}$. However, since we also have that $f_{m+1} \in \mathcal{H}$, we have that for any m -multisample,

$$\sum_{i=1}^m |f_i(x_i) - h_m(x_i)| \leq \sum_{i=1}^m |f_i(x_i) - f_{m+1}(x_i)|. \quad (6.18)$$

Moreover, we have that

$$\begin{aligned} \hat{\text{er}}_m(f_{m+1}, h_m) &= \frac{1}{m} \sum_{i=1}^m |f_{m+1}(x_i) - h_m(x_i)| \\ &\leq \frac{1}{m} \sum_{i=1}^m |f_i(x_i) - f_{m+1}(x_i)| + \frac{1}{m} \sum_{i=1}^m |f_i(x_i) - h_m(x_i)| \\ &\leq \frac{2}{m} \sum_{i=1}^m |f_i(x_i) - f_{m+1}(x_i)|, \end{aligned} \quad (6.19)$$

where the equality is due to (6.5), and the first inequality is by adding and subtracting $f_i(x_i)$ in each term in the summation and applying the triangle inequality. The last inequality is due to (6.18).

Since (6.19) holds for any m -multisample, we have that

$$\begin{aligned} \{(x_1, \dots, x_m) \in X^m : \frac{1}{m} \sum_{i=1}^m |f_i(x_i) - f_{m+1}(x_i)| \leq 2\bar{\mu}\} \\ \subseteq \{(x_1, \dots, x_m) \in X^m : \hat{\text{er}}_m(f_{m+1}, h_m) \leq 4\bar{\mu}\}. \end{aligned} \quad (6.20)$$

As a result, by (6.17) and (6.20) we obtain

$$\begin{aligned} \mathbb{P}^m\{E \cap \overline{A}\} \\ \leq \mathbb{P}^m\{(x_1, \dots, x_m) \in X^m : \text{er}(f_{m+1}, h_m) > 4\bar{\mu} + \epsilon \text{ and } \hat{\text{er}}_m(f_{m+1}, h_m) \leq 4\bar{\mu}\}, \end{aligned} \quad (6.21)$$

where, unlike (6.17), (6.21) holds with inequality due to the inclusion in (6.20).

Notice that (6.21) takes the form of (6.8), with f_{m+1} , h_m and $4\bar{\mu}$ in place of f , h and ρ , respectively. Theorem 6.1 with $\delta/2$ in place of δ implies that

$$m \geq \frac{5(4\bar{\mu} + \epsilon)}{\epsilon^2} \left(\ln \frac{8}{\delta} + d \ln \frac{40(4\bar{\mu} + \epsilon)}{\epsilon^2} \right) \implies \mathbb{P}^m\{E \cap \overline{A}\} \leq \frac{\delta}{2}. \quad (6.22)$$

By (6.16) and (6.22), we obtain that if $m \geq m_0(\epsilon, \delta)$, where $m_0(\epsilon, \delta)$ is as in (6.11), we have that $\mathbb{P}^m\{E\} \leq \delta$, thus concluding the proof. $\square$

The proof of Theorem 6.2 is inspired by [72, Theorem 1]. However, that result does not involve two layers of probability and effectively provides a bound on the expectation of the probability of incorrectly tracking the target. Moreover, only an upper bound on the target change is considered in [72]. This is due to the fact that a term similar to $e^{-\frac{2\mu^2}{m}}$ was bounded by $e^{-2m\bar{\mu}^2}$, which is, however, not valid as $\mu \leq m\bar{\mu}$. Here we correct this issue, by introducing a lower bound on the target change and obtaining a bound as in (6.15).

The sample size bound in (6.11) depends polynomially on $1/\epsilon$ and logarithmically on δ . This implies that we could make the confidence $1 - \delta$ high without an unaffordable increase on the number of samples required. Figure 6.2 illustrates the number of samples as a function of ϵ . The dashed line indicates the first term in (6.11), while the solid line the second one. The colour code corresponds to different values of $\underline{\mu}, \bar{\mu}$. For a given accuracy level the minimum number of samples required is denoted by the intersection of the dashed and the solid line (corresponding to a given $\underline{\mu}, \bar{\mu}$). A graph like this could provide guidelines to the decision maker on the achievable accuracy levels. It should be noted that the overall accuracy level for the prediction properties of our hypothesis is $4\bar{\mu} + \epsilon$. Even though ϵ is user-chosen, $\bar{\mu}$ is a property of the target, and as such the labelling mechanism is considered to be PAC learnable to accuracy $4\bar{\mu}$. Therefore, insightful accuracy levels can be achieved if $\bar{\mu}$ is relatively low, i.e., for moderately changing target functions. A few remarks are in order.

Remark 6.1 (Effect of $\underline{\mu}, \bar{\mu}$). *As evident from (6.11), the minimum number of samples that need to be generated is the maximum of two terms: the first one depends only on the minimum target change $\underline{\mu}$, while the second one depends on the maximum target change $\bar{\mu}$. These two sample size bounds that comprise $m_0(\epsilon, \delta)$ emanate from bounding the terms $\mathbb{P}^m\{A\}$ and $\mathbb{P}^m\{E \cap \bar{A}\}$, respectively, by $\delta/2$ each.*

If $\underline{\mu}$ is sufficiently low (the target could move slowly), then the first sample size bound in (6.11) dominates. This implies that we need a sufficiently high number of samples to ensure that $\mathbb{P}^m\{A\} \leq \delta/2$. Intuitively, this implies that if the target could move slowly, then learning the actual probability of change from the empirical

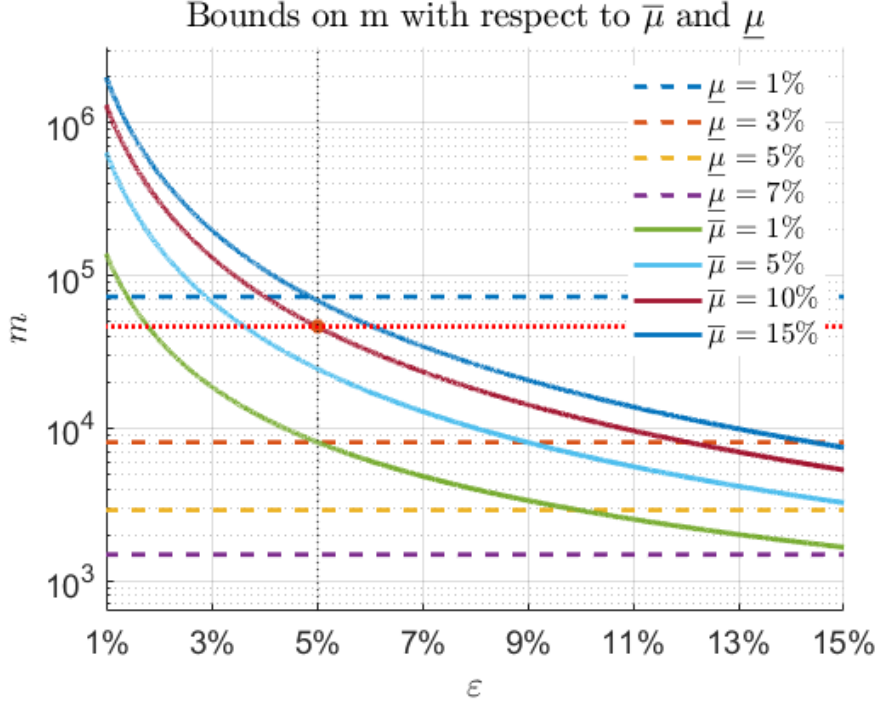


Figure 6.2: Number of samples required according to (6.11) for different accuracy levels ϵ and $\delta = 10^{-6}$. The dashed line indicates the first bound in (6.11), while the solid line the second one. The colour code corresponds to different values of $\underline{\mu}, \bar{\mu}$. For a given accuracy level the minimum number of samples required is denoted by the intersection of these lines, e.g., considering $\epsilon = 5\%$ and $\bar{\mu} = 10\%$, as long as $\underline{\mu} \geq 1.25\%$ (dotted red line), choosing $m \geq 4.7 \cdot 10^4$ samples will result in the desired $\epsilon - \delta$ guarantees.

one (this is encoded in the definition of event A) requires more samples, as with few samples we might get misleading results due to an initial slow change of the target. With reference to Figure 6.2, the minimum number of samples required increases as $\underline{\mu}$ decreases.

Conversely, if we allow for a large change of the target, encoded by a large $\bar{\mu}$, we need a sufficiently high number of samples to, with high confidence, reject the event that the true change with respect to the subsequent label, f_{m+1} , is not considerably lower than the observed, empirical, change. This is captured by ensuring $\mathbb{P}^m\{E \cap \bar{A}\} \leq \delta/2$. With reference to Figure 6.2, for any fixed ϵ , the minimum number of samples required increases as $\bar{\mu}$ increases.

To reject both events $E \cap \bar{A}$ and A , we take the maximum, ensuring that in both a fast and slow-moving target case, the empirical observed change, will with high likelihood correspond to the true probability of change.

Remark 6.2 (Constant target). *The case of a constant target can be obtained as a direct byproduct of the proof of Theorem 6.2. To see this, notice that a constant target implies that $\underline{\mu} = \bar{\mu} = 0$, i.e., if all target functions are the same, their mutual error is zero. In this case, $\mathbb{P}^m\{A\} = \mathbb{P}^m\{(x_1, \dots, x_m) \in X^m : \sum_{i=1}^m |f_i(x_i) - f_{m+1}(x_i)| > 0\} = 0$, as all targets functions are equal and hence necessarily produce the same label. Therefore, we only need to bound $\mathbb{P}^m\{E \cap \bar{A}\}$ in the right-hand side of (6.14) by δ rather than $\delta/2$. This can be achieved as in the second case (see (6.22)) in the proof of Theorem 6.2, replacing $\delta/2$ with δ , and setting $\bar{\mu} = 0$ as the target is not moving. This results in*

$$m \geq \frac{5}{\epsilon} \left(\ln \frac{4}{\delta} + d \ln \frac{40}{\epsilon} \right). \quad (6.23)$$

This sample size bound is then identical to that of Theorem 6.1 with $\rho = 0$. Notice that in that case, and since h_m is chosen to be in minimal disagreement with the (constant) target, then $\hat{\text{er}}_m(f, h_m) = \rho = 0$, as zero error is the minimum if the target is constant. Therefore,

$$\begin{aligned} & \mathbb{P}^m\{(x_1, \dots, x_m) \in X^m : \text{er}(f, h_m) > \epsilon\} \\ & \leq \mathbb{P}^m\{(x_1, \dots, x_m) \in X^m : \exists h \in \mathcal{H} \text{ such that } \hat{\text{er}}_m(f, h) = 0 \text{ and } \text{er}(f, h) > \epsilon\} \\ & \leq \delta. \end{aligned} \quad (6.24)$$

This implies, that when it comes into providing guarantees for the minimal disagreement hypothesis and for the case where the target is constant, Theorem 6.2 specialises to the result of Theorem 6.1 with $\rho = 0$. With reference to Problem 6.1, notice also that in this case, $\epsilon_0 = 4\bar{\mu} = 0$.

6.3.2 Hypothesis computation

We now consider the construction of the hypothesis h_m that minimises the empirical error with respect to the labelled m -multisample, i.e., to determine $h_m \in M_m$. To this end, we consider the labelling functions f_i , $i = 1, \dots, m$, to be defined as in (6.2), with the sets B_i , $i = 1, \dots, m$, being non-empty convex polytopes in $\mathbb{R}^n$, each of them having $n_f = 2^n$ facets. As the hypothesis belongs to the same

class with the target functions, we seek to find a convex polytope, denoted by B_∞ , such that the hypothesis h_m defined as

$$h_m(x) = \mathbb{1}_{B_\infty}(x) = \begin{cases} 1 & \text{if } x \in B_\infty \\ 0 & \text{otherwise} \end{cases}, \quad (6.25)$$

is in minimal disagreement with the observed labels. Since B_∞ is a convex polytope we represent it by means of n_f linear inequality constraints as $Ax + b \leq 0$, where $A \in \mathbb{R}^{n_f \times n}$ and $b \in \mathbb{R}^{n_f}$. Denote each row-vector of A (b) by a_j (b_j), $j = 1, \dots, n_f$. For each $j = 1, \dots, n_f$, $a_j x + b_j$ denotes then a facet of B_∞ . We make this parameterisation explicit by denoting the convex polytope as $B_\infty(A, b)$. Moreover, we assume that for each $j = 1, \dots, n_f$, $(a_j^\top, b_j) \in C_j \subset \mathbb{R}^{n_f+1}$, where C_j is some arbitrarily large compact set that contains the origin in each interior. The purpose of the set will become clear in the sequel.

We show how to construct a Mixed Integer Linear Program (MILP) that will in turn return the parameterisation of B_∞ , namely A and b , that result in a hypothesis $h_m \in M_m$. To this end, let I_1 and I_0 be the set of sample indices for which the label is 1 and 0, respectively, i.e.,

$$I_1 = \{i \in \{1, \dots, m\} \text{ such that } f_i(x_i) = 1\}, \quad (6.26)$$

$$I_0 = \{i \in \{1, \dots, m\} \text{ such that } f_i(x_i) = 0\}. \quad (6.27)$$

We instantiate the MILP that returns the minimal disagreement hypothesis in the following main steps:

1. *Disagreement with the sample indices in I_1 .* Fix any $i \in I_1$, and let x_i be the associated sample. Fix also a parameterisation A, b of B_∞ . If $h_m(x_i) = f_i(x_i) = 1$, i.e., the label that a hypothesis constructed on the basis of $B_\infty(A, b)$ provides on x_i agrees with that of f_i , then $x_i \in B_\infty(A, b)$ since $i \in I_1$. We thus have that

$$x_i \in B_\infty(A, b) \iff a_j x_i + b_j \leq 0, \quad \forall j = 1, \dots, n_f. \quad (6.28)$$

However, we are seeking a hypothesis that is in minimal disagreement with the samples, rather than in zero disagreement. As such, we want to allow for a certain number of incorrect labels, or equivalently, we want to allow violating the right-hand

side of (6.28). Therefore, we introduce the slack variables $s_{ij} \geq 0$, $j = 1, \dots, n_f$, $i = 1, \dots, m$. As such, for each $i \in I_1$, we consider the relaxed constraints

$$a_j x_i + b_j \leq s_{ij}, \quad \forall j = 1, \dots, n_f. \quad (6.29)$$

By means of (6.28) and the definition of h_m , enforcing (6.29), implies that

$$\begin{cases} h_m(x_i) \neq f_i(x_i) & \text{if } \sum_{j=1}^{n_f} s_{ij} > 0, \\ h_m(x_i) = f_i(x_i) & \text{otherwise.} \end{cases} \quad (6.30)$$

In words, if $\sum_{j=1}^{n_f} s_{ij} > 0$ (which is satisfied if at least one s_{ij} , $j = 1, \dots, n_f$ is positive as the slack variables are non-negative) implies that the hypothesis h_m disagrees with the target function f_i on the sample x_i . If all slack variables are zero, then h_m agrees with f_i on x_i , $i \in I_1$.

2. *Disagreement with the sample indices in I_0 .* Fix any $i \in I_0$, and let x_i be the associated sample. Fix also a parameterisation A, b of B_∞ . If $h_m(x_i) = f_i(x_i) = 0$, i.e., the hypothesis and the target function f_i agree on x_i , then $x_i \notin B_\infty(A, b)$. This exclusion can imply that the sample x_i would violate the half-space constraint encoding the facets of $B_\infty(A, b)$ for at least one facet. This can be written as a logical constraint; employing the developments of [16, 105], we equivalently reformulate it to mixed-integer inequalities by introducing the binary variables $z_{ij} \in \{0, 1\}$, $j = 1, \dots, n_f$, $i = 1, \dots, m$. Fix $\eta > 0$ arbitrarily small and let $M_j = \sup_{x \in X, (a_j^\top, b_j) \in C_j} a_j^\top x + b_j$, $m_j = \inf_{x \in X, (a_j, b_j) \in C_j} a_j^\top x + b_j$, $j = 1, \dots, m$. Note that these exist and are finite, as X and C_j , $j = 1, \dots, n_f$, are assumed to be compact. We then have that

$$x_i \notin B_\infty(A, b) \iff \begin{cases} a_j x_i + b_j \leq M_j(1 - z_{ij}), \quad \forall j = 1, \dots, n_f, \\ a_j x_i + b_j \geq \eta + (m_j - \eta)z_{ij}, \quad \forall j = 1, \dots, n_f, \\ \sum_{j=1}^{n_f} z_{ij} \leq n_f - 1. \end{cases} \quad (6.31)$$

Notice that if $z_{ij} = 0$, then the first inequality in (6.31) becomes $a_j x_i + b_j \leq M_j$ (trivially satisfied by the definition of M_j), while the second one reduces to $a_j x_i + b_j \geq \eta$. The latter implies then that $x_i \notin B_\infty(A, b)$ as it violates the constraint of its j -th facet. On the contrary, if $z_{ij} = 1$, then the first inequality in (6.31) implies that x_i is within the half-space defined by the j -th facet of $B_\infty(A, b)$. For x_i to be

inside $B_\infty(A, b)$, i.e., $x_i \in B_\infty(A, b)$, this has to be the case for all $j = 1, \dots, n_f$, or equivalently $\sum_{j=1}^{n_f} z_{ij} = n_f$. This justifies the last constraint in (6.31), which implies that the sample x_i is outside of at least one facet of $B_\infty(A, b)$, hence $x_i \notin B_\infty(A, b)$.

Since we only seek a hypothesis in minimal (rather than in zero) disagreement with the target functions, we relax these constraints by introducing slack variables $s_{ij} \geq 0$, $j = 1, \dots, n_f$, $i = 1, \dots, m$. As such, for each $i \in I_0$, the associated relaxed constraints are given by

$$\begin{cases} a_j x_i + b_j \leq M_j(1 - z_{ij}), \quad \forall j = 1, \dots, n_f, \\ a_j x_i + b_j \geq \eta + (m_j - \eta)z_{ij} - s_{ij}, \quad \forall j = 1, \dots, n_f, \\ \sum_{j=1}^{n_f} z_{ij} \leq n_f - 1. \end{cases} \quad (6.32)$$

Notice that we do not need to introduce a slack variable in the first inequality in (6.32), as this becomes non-redundant only if $z_{ij} = 1$. In this case, however, satisfying the resulting inequality would already mean disagreeing with the target, so we do not need to relax that condition.

By means of (6.31) and the definition of h_m , enforcing (6.32) leads to the same disagreement implications as in (6.30). It should be remarked that in this case, casting $\sum_{j=1}^{n_f} s_{ij} > 0$ as a disagreement results in upper-bounding the exact number of disagreements. To see this, notice that if $z_{ij} = 0$ (then the second inequality in (6.32) becomes non-redundant), we have that $a_j x_i + b_j \geq \eta - s_{ij}$. Due to the presence of $\eta > 0$, if $s_{ij} > 0$ but $\eta - s_{ij} > 0$, then x_i may still be outside of a facet of a convex polytope thus agreeing with the target (recall that label agreement here means being outside $B_\infty(A, b)$) despite the fact that the associated slack is non-zero.

3. Minimising disagreements. In view of constructing the hypothesis that is in minimal disagreement with the target functions, we need to be able to count the number of disagreements. However, if $i \in I_1$ we have a disagreement if $x_i \notin B_\infty(A, b)$, while if $i \in I_0$ we have a disagreement if $x_i \in B_\infty(A, b)$. By (6.30) (recall that the same conclusion holds also for the case where disagreement with the samples with indices in I_0), the hypothesis disagrees with the target on the sample indexed by i , if $\sum_{j=1}^{n_f} s_{ij} > 0$. Therefore, we introduce the binary

variable $v_i \in \{0, 1\}$, $i = 1, \dots, m$, defined as

$$v_i = \begin{cases} 1 & \text{if } \sum_{j=1}^{n_f} s_{ij} > 0, \\ 0 & \text{otherwise.} \end{cases} \quad (6.33)$$

The total number of disagreements that we then seek to minimise is given by $\sum_{i=1}^m v_i$.

For each $j = 1, \dots, n_f$, we have assumed that $(a_j^\top, b_j) \in C_j$, where C_j is compact and contains the origin in its interior. As such, $M_j > 0$ and $m_j < 0$ for all $j = 1, \dots, n_f$. Therefore, by (6.29) and the definition of M_j , $s_{ij} \leq M_j$, for all $i \in I_1$. Similarly, by (6.32) and the definition of m_j , $s_{ij} \leq -m_j + \eta$, for all $i \in I_0$. Summing the across $j = 1, \dots, n_f$, we obtain

$$\begin{cases} \sum_{j=1}^{n_f} s_{ij} \in [0, \sum_{j=1}^{n_f} M_j], & \text{if } i \in I_1 \\ \sum_{j=1}^{n_f} s_{ij} \in [0, -\sum_{j=1}^{n_f} m_j + n_f \eta], & \text{if } i \in I_0. \end{cases} \quad (6.34)$$

The logical implication in (6.33) can be then reformulated as

$$\begin{cases} \sum_{j=1}^{n_f} s_{ij} - v_i \sum_{j=1}^{n_f} M_j \leq 0, & \text{if } i \in I_1, \\ \sum_{j=1}^{n_f} s_{ij} - v_i \left(-\sum_{j=1}^{n_f} m_j + n_f \eta \right) \leq 0, & \text{if } i \in I_0. \end{cases} \quad (6.35)$$

To see the equivalence between (6.35) and (6.33), consider the former inequality in (6.35). Notice that if $\sum_{j=1}^{n_f} s_{ij} > 0$ then this implies that we must have $v_i \sum_{j=1}^{n_f} M_j > 0$ which, since $M_j > 0$ for all $j = 1, \dots, n_f$, implies that $v_i = 1$. On the other hand, if $\sum_{j=1}^{n_f} s_{ij} = 0$, then (6.35) implies that $v_i \sum_{j=1}^{n_f} M_j \geq 0$. However, since we are seeking the minimal disagreement hypothesis and hence we will be minimising $\sum_{i=1}^m v_i$, the minimum value of v_i for which the previous inequality is satisfied is $v_i = 0$. A similar reasoning applies also to the equivalence between the second inequality in (6.35) and (6.33).

4. Minimal disagreement MILP. The MILP that results in a hypothesis that is in minimal disagreement with respect to the target functions on the m -multisample,

i.e. $h_m \in M_m$, is given by:

$$\underset{A, b, \{\{z_{ij}, s_{ij}\}_{j=1}^{n_f}\}_{i=1}^m, \{v_i\}_{i=1}^m}{\text{minimise}} \sum_{i=1}^m v_i \quad (6.36)$$

subject to

$$\forall i \in I_1 : \begin{cases} a_j x_i + b_j \leq s_{ij}, \quad \forall j = 1, \dots, n_f, \\ \sum_{j=1}^{n_f} s_{ij} - v_i \sum_{j=1}^{n_f} M_j \leq 0, \end{cases} \quad (6.37)$$

$$\forall i \in I_0 : \begin{cases} a_j x_i + b_j \leq M_j(1 - z_{ij}), \quad \forall j = 1, \dots, n_f, \\ a_j x_i + b_j \geq \eta + (m_j - \eta)z_{ij} - s_{ij}, \\ \sum_{j=1}^{n_f} z_{ij} \leq n_f - 1, \\ \sum_{j=1}^{n_f} s_{ij} - v_i \left(- \sum_{j=1}^{n_f} m_j + n_f \eta \right) \leq 0. \end{cases} \quad \forall j = 1, \dots, n_f, \quad (6.38)$$

The constraints in (6.37) correspond to (6.29) and the first inequality in (6.35), encoding (relaxed) agreement on the sampled with $i \in I_1$, and determining disagreements for this case, respectively. Similarly, the constraints in (6.38) correspond to (6.32) and the second inequality in (6.35), and admit a similar interpretation. The objective function minimises the total number of disagreements $\sum_{i=1}^m v_i$. We use the volume of the convex polytope parameterised by A, b , namely, $\text{vol}(A, b)$, as a tie-break rule to single out a unique solution. Once the optimal A, b is determined, we can construct $B_\infty(A, b)$, and hence h_m by means of (6.25).

6.4 Numerical example

6.4.1 Problem set-up

We demonstrate numerically our theoretical developments on a case study that involves Autonomous Emergency Braking (AEB) systems. Furthermore, we consider the computational feasibility of the MILP and introduce a sample discarding approach to reduce the constraints of the MILP.

Let us consider a car driving along a road while receiving measurements of the distance to any vehicle or obstacle ahead, l , as well as the current velocity of our car, v . If the braking distance at the current velocity exceeds the available distance to the car or obstacle ahead, we want the AEB system to engage the brakes autonomously. The necessary braking distance in case of an emergency

stop can be calculated by setting the braking force times the distance equal to the kinetic energy of our vehicle. Thus if

$$\frac{1}{2}v^2\frac{\mathcal{M}}{F} \leq l, \quad (6.39)$$

where $\mathcal{M}$ is the vehicle weight and F is the braking force, then there is a sufficient distance to the vehicle or obstacle ahead, hence the corresponding measurement is classified as safe. In view of (6.39) depending on v^2 , hereafter we consider $x = (l, v^2)$ as the measurement vector.

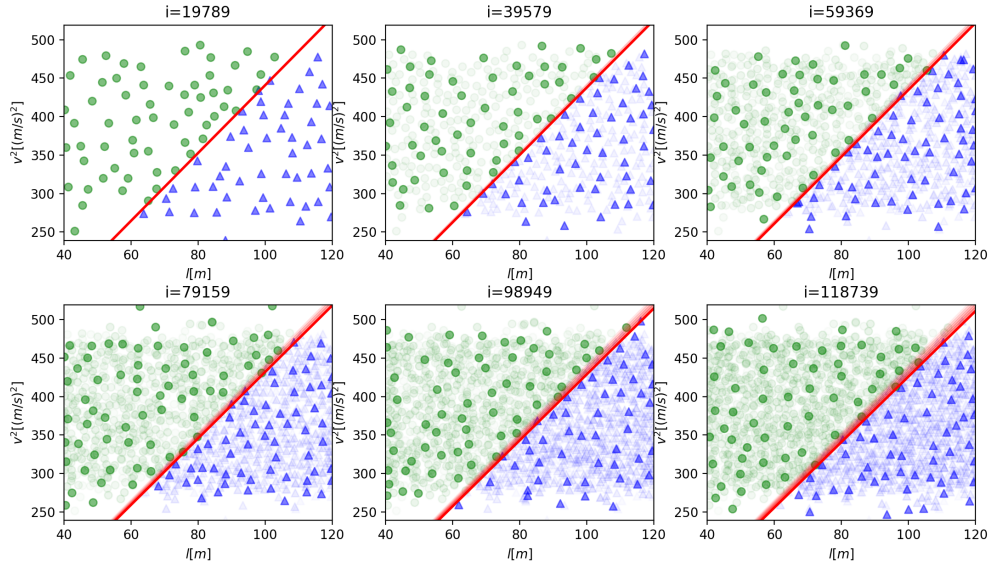


Figure 6.3: The evolution of the braking performance over time. Green circles indicate samples with label 0, while blue triangles show samples with label 1. The bold red halfplane represents the true safety label at the given iteration, while the opaque halfplanes show the safety boundary at previous iterations.

The braking force will depend on the friction coefficient of the brakes and will deteriorate over time. Similarly, the vehicle mass will depend on the fuel, passengers, and cargo, which will also change over time. In line with our theoretical developments, we consider x_i , $i = 1, \dots, m$ to be independent measurements, with the index i acting as a time-stamp. Let F_i denote the corresponding braking force, which depends on i to reflect the change of the friction coefficient, and let $\mathcal{M}_i$ denote the vehicle mass, which will also depend on i to reflect changes to the vehicle

mass. This dependence of F_i and $\mathcal{M}_i$ on i induces a different labelling function f_i . In particular, we label a sample $x = (l, v^2)$ by means of

$$f_i(x) = \begin{cases} 1 & \text{if } \frac{1}{2}v^2 \frac{\mathcal{M}_i}{F_i} \leq l \\ 0 & \text{otherwise.} \end{cases} \quad (6.40)$$

For the construction of the hypothesis, we collect a measurement x after each engagement of the vehicle's brakes. In addition to obtaining measurements x , we assume to obtain a brake performance measurement $\frac{\mathcal{M}_i}{F_i}$ from which to construct the label f_i . Furthermore, we assume that we have knowledge of the expected minimum and maximum degradation of the braking performance, allowing us to obtain values for $\underline{\mu}$ and $\bar{\mu}$, respectively.

Using numerical values for the braking parameters and vehicle mass as defined in the sequel, the evolution of the braking performance is shown in Figure 6.3. For visual clarity, we only depict a random subset of the samples.

For an effective AEB system, we will want to classify a future measurement x as safe or unsafe without having to first engage the brakes to receive a measurement of the new braking performance. In light of this, we will utilise the results from Section 6.3 to construct a hypothesis on the basis of a labelled m -multisample, allowing us to *a-priori* classify a sample x as safe or unsafe.

Aligned with the theoretical developments of Section 6.3.2 we consider our hypothesis to be a convex polytope in the 2D plane as illustrated in Figure 6.4. We assume that matrix A parameterising the convex polytope is fixed, and is given by

$$A = \begin{bmatrix} \cos \theta & \sin \theta \\ -\sin \theta & \cos \theta \\ -\cos \theta & -\sin \theta \\ \sin \theta & -\cos \theta \end{bmatrix}, \quad (6.41)$$

where $\theta := \tan^{-1} \frac{\mathcal{M}}{2F}$ denotes the rotation of the convex polytope. Since the safety label is defined by a single half-plane, only one of the facets of the convex polytope becomes relevant, namely $a_3 = [-\cos \theta \ -\sin \theta]$. This observation reduces the VC-dimension (employed in the sample complexity bounds) to $d = 1$. Considering the evolution of the braking performance as shown in Figure 6.3, the rotation of the convex polytope is minimal. Since the inclusion of the variable rotation

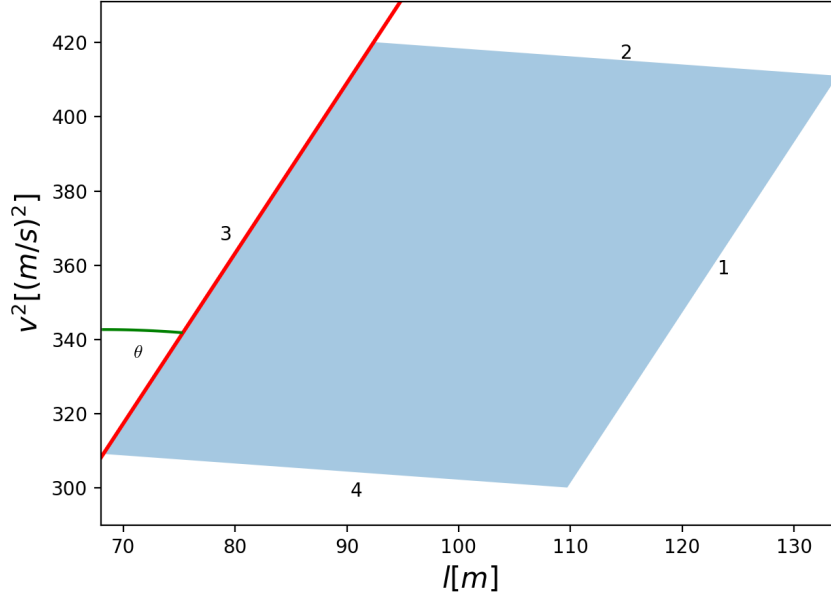


Figure 6.4: Illustration of the facets of the convex polytope. Since the safety label relies in this case on a single halfplane (drawn in red), we only need to consider facet 3.

introduces a nonlinearity into the MILP, for the sake of clarity, we will consider the angle θ to be fixed in the subsequent computation of the hypothesis (however leave the true safety label unchanged).

To reduce the size of the MILP (6.36 - 6.38), we will consider how samples can be discarded prior to computing the hypothesis. Recall that the MILP minimises the total number of disagreements between the hypothesis and the sample labels. Thus, at best, we can obtain zero disagreement between the hypothesis and the samples in I_1 . For the AEB example, this implies that the halfplane constructed by the hypothesis will need to lie to the left of all samples in I_1 . This is illustrated in Figure 6.5 by the cyan dotted line. However, any halfplane that lies further to the left of this line, would unnecessarily label samples from I_0 with label *one*, increasing the total number of label disagreements. However, the constructed MILP aims at minimising the total number of label disagreements; as such the cyan-dotted halfplane would always be preferable, rendering any samples in the cyan-coloured area redundant. If we have non-zero disagreement with respect to the samples from I_1 , the halfplane will lie further to the right of the cyan dotted

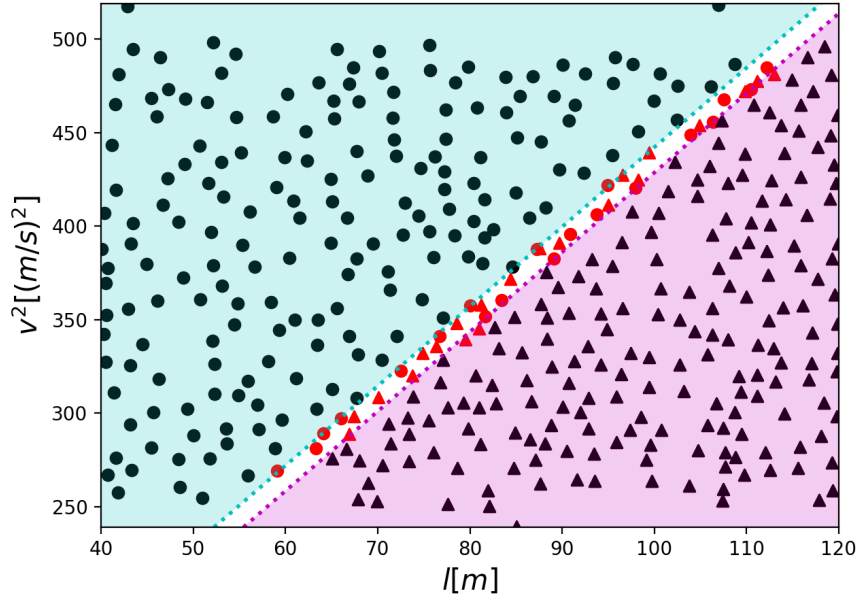


Figure 6.5: All samples in black are discarded, while the red samples are kept for the computation of the hypothesis. This results in 95% of the samples being omitted, greatly improving the computational feasibility.

one. Similarly, to obtain zero disagreement between the hypothesis and the samples in I_0 , the halfplane constructed by the hypothesis will need to lie to the right of all samples in I_0 . Following a similar argumentation as before, any sample in the magenta-coloured area will not change the solution of the MILP.

Since we know that the samples in both the blue and the magenta regions will not affect the solution of the MILP, we can discard these samples prior to computing the hypothesis, resulting in only the red samples in Figure 6.5 being considered. Following similar argumentation for each facet of the convex polytope, the discarding approach is applicable also for higher-order convex polytopes.

6.4.2 Simulation results

While no knowledge of the distribution of the samples (l, v^2) needs to be known for generating the hypothesis, for simulation purposes we draw l from a uniform distribution over the interval $[40m, 120m]$ and draw v^2 from a normal distribution with mean $\bar{v^2} = (70km/h)^2$ and standard deviation $\sigma_{v^2} = (20km/h)^2$. The

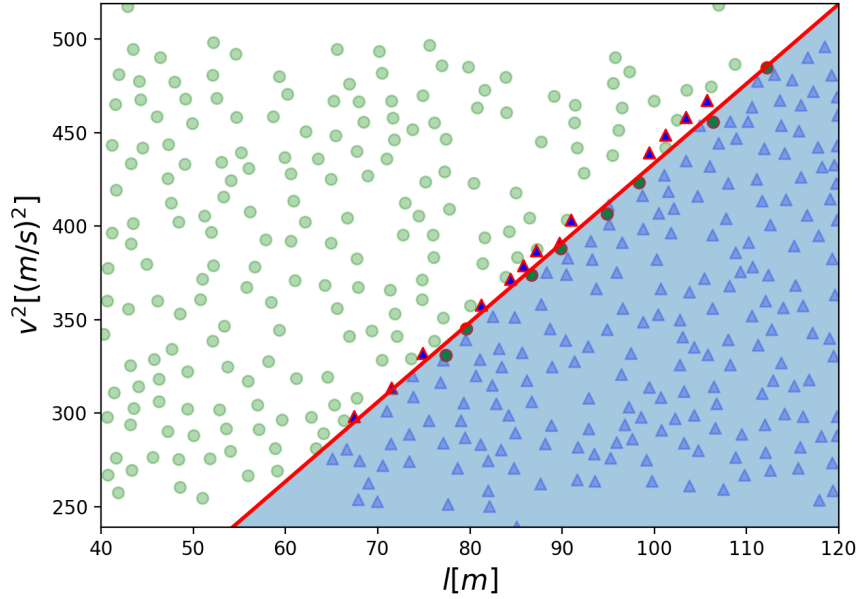


Figure 6.6: The generated hypothesis; we only show the halfplane responsible for the labelling, illustrated by red. For visual ease, we randomly omit samples close to one another. Red samples are violations as defined in (6.33).

performance of the brakes at each time step will deteriorate by a factor of ω_F , i.e. $F_{i+1} = \omega_F F_i$, where ω_F is a random variable drawn from a normal distribution with mean $\mu = (1 - 3 \cdot 10^{-7})$ and standard deviation $\sigma = 10^{-6}$. The initial car mass is $\mathcal{M} = 900kg$ and will randomly change by a factor of $\omega_{\mathcal{M}}$, where $\omega_{\mathcal{M}}$ is a random variable drawn from a normal distribution with mean $\mu = 1$ and standard deviation $\sigma = 10^{-3}$.

For the construction of the hypothesis, the confidence level is chosen as $\delta = 10^{-6}$ with an accuracy of $\epsilon = 1\%$. For the satisfaction of Assumption 6.1, we choose $\frac{1}{m} \sum_{i=1}^m \text{er}(f_i, f_{m+1})$ to be bounded by $\bar{\mu} \leq 2\%$ and $\underline{\mu} \geq 0.78\%$. By Theorem 6.2 it then follows that we need at least 119,237 samples to accurately predict the safety label of the subsequent timesteps.

Using the aforementioned discarding approach, we can discard 95% of the samples prior to instantiating the MILP constraints. The discarding approach is illustrated in Figure 6.5 where, for the purpose of visualisation, we omit samples close to one another to prevent the image from being cluttered.

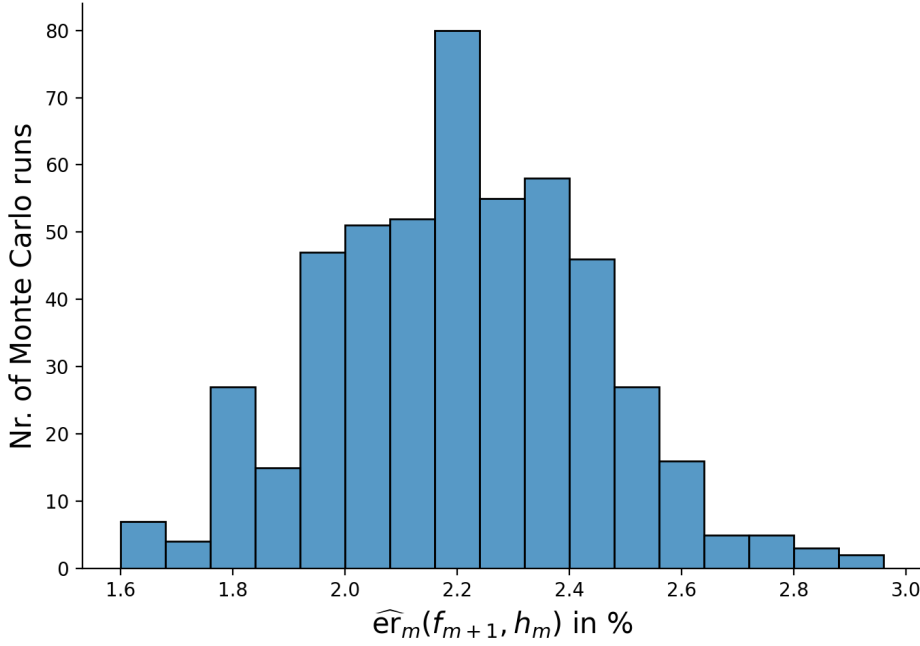


Figure 6.7: Empirical distribution of the disagreement $\hat{er}_m(f_{m+1}, h_m)$, constructed by means of 500 Monte Carlo runs.

The hypothesis in minimal disagreement with the labelled samples, computed by means of the MILP (6.36 - 6.38), is shown in Figure 6.6. Solving the MILP took 993 seconds, making the deployment of the approach computationally feasible. The number of violations, v , is 1391. We have made all code for generating and reproducing our results available online².

We empirically validate our risk level by means of Monte Carlo simulations. For each run, we generate a new labelling mechanism f_{m+1} , corresponding to the random deterioration of the braking force and change to the vehicle mass. We then draw 5000 samples for which we evaluate the corresponding label (by means of f_{m+1}) and compare this with the label assigned by means of the hypothesis constructed by our methodology, thus calculating $\hat{er}_m(f_{m+1}, h_m)$. We repeat this for 500 runs, each time generating a new label f_{m+1} . In Figure 6.7 the frequency of certain $\hat{er}_m(f_{m+1}, h_m)$ values is shown. Recall that μ was upper bounded by 2%, such that for the chosen $\epsilon = 1\%$, Theorem 6.2 implies that $er(f_{m+1}, h_m) \leq 4\bar{\mu} + \epsilon = 9\%$ with high confidence. The Monte Carlo simulation supports this, with the average

²www.vertovec.info/code/learning-moving-targets

empirical disagreement $\widehat{\text{er}}_m(f_{m+1}, h_m)$ being approximately 2.2% (see Figure 6.7), well below the theoretically predicted 9%.

6.5 Conclusion

We considered learning a moving target from a finite set of samples and show that, when the labelling mechanism changes in a structured manner, it remains PAC learnable, meeting certain accuracy-confidence levels. Furthermore, for the class of convex polytopes, we present a constructive method to generate the hypothesis based on a Mixed Integer Linear Program (MILP) thus rendering our methodology constructive. We illustrated the applicability of our theoretical developments to a case study involving an Autonomous Emergency Braking (AEB) system.

7

Summary and future research directions

Safety-critical control for complex and high-dimensional systems continues to pose significant challenges within the control theory literature. This thesis has considered three distinct notions of safety and has presented application-specific as well as generalisable results towards dealing with scalability, computational efficiency and data sparsity. Chapters 2-4 consider the *classical* notion of safety often found in the control theory literature, whereby the safety constraints are defined as state-space constraints. Additional safety considerations were posed by the application-specific requirements to maintain a compromise between two competing objectives (Chapter 2), as well as minimising the degradation of performance when enforcing safety (Chapter 4). In Chapter 5 we consider an alternative notion of safety whereby it is critical to avoid sharing information considered private, while Chapter 6 considers robustifying towards uncertainty that could otherwise compromise safety. Each of these three distinct notions of safety will play a crucial role in the years to come, especially as learning-based control strategies become more prominent. Considering data privacy concerns and minimising communication overhead in distributed systems, understanding the effects of training data on learned models and controllers, and employing such controllers to enforce safety in conjunction with other control strategies will become increasingly more prevalent. To this end, future work will focus on further adopting these notions of safety within

learning-based frameworks. Beyond considering how we might ensure that a learned control strategy maintains safety, it is of particular interest to also ensure that safety-critical controllers can be learned safely in unknown environments, i.e. enforcing safety during training.

7.1 Safety in the sense of state constraint satisfaction

Chapters 2-4 address the *classical* notion of safety. In Chapter 2 we consider MOCPs that allow us to find a compromise between two competing objectives. Even though it might sometimes be undesirable to place hard constraints on certain performance variables, a sensible compromise between objectives is often nevertheless necessary for an overall safe operation. In the context of spacecraft-trajectory design, for instance, understanding the effects of lower fuel consumption on mission duration is vital to guarantee an overall safe operation, as long transfer times place higher demands on the lifespan requirements of the spacecraft.

A noticeable assumption made when synthesising safety-critical controllers by means of HJ reachability analysis is that the underlying quasi-variational inequality was solved to a sufficient level of accuracy. To this end, Chapter 3 addresses how to estimate the confidence in an underlying synthesised control law. The final chapter considering the *classical* notion of safety, Chapter 4, considers integrating safety constraints with minimal implications on performance.

We will further summarise some of the key contributions and discuss future research directions of each Chapter below.

7.1.1 Multi-objective Hamilton-Jacobi reachability analysis

In Chapter 2 we presented a novel method of using the value function of a quasi-variational inequality to compute the decision space of multi-objective optimisation problems. The feasibility and effectiveness of the proposed approach was demonstrated by applying it to the problem of low-thrust trajectory design. The approach is applicable to arbitrary multi-objective optimisation problems where

the optimization variables are required to lie within a reachable set. By exploiting application-specific properties, in particular, knowledge of the gravitational field, we were able to reduce the number of states required for controller synthesis and solve the trajectory design problem to a high degree of accuracy.

Once the thrust angles have been fixed, the Hamiltonian becomes affine with respect to the thrust magnitude. Future research will aim at exploiting this fact by integrating classification-based approaches [123]. Furthermore, utilising approximations of the reachable set as in [12], as well as decomposing the reachable sets as in [39], appears promising.

7.1.2 Verification of safety-critical control policies

In Chapter 3 we consider estimating the variation between the BRT obtained from the sub-zero level set of the value function inherent in HJ reachability, and the BRT computed by means of simulating the trajectories and applying the synthesised controllers. Using regression models we extrapolate the numerical error across the state-space. This work provides a solid proof of concept, however, is only of an empirical nature and lacks probabilistic guarantees. In light of this, there are several scopes of improvement.

In the current work, the training points for the regression analysis are chosen at random. It would be of interest to instead choose deterministic points, and, as is done in quasi-monte carlo methods [136], quantify the discrepancy. Since the numerical error is an integration error, it might be possible to further draw on results from quasi-monte carlo methods and potentially apply the Koksma-Hlawka inequality [136] to provide an upper bound on the numerical error.

Our approach also shares similarities with probabilistic numerics [73], whereby a numerical problem is treated as one of statistical inference. It would thus be of interest to further investigate if results from probabilistic numerics can be applied to HJ reachability.

7.1.3 Safety-Aware Hybrid Control of Airborne Wind Energy Systems

In Chapter 4 we consider the integration of a safety-critical control policy within the complex and high-dimensional application of Airborne Wind Energy. In addition to synthesising the controller on the basis of an abstract model, we discussed a hybrid switching law so as to ensure safety with minimal implications on performance.

While we have demonstrated the efficacy of the proposed safety controller within a relatively high-fidelity model that includes actuator delays, such actuator delays could be formally included during controller synthesis. This would increase the number of states of the system and thus further research would be required to maintain computational feasibility.

An alternative low-dimensional model of the AWE system based on quaternions presented in [52] might be sufficient for safety-critical control of the kite/aircraft for safety conditions pertaining to states such as the airspeed or position in the NED frame. This would greatly reduce the required number of states needed for HJ reachability analysis, however, would necessitate a winch controller that ensures the tether force is kept to within a safe limit. Preliminary results of such a winch controller used in conjuncture with a control barrier function have appeared in [143].

7.2 Safety in the sense of information privacy

Particularly in multi-agent settings, it can often be necessary to limit the amount of data that is required to be shared among agents. Even in a cooperative setting, agents might not be able to share certain data as this can compromise data privacy. This notion of safety is considered in Chapter 5, where we proposed a distributed algorithm to solve dynamic programming problems whereby each agent has only partial knowledge of the state transition costs and probability. To minimise communication overhead and limit the amount of data agents are required to share with one another, state costs are aggregated and shared between agents only when the updated costs are expected to significantly influence the solution of other agents. We provide consensus proofs for the presented algorithm and show

that the error with respect to the *true solution* can be bounded. Finally, we apply the proposed distributed aggregation method to a large-scale urban traffic routing problem. Individual agents trying to reach a common access point, e.g. a motorway, share local congestion information with other agents allowing for fully decentralised routing with minimal communication between agents.

Future work aims at extending Theorem 5.2, which is based on the additional assumption of full network connectivity at all iterations, to the more general case of Assumption 5.1. It would also be of interest to further explore the choice of disaggregation and aggregation probabilities. In the computer aided verification literature, e.g. [1, 2], methods for choosing these probabilities have been presented, however, the context is slightly different and does not involve transition costs, as with the classical aggregation framework of [19].

7.3 Safety in the sense of robustness against uncertainty

When limited data is available, it can become critical to understand how this compromises safety. To this end, we take a stochastic approach to safety in Chapter 6 and consider a binary classification problem whereby the classifier (a 0-1 labelling mechanism) changes in a structured manner after each drawn sample. We demonstrate the problem remains probably approximately correct (PAC) learnable and provide the minimal number of samples necessary to accurately predict the next encountered label. Furthermore, we introduce a constructive method of generating the hypothesis using a Mixed Integer Linear Program (MILP). By considering the binary classifiers to lie in the class of hyper-rectangles, we enable the elimination of a significant majority of samples (95% in the provided example) before constructing the MILP.

Future research will further investigate relaxing the assumption of a hypothesis being in minimal disagreement, as this would allow us to use suboptimal solutions obtained by the MILP while maintaining probabilistic guarantees on misclassifications. Possibly, this could entail including knowledge of the duality gap into Equation (6.18)

and then considering the solution of the dual of the MILP as the obtained hypothesis. This is ongoing work and it is as yet unclear if this would be possible.

References

- [1] Alessandro Abate et al. “Adaptive Aggregation of Markov Chains: Quantitative Analysis of Chemical Reaction Networks”. In: *Computer Aided Verification*. Springer International Publishing, 2015, pp. 195–213. DOI: 10.1007/978-3-319-21690-4_12.
- [2] Alessandro Abate et al. “Adaptive formal approximations of Markov chains”. In: *Performance Evaluation* 148 (July 2021), p. 102207. DOI: 10.1016/j.peva.2021.102207.
- [3] Matthew J. Abrahamson et al. “Dawn orbit determination team: Trajectory modeling and reconstruction processes at Vesta”. In: *Advances in the Astronautical Sciences* 148. February (2013), pp. 2271–2290.
- [4] Teodoro Alamo, Roberto Tempo, and Eduardo F. Camacho. “Randomized Strategies for Probabilistic Solutions of Uncertain Feasibility and Optimization Problems”. In: *IEEE Transactions on Automatic Control* 54.11 (2009), pp. 2545–2559. DOI: 10.1109/TAC.2009.2031207.
- [5] Albert Altarovici, Olivier Bokanowski, and Hasnaa Zidani. “A general Hamilton-Jacobi framework for non-linear state-constrained control problems”. In: *ESAIM - Control, Optimisation and Calculus of Variations* 19.2 (2013), pp. 337–357. DOI: 10.1051/cocv/2012011.
- [6] Aaron D. Ames et al. “Control Barrier Functions: Theory and Applications”. In: *2019 18th European Control Conference (ECC)*. IEEE, June 2019. DOI: 10.23919/ecc.2019.8796030.
- [7] P. G. Antreasian et al. “Galileo orbit determination for the Ida encounter”. In: *Advances in the Astronautical Sciences* 87.2 (1994), pp. 1027–1048.
- [8] Mohamed Assellaou et al. “A Hamilton-Jacobi-Bellman approach for the optimal control of an abort landing problem”. In: *2016 IEEE 55th Conference on Decision and Control, CDC 2016*. 2016, pp. 3630–3635. DOI: 10.1109/CDC.2016.7798815.
- [9] Jean Pierre Aubin et al. “Impulse differential inclusions: A viability approach to hybrid systems”. In: *IEEE Transactions on Automatic Control* 47.1 (2002), pp. 2–20. DOI: 10.1109/9.981719.
- [10] Jean-Pierre Aubin, Alexandre M. Bayen, and Patrick Saint-Pierre. *Viability Theory*. Springer Berlin Heidelberg, 2011. DOI: 10.1007/978-3-642-16684-6.
- [11] J.H. Baayen and W.J. Ockels. “Tracking control with adaption of kites”. In: *IET Control Theory & Applications* 6.2 (2012), pp. 182–191. DOI: 10.1049/iet-cta.2011.0037.

- [12] Somil Bansal and Claire J. Tomlin. “DeepReach: A Deep Learning Approach to High-Dimensional Reachability”. In: *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, May 2021, pp. 1817–1824. DOI: 10.1109/icra48506.2021.9561949.
- [13] Somil Bansal et al. “Hamilton-jacobi reachability: A brief overview and recent advances”. In: *2017 IEEE 56th Annual Conference on Decision and Control (CDC)*. 2017, pp. 2242–2253. DOI: 10.1109/CDC.2017.8263977.
- [14] Bardi, Martino and Capuzzo-Dolcetta, Italo. *Optimal Control and Viscosity Solutions of Hamilton-Jacobi-Bellman Equations*. Birkhäuser Basel, 1997. DOI: 10.1007/978-0-8176-4755-1.
- [15] James C. Bean, John R. Birge, and Robert L. Smith. “Aggregation in Dynamic Programming”. In: *Operations Research* 35.2 (1987), pp. 215–220. DOI: 10.1287/opre.35.2.215.
- [16] Alberto Bemporad and Manfred Morari. “Control of systems integrating logic, dynamics, and constraints”. In: *Automatica* 35.3 (1999), pp. 407–427. DOI: 10.1016/S0005-1098(98)00178-2.
- [17] Felix Berkenkamp et al. “Safe model-based reinforcement learning with stability guarantees”. In: *Advances in Neural Information Processing Systems* 2017-Decem.Nips (2017), pp. 909–919.
- [18] Andrea Berra and Lorenzo Fagiano. “An optimal reeling control strategy for pumping airborne wind energy systems without wind speed feedback”. In: *2021 European Control Conference (ECC)*. 2021, pp. 1199–1204. DOI: 10.23919/ECC54610.2021.9655018.
- [19] D. Bertsekas and J.N. Tsitsiklis. *Neuro-Dynamic Programming*. Athena Scientific, 1996.
- [20] Dimitri Bertsekas. “Multiagent value iteration algorithms in dynamic programming and reinforcement learning”. In: *Results in Control and Optimization* 1 (2020), p. 100003. DOI: 10.1016/j.rico.2020.100003.
- [21] Dimitri P. Bertsekas. “Feature-based aggregation and deep reinforcement learning: A survey and some new implementations”. In: *IEEE/CAA Journal of Automatica Sinica* 6 (1 2019), pp. 1–31. DOI: 10.1109/JAS.2018.7511249.
- [22] O. Bokanowski et al. “Global Optimization Approach for the Ascent Problem of Multi-stage Launchers”. In: *Modeling, Simulation and Optimization of Complex Processes HPSC 2018*. Ed. by Hans Georg Bock et al. Cham: Springer International Publishing, 2021, pp. 1–42.
- [23] Olivier Bokanowski, Nicolas Forcadel, and Hasnaa Zidani. “Reachability and minimal times for state constrained nonlinear problems without any controllability assumption”. In: *SIAM Journal on Control and Optimization* 48.7 (2010), pp. 4292–4316. DOI: 10.1137/090762075.
- [24] Olivier Bokanowski et al. “Global optimization approach for the climbing problem of multi-stage launchers ”. working paper or preprint. Feb. 2015. URL: <https://hal-ensta-paris.archives-ouvertes.fr/hal-01113819>.

- [25] Olivier Bokanowski et al. “Payload optimization for multi-stage launchers using HJB approach and application to a SSO mission”. In: *IFAC-PapersOnLine* 50.1 (2017), pp. 2904–2910. DOI: 10.1016/j.ifacol.2017.08.647.
- [26] Olivier Bokanowski et al. *ROC-HJ solver*. URL: <https://uma.ensta-paris.fr/soft/ROC-HJ/>.
- [27] G.C. Calafiore and M.C. Campi. “The Scenario Approach to Robust Control Design”. In: *IEEE Transactions on Automatic Control* 51.5 (May 2006), pp. 742–753. DOI: 10.1109/tac.2006.875041.
- [28] Giuseppe Carlo Calafiore. “Random Convex Programs”. In: *SIAM Journal on Optimization* 20.6 (2010), pp. 3427–3464. DOI: 10.1137/090773490.
- [29] M. C. Campi and S. Garatti. “The Exact Feasibility of Randomized Solutions of Uncertain Convex Programs”. In: *SIAM Journal on Optimization* 19.3 (2008), pp. 1211–1230. DOI: 10.1137/07069821X.
- [30] M. C. Campi and S. Garatti. “A Sampling-and-Discarding Approach to Chance-Constrained Optimization: Feasibility and Optimality”. In: *Journal of Optimization Theory and Applications* 148.2 (Oct. 2010), pp. 257–280. DOI: 10.1007/s10957-010-9754-6.
- [31] M. C. Campi and S. Garatti. “Wait-and-judge scenario optimization”. In: *Mathematical Programming* 167.1 (July 2016), pp. 155–189. DOI: 10.1007/s10107-016-1056-9.
- [32] Marco C Campi and Simone Garatti. *Introduction to the Scenario Approach*. Philadelphia, PA: Society for Industrial and Applied Mathematics, Nov. 2018, p. 122. DOI: 10.1137/1.9781611975444.
- [33] Marco C. Campi and Simone Garatti. *Compression, Generalization and Learning*. 2023.
- [34] Marco C. Campi, Simone Garatti, and Maria Prandini. “The scenario approach for systems and control design”. In: *Annual Reviews in Control* 33.2 (2009), pp. 149–157. DOI: 10.1016/j.arcontrol.2009.07.001.
- [35] Marco Claudio Campi, Simone Garatti, and Federico Alessandro Ramponi. “A General Scenario Theory for Nonconvex Optimization and Decision Making”. In: *IEEE Transactions on Automatic Control* 63.12 (2018), pp. 4067–4078. DOI: 10.1109/TAC.2018.2808446.
- [36] Massimo Canale, Lorenzo Fagiano, and Mario Milanese. “High Altitude Wind Energy Generation Using Controlled Power Kites”. In: *IEEE Transactions on Control Systems Technology* 18.2 (2010), pp. 279–293. DOI: 10.1109/TCST.2009.2017933.
- [37] Françoise Chatelin and Willard L. Miranker. “Acceleration by aggregation of successive approximation methods”. In: *Linear Algebra and its Applications* 43 (Mar. 1982), pp. 17–47. DOI: 10.1016/0024-3795(82)90242-7.
- [38] Mo Chen, Jaime Fisac, and Claire Tomlin. *Berkeley Efficient API in C++ for Level Set methods*. URL: <https://github.com/HJReachability/beacsls>.
- [39] Mo Chen et al. “Decomposition of Reachable Sets and Tubes for a Class of Nonlinear Systems”. In: *IEEE Transactions on Automatic Control* 63.11 (2018). DOI: 10.1109/TAC.2018.2797194.

- [40] Mo Chen et al. *Optimal control helper toolbox*. URL: <https://github.com/HJReachability/helperOC>.
- [41] Antonello Cherubini et al. “Airborne Wind Energy Systems: A review of the technologies”. In: *Renewable and Sustainable Energy Reviews* 51 (2015), pp. 1461–1476. DOI: 10.1016/j.rser.2015.07.053.
- [42] G. S. Christensen, M. E. El-Hawary, and S. A. Soliman. “Dynamic Optimal Load Flow”. In: *Optimal Control Applications in Electric Power Systems*. Springer US, 1987, pp. 21–53. DOI: 10.1007/978-1-4899-2085-0_3.
- [43] European Commission, Directorate-General for Research, and Innovation. *Study on challenges in the commercialisation of airborne wind energy systems*. Publications Office, 2018. DOI: 10.2777/87591.
- [44] S. Costello et al. “Analysis of the maximum efficiency of kite-power systems”. In: *Journal of Renewable and Sustainable Energy* 7.5 (2015), pp. 1–16. DOI: 10.1063/1.4931111.
- [45] Sarah Dean et al. “On the Sample Complexity of the Linear Quadratic Regulator”. In: *Foundations of Computational Mathematics* 20.4 (Aug. 2019), pp. 633–679. DOI: 10.1007/s10208-019-09426-y.
- [46] Anna Désilles and Hasnaa Zidani. “Pareto front characterization for multiobjective optimal control problems using Hamilton-Jacobi approach”. In: *SIAM Journal on Control and Optimization* 57.6 (2019), pp. 3884–3910. DOI: 10.1137/18M1176993.
- [47] Moritz Diehl. “Airborne wind energy: Basic concepts and physical foundations”. In: *Airborne Wind Energy*. Ed. by Uwe Ahrens, Moritz Diehl, and Roland Schmehl. Green Energy and Technology. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 3–22. DOI: 10.1007/978-3-642-39965-7.
- [48] Thinh Doan, Siva Maguluri, and Justin Romberg. “Finite-Time Analysis of Distributed TD(0) with Linear Function Approximation on Multi-Agent Reinforcement Learning”. In: *Proceedings of the 36th International Conference on Machine Learning*. Vol. 97. PMLR, June 2019, pp. 1626–1635.
- [49] Dylan Eijkelhof et al. “Reference Design and Simulation Framework of a Multi-Megawatt Airborne Wind Energy System”. In: *Journal of Physics: Conference Series* 1618.3 (2020). DOI: 10.1088/1742-6596/1618/3/032020.
- [50] Jacob Englander, Donald Ellison, and B.A. Conway. “Global optimization of low-thrust, multiple-flyby trajectories at medium and medium-high fidelity”. In: vol. 152. Jan. 2014, pp. 1539–1558.
- [51] Paul J. Enright and Brace A. Conway. “Optimal finite-thrust spacecraft trajectories using collocation and nonlinear programming”. In: *Journal of Guidance, Control, and Dynamics* 14.5 (Sept. 1991), pp. 981–985. DOI: 10.2514/3.20739.
- [52] Michael Erhard, Greg Horn, and Moritz Diehl. “A quaternion-based model for optimal control of an airborne wind energy system”. In: *ZAMM - Journal of Applied Mathematics and Mechanics / Zeitschrift für Angewandte Mathematik und Mechanik* 97.1 (Sept. 2017), pp. 7–24. DOI: 10.1002/zamm.201500180.

- [53] Michael Erhard and Hans Strauch. “Control of towing kites for seagoing vessels”. In: *IEEE Transactions on Control Systems Technology* 21 (5 Sept. 2013), pp. 1629–1640. DOI: 10.1109/TCST.2012.2221093.
- [54] Michael Erhard and Hans Strauch. “Theory and Experimental Validation of a Simple Comprehensible Model of Tethered Kite Dynamics Used for Controller Design”. In: *Airborne Wind Energy*. Springer Berlin Heidelberg, 2013, pp. 141–165. DOI: 10.1007/978-3-642-39965-7_8.
- [55] Michael Erhard and Hans Strauch. “Flight control of tethered kites in autonomous pumping cycles for airborne wind energy”. In: *Control Engineering Practice* 40 (2015), pp. 13–26. DOI: 10.1016/j.conengprac.2015.03.001.
- [56] L.C. Evans and Panagiotis E. Souganidis. “Differential games and representation formulas for solutions of Hamilton-Jacobi-Isaacs equations”. In: *Indiana Univ. Math. J* 33.5 (1984), pp. 773–797. DOI: 10.1512/iumj.1984.33.33040.
- [57] Lorenzo Fagiano, Mario Milanese, and Dario Piga. “Optimization of airborne wind energy generators”. In: *International Journal of Robust and Nonlinear Control* 22.18 (2012), pp. 2055–2083. DOI: 10.1002/rnc.1808.
- [58] Lorenzo Fagiano and Carlo Novara. “Automatic crosswind flight of tethered wings for airborne wind energy: a direct data-driven approach”. In: *IFAC Proceedings Volumes* 47.3 (2014). 19th IFAC World Congress, pp. 4927–4932. DOI: 10.3182/20140824-6-ZA-1003.01896.
- [59] Lorenzo Fagiano et al. “Automatic crosswind flight of tethered wings for airborne wind energy: Modeling, control design, and experimental results”. In: *IEEE Transactions on Control Systems Technology* 22.4 (2014), pp. 1433–1447. DOI: 10.1109/TCST.2013.2279592.
- [60] Lorenzo Fagiano et al. “Autonomous Airborne Wind Energy Systems: Accomplishments and Challenges”. In: *Annual Review of Control, Robotics, and Autonomous Systems* 5.1 (2022), pp. 603–631. DOI: 10.1146/annurev-control-042820-124658.
- [61] Uwe Fechner et al. “Dynamic model of a pumping kite power system”. In: *Renewable Energy* 83 (2015), pp. 705–716. DOI: 10.1016/j.renene.2015.04.028.
- [62] Filiberto Fele and Kostas Margellos. “Probably Approximately Correct Nash Equilibrium Learning”. In: *IEEE Transactions on Automatic Control* 66.9 (2021), pp. 4238–4245. DOI: 10.1109/TAC.2020.3030754.
- [63] I.J. Fialho and T.T. Georgiou. “Worst case analysis of nonlinear systems”. In: *IEEE Transactions on Automatic Control* 44.6 (1999), pp. 1180–1196. DOI: 10.1109/9.769372.
- [64] Jaime F. Fisac et al. “Reach-Avoid Problems with Time-Varying Dynamics, Targets and Constraints”. In: *Proceedings of the 18th International Conference on Hybrid Systems: Computation and Control*. HSCC ’15. Seattle, Washington: Association for Computing Machinery, 2015, pp. 11–20. DOI: 10.1145/2728606.2728612.
- [65] Jaime F. Fisac et al. “A General Safety Framework for Learning-Based Control in Uncertain Robotic Systems”. In: *IEEE Transactions on Automatic Control* 64.7 (2019), pp. 2737–2752. DOI: 10.1109/TAC.2018.2876389.

- [66] Adrian Gambier and Essameddin Badreddin. “Multi-objective Optimal Control: An Overview”. In: *2007 IEEE International Conference on Control Applications*. IEEE, Oct. 2007. DOI: 10.1109/cca.2007.4389225.
- [67] S. Garatti and M. C. Campi. “Risk and complexity in scenario optimization”. In: *Mathematical Programming* 191.1 (Jan. 2022), pp. 243–279. DOI: 10.1007/s10107-019-01446-4.
- [68] Konstantinos Gatsis. “Federated Reinforcement Learning at the Edge: Exploring the Learning-Communication Tradeoff”. In: *2022 European Control Conference (ECC)*. 2022, pp. 1890–1895. DOI: 10.23919/ECC55457.2022.9837987.
- [69] Jeremy H. Gillula and Claire J. Tomlin. “Guaranteed Safe Online Learning via Reachability: tracking a ground target using a quadrotor”. In: *2012 IEEE International Conference on Robotics and Automation*. 2012, pp. 2723–2730. DOI: 10.1109/ICRA.2012.6225136.
- [70] Donald T Greenwood. *Principles of dynamics*. eng. 2nd ed. Englewood Cliffs, N.J: Prentice-Hall, 1988.
- [71] Xingang Guo and Bin Hu. “Exact Formulas for Finite-Time Estimation Errors of Decentralized Temporal Difference Learning with Linear Function Approximation”. In: *ArXiv* (2022).
- [72] David P. Helmbold and Philip M. Long. “Tracking drifting concepts by minimizing disagreements”. In: *Machine Learning* 14.1 (Jan. 1994), pp. 27–45. DOI: 10.1007/BF00993161.
- [73] Philipp Hennig, Michael A. Osborne, and Hans P. Kersting. *Probabilistic Numerics: Computation as Machine Learning*. Cambridge University Press, 2022. DOI: 10.1017/9781316681411.
- [74] Sylvia Herbert et al. “Scalable Learning of Safety Guarantees for Autonomous Systems using Hamilton-Jacobi Reachability”. In: *2021 IEEE International Conference on Robotics and Automation (ICRA)*. 2021, pp. 5914–5920. DOI: 10.1109/ICRA48506.2021.9561561.
- [75] C. Hermosilla, P. R. Wolenski, and H. Zidani. “The Mayer and Minimum Time Problems with Stratified State Constraints”. In: *Set-Valued and Variational Analysis* 26.3 (2018), pp. 643–662. DOI: 10.1007/s11228-017-0413-z.
- [76] João P. Hespanha. *Hybrid Control and Switched Systems, Lecture Notes*. 2005. URL: <https://web.ece.ucsb.edu/~hespanha/ece229/>.
- [77] Wassily Hoeffding. “Probability Inequalities for Sums of Bounded Random Variables”. In: *Journal of the American Statistical Association* 58.301 (1963), pp. 13–30. DOI: 10.1080/01621459.1963.10500830.
- [78] R. Scott Hudson and Steven J. Ostro. “Shape of asteroid 4769 Castalia (1989 PB) from inversion of radar images”. In: *Science* (1994). DOI: 10.1126/science.263.5149.940.
- [79] IEA. *Wind Power*. Nov. 2021. URL: <https://www.iea.org/reports/wind-power>.
- [80] Claudius Jehle and Roland Schmehl. “Applied tracking control for kite power systems”. In: *Journal of Guidance, Control, and Dynamics* 37.4 (2014), pp. 1211–1222. DOI: 10.2514/1.62380.

- [81] Yu Jiang and Hexi Baoyin. “Orbital Mechanics near a Rotating Asteroid”. In: *Journal of Astrophysics and Astronomy* (2014). DOI: 10.1007/s12036-014-9259-z.
- [82] Maryam Kamgarpour et al. “Hybrid Optimal Control for Aircraft Trajectory Design with a Variable Sequence of Modes”. In: *IFAC Proceedings Volumes* 44.1 (Jan. 2011), pp. 7238–7243. DOI: 10.3182/20110828-6-it-1002.01643.
- [83] Hassan K. Khalil. *Nonlinear Control*. Pearson Education, 2014, p. 400.
- [84] Basil Kouvaritakis and Mark Cannon. *Model Predictive Control: Classical, Robust and Stochastic*. eng. Advanced Textbooks in Control and Signal Processing. Cham: Springer Nature, 2015.
- [85] Shankar Kulumani and Taeyoung Lee. “Low-thrust trajectory design using reachability sets near asteroid 4769 Castalia”. In: *AIAA/AAS Astrodynamics Specialist Conference, 2016* (Sept. 2016), pp. 1–12. DOI: 10.2514/6.2016-5376.
- [86] Daniel Liberzon. *Switching in Systems and Control*. Birkhäuser Boston, 2003. DOI: 10.1007/978-1-4612-0017-8.
- [87] Daniel Liberzon. *Calculus of variations and optimal control theory: A concise introduction*. Princeton University Press, 2011. DOI: 10.1515/9781400842643.
- [88] G. Licitra et al. “Performance assessment of a rigid wing Airborne Wind Energy pumping system”. In: *Energy* 173 (2019), pp. 569–585. DOI: 10.1016/j.energy.2019.02.064.
- [89] Miles L. Loyd. “Crosswind Kite Power.” In: *Journal of energy* 4.3 (1980), pp. 106–111. DOI: 10.2514/3.48021.
- [90] Peng Lu et al. “Aircraft fault-tolerant trajectory control using Incremental Nonlinear Dynamic Inversion”. In: *Control Engineering Practice* 57 (2016), pp. 126–141. DOI: 10.1016/j.conengprac.2016.09.010.
- [91] Yimeng Lu and Maryam Kamgarpour. “Safe Mission Planning under Dynamical Uncertainties”. In: *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, May 2020. DOI: 10.1109/icra40945.2020.9196515.
- [92] Rolf H. Luchsinger. “Pumping Cycle Kite Power”. In: *Airborne Wind Energy*. Springer Berlin Heidelberg, 2013, pp. 47–64. DOI: 10.1007/978-3-642-39965-7_3.
- [93] Sergio Lucia and Sebastian Engell. “Control of towing kites under uncertainty using robust economic nonlinear model predictive control”. In: *2014 European Control Conference (ECC)*. 2014, pp. 1158–1163. DOI: 10.1109/ECC.2014.6862335.
- [94] John Lygeros, Claire Tomlin, and Shankar Sastry. “Controllers for reachability specifications for hybrid systems”. In: *Automatica* 35.3 (1999), pp. 349–370. DOI: 10.1016/S0005-1098(98)00193-9.
- [95] Danylo Malyuta et al. “Convex Optimization for Trajectory Generation: A Tutorial on Generating Dynamically Feasible Trajectories Reliably and Efficiently”. In: *IEEE Control Systems* 42.5 (Oct. 2022), pp. 40–113. DOI: 10.1109/mcs.2022.3187542.

- [96] E.C. Malz et al. “A reference model for airborne wind energy systems for optimization and control”. In: *Renewable Energy* 140 (2019), pp. 1004–1011. DOI: 10.1016/j.renene.2019.03.111.
- [97] Kostas Margellos and John Lygeros. “Hamilton-jacobi formulation for reach-avoid differential games”. In: *IEEE Transactions on Automatic Control* 56.8 (Aug. 2011), pp. 1849–1861. DOI: 10.1109/TAC.2011.2105730.
- [98] Kostas Margellos and John Lygeros. “Toward 4-D trajectory management in air traffic control: A study based on monte carlo simulation and reachability analysis”. In: *IEEE Transactions on Control Systems Technology* 21.5 (2013), pp. 1820–1833. DOI: 10.1109/TCST.2012.2220773.
- [99] Kostas Margellos, Maria Prandini, and John Lygeros. “On the Connection Between Compression Learning and Scenario Based Single-Stage and Cascading Optimization Problems”. In: *IEEE Transactions on Automatic Control* 60.10 (2015), pp. 2716–2721. DOI: 10.1109/TAC.2015.2394874.
- [100] R.B. Martin. “Optimal control drug scheduling of cancer chemotherapy”. In: *Automatica* 28.6 (Nov. 1992), pp. 1113–1123. DOI: 10.1016/0005-1098(92)90054-j.
- [101] *MATLAB Aerospace Toolbox*. The MathWorks, Natick, MA, USA. 2021.
- [102] Kaisa Miettinen. *Nonlinear Multiobjective Optimization*. Vol. 12. International Series in Operations Research & Management Science. Boston, MA: Springer US, 1998. DOI: 10.1007/978-1-4615-5563-6.
- [103] Ian M. Mitchell. “The flexible, extensible and efficient toolbox of level set methods”. In: *Journal of Scientific Computing* 35.2-3 (2008), pp. 300–329. DOI: 10.1007/s10915-007-9174-4.
- [104] Ian M. Mitchell, Alexandre M. Bayen, and Claire J. Tomlin. “A time-dependent Hamilton-Jacobi formulation of reachable sets for continuous dynamic games”. In: *IEEE Transactions on Automatic Control* 50.7 (2005), pp. 947–957. DOI: 10.1109/TAC.2005.851439.
- [105] Manfred Morari. “Hybrid system analysis and control via mixed integer optimization”. In: *IFAC Proceedings Volumes* 34.25 (2001). 6th IFAC Symposium on Dynamics and Control of Process Systems 2001, Jejudo Island, Korea, 4-6 June 2001, pp. 1–12. DOI: 10.1016/S1474-6670(17)33796-5.
- [106] M. Thamban Nair and Arindama Singh. *Linear algebra*. eng. Springer, Singapore, 2018. DOI: 10.1007/978-981-13-0926-7.
- [107] Angelia Nedic and Asuman Ozdaglar. “Distributed Subgradient Methods for Multi-Agent Optimization”. In: *IEEE Transactions on Automatic Control* 54.1 (2009), pp. 48–61. DOI: 10.1109/TAC.2008.2009515.
- [108] S. Ober-Blöbaum, M. Ringkamp, and G. zum Felde. “Solving multiobjective Optimal Control problems in space mission design using Discrete Mechanics and reference point techniques”. In: *2012 IEEE 51st IEEE Conference on Decision and Control (CDC)*. 2012, pp. 5711–5716. DOI: 10.1109/CDC.2012.6426285.
- [109] John Oprea. *Differential geometry and its applications*. eng. 2nd ed. Washington, D.C.: Mathematical Association of America, 2007. DOI: 10.1090/clrm/059.

- [110] Stanley Osher and Ronald Fedkiw. *Level Set Methods and Dynamic Implicit Surfaces*. Applied Mathematical Sciences. New York, NY: Springer New York, 2003. DOI: 10.1007/b98879.
- [111] Daniel W. Parcher. “Low-thrust orbit transfer design for Dawn operations at Vesta”. In: *Advances in the Astronautical Sciences* 140.818 (2011), pp. 1213–1232.
- [112] Nathalie Paul et al. “Multi-Agent Neural Rewriter for Vehicle Routing with Limited Disclosure of Costs”. In: *ArXiv* (2022).
- [113] Iman Rahimi et al. “A Review on Constraint Handling Techniques for Population-based Algorithms: from single-objective to multi-objective optimization”. In: *Archives of Computational Methods in Engineering* 30.3 (Dec. 2022), pp. 2181–2209. DOI: 10.1007/s11831-022-09859-9.
- [114] Krishan Rana et al. “Bayesian Controller Fusion: Leveraging Control Priors in Deep Reinforcement Learning for Robotics”. In: *CoRR* abs/2107.09822 (2021). URL: <https://arxiv.org/abs/2107.09822>.
- [115] Sebastian Rapp. “Robust Automatic Pumping Cycle Operation of Airborne Wind Energy Systems Rapp”. PhD thesis. Delft University of Technology, 2021. DOI: 10.4233/uuid:ab2adf33-ef5d-413c-b403-2cfb4f9b6bae.
- [116] Sebastian Rapp and Roland Schmehl. “Vertical takeoff and landing of flexible wing kite power systems”. In: *Journal of Guidance, Control, and Dynamics* 41.11 (2018), pp. 2386–2400. DOI: 10.2514/1.G003535.
- [117] Sebastian Rapp and Roland Schmehl. “Enhancing Resilience of Airborne Wind Energy Systems Through Upset Condition Avoidance”. In: *Journal of Guidance, Control, and Dynamics* 44.2 (2021), pp. 251–265. DOI: 10.2514/1.G005189.
- [118] Sebastian Rapp et al. “Cascaded pumping cycle control for rigid wing airborne wind energy systems”. In: *Journal of Guidance, Control, and Dynamics* 42.11 (2019), pp. 2456–2473. DOI: 10.2514/1.G004246.
- [119] Carl Edward Rasmussen. “Gaussian Processes in Machine Learning”. In: *Advanced Lectures on Machine Learning: ML Summer Schools 2003, Canberra, Australia, February 2 - 14, 2003, Tübingen, Germany, August 4 - 16, 2003, Revised Lectures*. Ed. by Olivier Bousquet, Ulrike von Luxburg, and Gunnar Rätsch. Berlin, Heidelberg: Springer Berlin Heidelberg, 2004, pp. 63–71. DOI: 10.1007/978-3-540-28650-9_4.
- [120] David F. Rogers et al. “Aggregation and Disaggregation Techniques and Methodology in Optimization”. In: *Operations Research* 39.4 (1991), pp. 553–582. DOI: 10.1287/opre.39.4.553.
- [121] Licio Romao, Kostas Margellos, and Antonis Papachristodoulou. “Probabilistic feasibility guarantees for convex scenario programs with an arbitrary number of discarded constraints”. In: *Automatica* 149 (2023), pp. 1–9. DOI: 10.1016/j.automatica.2022.110601.
- [122] Licio Romao, Antonis Papachristodoulou, and Kostas Margellos. “On the exact feasibility of convex scenario programs with discarded constraints”. In: *IEEE Transactions on Automatic Control* 68.4 (2023), pp. 1986–2001. DOI: 10.1109/TAC.2022.3165320.

- [123] Vicenc Rubies-Royo et al. “A Classification-based Approach for Approximate Reachability”. In: *Proceedings - IEEE International Conference on Robotics and Automation* 2019-May (2019), pp. 7697–7704. DOI: 10.1109/ICRA.2019.8793919.
- [124] Volkan Salma, Felix Friedl, and Roland Schmehl. “Improving reliability and safety of airborne wind energy systems”. In: *Wind Energy* 23.2 (2020), pp. 340–356. DOI: 10.1002/we.2433.
- [125] Shankar Sastry. *Nonlinear systems : analysis, stability, and control*. eng. Interdisciplinary applied mathematics ; v. 10. New York ; London: Springer, 1999.
- [126] D. J. Scheeres, J. K. Miller, and D. K. Yeomans. “The Orbital Dynamics Environment of 433 Eros: A Case Study for Future Asteroid Missions”. In: *BBA - Bioenergetics* (1987). DOI: 10.1016/0005-2728(87)90102-2.
- [127] D. J. Scheeres et al. “Orbits close to asteroid 4769 Castalia”. In: *Icarus* 121.1 (1996), pp. 67–87. DOI: 10.1006/icar.1996.0072.
- [128] Roland Schmehl. *Airborne Wind Energy: Advances in Technology Development and Research*. Springer, Singapore, 2018. DOI: 10.1007/978-981-10-1947-0.
- [129] Eduardo Schmidt et al. “In-Flight Estimation of the Aerodynamics of Tethered Wings for Airborne Wind Energy”. In: *IEEE Transactions on Control Systems Technology* 28.4 (2020), pp. 1309–1322. DOI: 10.1109/TCST.2019.2907663.
- [130] Bernhard Schölkopf, John Platt, and Thomas Hofmann. “An Application of Reinforcement Learning to Aerobatic Helicopter Flight”. In: *Advances in Neural Information Processing Systems 19: Proceedings of the 2006 Conference*. 2007, pp. 1–8.
- [131] W. Seefelder. *Lunar Transfer Orbits Utilizing Solar Perturbations and Ballistic Capture*. Luft- und Raumfahrt. Utz, Wiss., 2002. URL: https://books.google.co.uk/books?id=NVg%5C_vYHePt0C.
- [132] Lawrence F. Shampine and Mark W. Reichelt. “The MATLAB ODE Suite”. In: *SIAM Journal on Scientific Computing* 18.1 (Jan. 1997), pp. 1–22. DOI: 10.1137/S1064827594276424.
- [133] Abolfazl Shirazi, Josu Ceberio, and Jose A. Lozano. “Spacecraft trajectory optimization: A review of models, objectives, approaches and solutions”. In: *Progress in Aerospace Sciences* 102 (2018), pp. 76–98. DOI: 10.1016/j.paerosci.2018.07.007.
- [134] R.M. Shoucri. “Pump function of the heart as an optimal control problem”. In: *Journal of Biomedical Engineering* 13.5 (Sept. 1991), pp. 384–390. DOI: 10.1016/0141-5425(91)90019-4.
- [135] Manuel Soler et al. “A Hybrid Optimal Control Approach to Fuel-Efficient Aircraft Conflict Avoidance”. In: *IEEE Transactions on Intelligent Transportation Systems* 17.7 (July 2016), pp. 1826–1838. DOI: 10.1109/tits.2015.2510824.
- [136] R Tempo, Giuseppe Calafiore, and Fabrizio Dabbene. *Randomized algorithms for analysis and control of uncertain systems*. eng. Communications and control engineering series. London: Springer, 2005. DOI: 10.1007/b137802.
- [137] Davide Todeschini et al. “Control of a rigid wing pumping Airborne Wind Energy system in all operational phases”. In: *Control Engineering Practice* 111 (2021), p. 104794. DOI: 10.1016/j.conengprac.2021.104794.

- [138] John N. Tsitsiklis and Benjamin van Roy. “Feature-based methods for large scale dynamic programming”. In: *Machine Learning* 22.1 (Mar. 1996), pp. 59–94. DOI: 10.1007/BF00114724.
- [139] J. Tzanos, K. Margellos, and J. Lygeros. “Optimal Wind Turbine Placement via Randomized Optimization Techniques”. In: *17th Power Systems Computation Conference 2011 (PSCC)*. 2011, pp. 1116–1123.
- [140] *U.S. Military Handbook MIL-HDBK-1797B*. Apr. 2012.
- [141] P. P. Varaiya. “On the Existence of Solutions to a Differential Game”. In: *SIAM Journal on Control* 5.1 (1967), pp. 153–162. DOI: 10.1137/0305009.
- [142] Chris Vermillion et al. “Electricity in the air: Insights from two decades of advanced control research and experimental flight testing of airborne wind energy systems”. In: *Annual Reviews in Control* 52 (2021), pp. 330–357. DOI: 10.1016/j.arcontrol.2021.03.002.
- [143] Nikolaus Vertovec. *Enforcing Safety in Airborne Wind Energy Systems [Conference presentation]*. Wind Energy Science Conference 2023. Glasgow, UK, May 2023.
- [144] Nikolaus Vertovec and Kostas Margellos. “State Aggregation for Distributed Value Iteration in Dynamic Programming”. In: *IEEE Control Systems Letters* 7 (2023), pp. 2269–2274. DOI: 10.1109/lcsys.2023.3285655.
- [145] Nikolaus Vertovec, Sina Ober-Blöbaum, and Kostas Margellos. “Multi-objective minimum time optimal control for low-thrust trajectory design”. In: *2021 European Control Conference (ECC)*. 2021, pp. 1975–1980. DOI: 10.23919/ECC54610.2021.9654919.
- [146] Nikolaus Vertovec, Sina Ober-Blöbaum, and Kostas Margellos. “Verification of safety critical control policies using kernel methods”. In: *2022 European Control Conference (ECC)*. 2022, pp. 1870–1875. DOI: 10.23919/ECC55457.2022.9838224.
- [147] Nikolaus Vertovec, Sina Ober-Blöbaum, and Kostas Margellos. “Multi-objective low-thrust spacecraft trajectory design using reachability analysis”. In: *European Journal of Control* 69 (2023), p. 100758. DOI: 10.1016/j.ejcon.2022.100758.
- [148] Nikolaus Vertovec, Sina Ober-Blöbaum, and Kostas Margellos. “Safety-Aware Hybrid Control of Airborne Wind Energy Systems”. In: *Journal of Guidance, Control, and Dynamics* 47.2 (2024), pp. 326–338. DOI: 10.2514/1.G007254.
- [149] M. Vidyasagar. *Learning and Generalisation*. Springer London, 2003. DOI: 10.1007/978-1-4471-3748-1.
- [150] O. von Stryk and R. Bulirsch. “Direct and indirect methods for trajectory optimization”. In: *Annals of Operations Research* 37 (1 Dec. 1992), pp. 357–373. DOI: 10.1007/BF02071065.
- [151] Zhenbo Wang and Michael J. Grant. “Optimization of Minimum-Time Low-Thrust Transfers Using Convex Programming”. In: *Journal of Spacecraft and Rockets* 55.3 (2018), pp. 586–598. DOI: 10.2514/1.A33995.
- [152] Nakeya D. Williams et al. “An optimal control approach for blood pressure regulation during head-up tilt”. In: *Biological Cybernetics* 113.1-2 (Oct. 2018), pp. 149–159. DOI: 10.1007/s00422-018-0783-9.

- [153] Paul Williams, Bas Lansdorp, and Wubbo Ockesl. “Optimal Crosswind Towing and Power Generation with Tethered Kites”. In: *Journal of Guidance, Control, and Dynamics* 31.1 (2008), pp. 81–93. DOI: 10.2514/1.30089.
- [154] Tony A. Wood and Maryam Kamgarpour. “Automaton-based stochastic control for navigation of emergency rescuers in buildings”. In: *2016 IEEE Conference on Control Applications (CCA)*. IEEE, Sept. 2016. DOI: 10.1109/cca.2016.7587893.
- [155] Tony A. Wood, Sean Summers, and John Lygeros. “A stochastic reachability approach to emergency building evacuation”. In: *52nd IEEE Conference on Decision and Control*. IEEE, Dec. 2013. DOI: 10.1109/cdc.2013.6760791.
- [156] Yao-nan Yu, Khien Vongsuriya, and Leonard Wedman. “Application of an Optimal Control Theory to a Power System”. In: *IEEE Transactions on Power Apparatus and Systems* PAS-89.1 (Jan. 1970), pp. 55–62. DOI: 10.1109/tpas.1970.292668.
- [157] Aldo U. Zraggen, Lorenzo Fagiano, and Manfred Morari. “Automatic Retraction and Full-Cycle Operation for a Class of Airborne Wind Energy Generators”. In: *IEEE Transactions on Control Systems Technology* 24.2 (2016), pp. 594–608. DOI: 10.1109/TCST.2015.2452230.