UNIVERSITY OF CALIFORNIA SAN DIEGO

Uniting and Balancing Control Objectives: Safety, Stability, Smoothness, and Resource Conservation

A dissertation submitted in partial satisfaction of the
requirements for the degree
Doctor of Philosophy

in

Engineering Sciences (Aerospace Engineering)

by

Pio Ong

Committee in charge:

Professor Jorge Cortés, Chair
Professor Maurício de Oliveira
Professor Miroslav Krstić
Professor Melvin Leok
Professor Behrouz Touri

2022

The dissertation of Pio Ong is approved, and it is acceptable in quality and form for publication on microfilm and electronically.

University of California San Diego

2022

DEDICATION

To my family.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGEMENTS

I would like to express my utmost gratitude to my advisor and mentor, Prof. Jorge Cortés, who goes above and beyond to ensure my success as a Ph.D. student. This dissertation is a culmination of my five-year research at UCSD, and all works in it would not have been possible without the guidance from Jorge. He spends time with me every week to give valuable insights, not only on my research, but also on how I can improve as a researcher. His genuine care for his students has been evident from our interactions throughout the years. Outside of our interactions, he is always on the lookout for new opportunities for me to further my career. He speaks about me and my works to others, and introduces me to potential collaborators. Jorge is a role model of an advisor whom I strive to be like one day in my own career. I feel like I have won the lottery to get an advisor like Jorge, and I really cannot ask for better. I sincerely thank Jorge for everything he has done for me in the past years, for providing me with the best experience as a Ph.D. student, and for being an exceptional mentor. I hope that we will cross paths and find ways to work together again in the future.

I also would like to thank my dissertation committee professors, Prof. Maurício de Oliveira, Prof. Miroslav Krstić,, Prof. Melvin Leok, and Prof. Behrouz Touri, for taking time out of their schedules and providing me with suggestions and criticisms.

My decision to return to UCSD for its Ph.D. program is one of the greatest choice I made. As a UCSD alumnus who had a very satisfied experience with its undergraduate program, my expectation was high. Yet, the Ph.D. program has exceeded my expectation. This is largely thanks to the faculty of Dynamic Systems and Control group at UCSD for providing the best education I have ever received. I am very grateful for all the professors who taught me.

Throughout my Ph.D., I was also fortunate to collaborate with amazing people outside of UCSD on different occasions. I want to thank my collaborators and coauthors Andrew Taylor, Prof. Aaron Ames, Beatrice Capelli, and Prof. Lorenzo Sabattini. I have enjoyed our collaboration. I look forward to working with Prof. Aaron Ames again at my postdoctoral position at CalTech in the immediate future. As for the others, I hope I can find the opportunities to work with them again in the future as well.

I also want to extend my gratitude to the members of the joint research group under Prof. Jorge Cortés and Prof. Sonia Martínez. I have enjoyed spending time with them throughout the years. I especially thank Tor, Dan, Aamodh, Parth, and Ahmed, for their kind friendship and for helping me in many ways, not only in my research, but also in my personal life. And most importantly, I am very glad to have met Priyank Srivastava, my closest friend at UCSD. Priyank is a great friend who has always been there to provide me with encouragement and support throughout the years. The time during my Ph.D. would have been dull without all the discussions and time spent with Priyank. Seeing that both of us are working in the same field and that we still keep in touch after his graduation, I look forward to our inevitable collaboration and the chance to meet again in the future.

Another group of important people during these years are my best friends, including Hongyu Jin, Natoyan Pyo, Adrian Teng-Amnuay, and Mason Dionne. These people often reach out to me to show support and to check if I am doing okay. I find comfort to have genuine friends like them who truly care for my success. Special thanks to Adrian who always welcome me to his home and spend time with me whenever I need to take days off from work.

Finally, I am grateful for my family for shaping me to be who I am today and for supporting my decision to pursue my career in academia. For their unconditional love, I thank my parents,

brothers, sister-in-laws, aunt-in-law, and cousins. Knowing that my loving family will always be there for me allows me to push forward without any doubts in my career decision. Special mentions to my toddler niece and nephew whom I also love wholeheartedly.

The research reported in this dissertation was supported in part by the National Science Foundation Awards CNS-1329619, CNS-1446891, and ECCS-1917177.

Chapter 3, in part, has been submitted for publication of the material [OCSC21] as it may appear as 'Nonsmooth Control Barrier Function Design of Continuous Constraints for Network Connectivity Maintenance' by P. Ong, B. Capelli, L. Sabattini, and J. Cortés, in The International Journal of Robotics Research. The dissertation author was the primary investigator and author of this paper.

Chapter 4, in part, is a reprint of the material [OC19] as it appears in 'Universal formula for smooth safe stabilization' by P. Ong and J. Cortés, in IEEE Conference on Decision and Control, 2019. The dissertation author was the primary investigator and author of this paper.

Chapter 5, in part, has been submitted for publication of the material [OC21b] as it may appear as 'Performance-barrier-based event-triggered control with applications to network systems' by P. Ong and J. Cortés, in IEEE Transactions on Automatic Control, 2021. The dissertation author was the primary investigator and author of this paper.

Chapter 6, in full, is a reprint of the material [TOCA21] as it appears in 'Safety-critical event triggered control via input-to-state safe barrier functions' by A. J. Taylor, P. Ong, J. Cortés, and A. D. Ames, in IEEE Control Systems Letters, 2021. The dissertation author was one of the primary investigators and authors of this paper.

Chapter 7, in part, is a reprint of the material [OC21a] as it appears in 'Opportunistic robot control for interactive multiobjective optimization under human performance limitations' by P. Ong

and J. Cortés, in Automatica, 2021. The dissertation author was the primary investigator and author of this paper.

VITA

2012	Bachelor of Science in Aerospace Engineering, University of California, San Diego
2013	Master of Science in Astronautical Engineering, University of Southern California
2022	Doctor of Philosophy in Engineering Sciences (Aerospace Engineering), University of California San Diego

PUBLICATIONS

Journal publications:

[1] P. Ong, B. Capelli, L. Sabattini, and J. Cortés. Nonsmooth control barrier function design of continuous constraints for network connectivity maintenance. International Journal of Robotics Research, 2021. Submitted

[2] P. Ong and J. Cortés. Performance-barrier-based event-triggered control with applications to network systems. *IEEE Transactions on Automatic Control*, 2021. Submitted.

[3] A. J. Taylor, P. Ong, J. Cortés, and A. D. Ames. Safety-critical event triggered control via input-to-state safe barrier functions. *IEEE Control Systems Letters*, vol. 5, no. 3, pp. 749–754, 2021.

[4] P. Ong and J. Cortés, Opportunistic robot control for interactive multiobjective optimization under human performance limitations. *Automatica*, vol. 123, pp. 109263, 2021.

Conference proceedings:

[5] P. Ong, B. Capelli, L. Sabattini, and J. Cortés. Network connectivity maintenance via non-smooth control barrier functions. In *IEEE Conf. on Decision and Control*, pp. 4780–4785, Austin, Texas, December 2021.

[6] A. J. Taylor, P. Ong, J. Cortés, and A. D. Ames. Safety-critical event triggered control via input-to-state safe barrier functions. In *IEEE Conf. on Decision and Control*, Jeju Island, South Korea, December 2020.

[7] P. Ong and J. Cortés. Universal formula for smooth safe stabilization. In *IEEE Conf. on Decision and Control*, pp. 2373–2378, Nice, France, December 2019.

[8] P. Ong and J. Cortés. Event-triggered control design with performance barrier. In *IEEE Conf. on Decision and Control*, pp. 951–956, Miami Beach, Florida, December 2018.

[9] P. Ong and J. Cortés. Event-triggered interactive gradient descent for real-time multi-objective optimization. In *IEEE Conf. on Decision and Control*, pp. 5445–5450, Melbourne, Australia, December 2017.

ABSTRACT OF THE DISSERTATION

Uniting and Balancing Control Objectives:
Safety, Stability, Smoothness, and Resource Conservation

by

Pio Ong

Doctor of Philosophy in Engineering Sciences (Aerospace Engineering)

University of California San Diego, 2022

Professor Jorge Cortés, Chair

Multi-robot systems can accomplish a variety of tasks through the power of coordination. There are mutliple benefits. These systems have many advantages over a single very complex robot in term of scalability, versatility, and adaptability. In many cases, the robots cannot accomplish much by itself, but coordination empowers them the ability to complete various objectives. Even when the individuals robots are very capable, coordination can increase robot efficiency by allocating robots with fitting tasks. In both scenarios, the problem of balancing the system objectives arise naturally, and properly addressing it can lead to better overall performance. Motivated by this

observation, this dissertation seek to understand how different objectives can be put together and how to strike a balance between them. We consider control objectives at the most fundamental level to control systems, such as stability, system safety, smoothness of the controller, performance, and resources spent for accomplishing tasks.

This dissertation is divided into two parts. The first part deals with control laws that consider both stability and safety objectives. We design controllers that can satisfy simultaneously conditions given by control Lyapunov functions and control barrier functions. Depending on the smoothness properties of the given functions, we guarantee the continuity or smoothness of the controller. In particular, we design a continuous controller for connectivity maintenance, and also design a universal formula for smooth safe stabilization. In the second part, we study the resource-efficient implementation of control laws using event-triggered control. We improve the existing event-triggered control framework for stabilization by incorporating prescribed performance into the design. The resulting framework further enhances the advantage of resource conservation characteristic of event-triggered control. We build on the proposed framework to design an intrinsically Zeno-free distributed triggering mechanisms for network systems. In addition, this dissertation also explores unconventional ways to utilize the event-triggered control framework. In one way, we deviate ourselves from trigger conditions that use Lyapunov functions replacing it instead with barrier certificate and develop an event-triggered control framework for safety objectives. Another interesting way we explore to use event-triggered control is in the context of human supervised multiobjective optimization. In this setting, we consider the human as a valuable resource, which should be used sparingly, and use event-triggered control to accommodate various models of human performance, such as constraints on the response time and the interaction frequency.

Chapter 1

Introduction

In the not so distant future, it is envisioned that robots will cooperate among themselves, and with humans, in performing a multitude of tasks in everyday life, ranging from routine jobs to dangerous missions. Thanks to the a rapid advances of computing, communication, and sensing capabilities, robots have become more capable and more sophisticated. More importantly, they can now be produced cheaply, making it more commonplace. In the past, most robots existed only in the industrial sector helping with manufacturing (e.g., welding, painting, handling), and they were few in numbers with minimal direct interactions with humans. Today, robots work alongside humans in large number, interacting with humans in many ways. One example is the warehouse robots that collaborate with humans in fulfillment centers. These robots move around in the same environment with humans and take on tasks that are too tedious or laborious for humans. They also interact with humans directly such as bringing items to humans for inspection and requesting human assistance when they encounter problems. The introduction of robots to these workplaces has greatly enhanced their production performance and efficiency. Following the trend of developments, it will not be long before robots are introduced into different settings and become an integral part of our lives.

We are interested in the possibilities for robots to interact among themselves and with humans. Robots are like smart devices in that they assist humans and facilitate our lives. We surround ourselves with smart devices, e.g., our phones, our assistant devices, and our house appliances, simply because they make our lives more comfortable. Undoubtedly, robots will follow suit and become ubiquitous for the same reasons. The difference between robots and smart devices are that robots will have some level of autonomy, possibly making decisions by themselves, and they may also have the power to communicate effectively. This naturally begs the question of how they will coordinate among themselves and how they should interact with humans. Today, we can observe that smart devices, even with limited communication capabilities, work together as a network in order to enhance their overall functionality and efficiency. Such concept is referred to as the *internet of things*, and we predict that future robots will also inherit the concept. Individual robots, although with their own autonomy and purposes, will be part of a bigger collaboration network that connects the robots together in order to augment their collective abilities and capabilities.

Multi-robot systems rely on this exact principle as they coordinate simple robots to accomplish a variety of tasks. In many scenarios, a group of less capable robots are more versatile and have better performance than a single specialized robot. For example, in a search-and-rescue operation, a group of robots can quickly cover a large search area and then coordinate among themselves to provide appropriate resources where they are needed. In addition, the robots can avoid searching the same area twice, and they can also reduce the total travel distance by having closest robots to explore a point of interest. The strength of a multi-robot system here lies not in its number but in the power of collaboration. With proper coordination, robots can reduce the total amount of workload. Indeed, there are also many scenarios where a specialized robot can outperform a multi-robot system. However, it is not about which one is better than which, but how we can understand and

utilize their respective strengths. It is not farfetched to think that future robots, regardless of how sophisticate they become, will work in collaboration. We imagine that even when robots are capable of accomplishing tasks on their own, they will still coordinate with others in order to make their tasks easier and obtain better performance.

There are still many technical challenges that we need to overcome before the aforementioned vision is realized. Robot coordination is an active field of research with many interesting ideas. Common research topics include algorithmic solutions that allow robots to operate with only partial network information, robustness of the coordination to failure of individual robots, and scalability of the collaboration network. Indeed, outside of these popular topics, there are many other aspects that we still need to explore and answer. This dissertation is motivated by the multiobjective nature of multi-robot systems. Since individual robots in the network may have their own objectives to accomplish, we face the question of how to prioritize between their collective objectives and how to come up with a solution that can satisfy all of the participating members. Addressing the question of how to handle multiple objectives effectively and simultaneously is the central theme of this dissertation.

Objectives in control systems can refer to goals other than stability of the desired equilibrium. Although we often seek to drive the systems to desired states, it is equally important to consider how it is done. One type of system objectives to consider is trajectory safety. We use the term safety objective to broadly refer to the goal of avoiding undesirable states, which may include not only states that signify total loss of robots but also states where system performance diminishes. An example of a safety objective is robot connectivity, where we want the robot network to avoid the states where they lose connectivity. Regardless of the severity, safety is an important control objective that can dictate how we stabilize an equilibrium of a system.

Another type of control objectives is resource conservation. The advent of increasingly capable devices operating in complex scenarios raises the question of using the available resources efficiently in order to meet task specifications and provide algorithmic solutions that can scale up. Typically spending the more resources leads to better system performance, but due to diminishing returns, choosing to withhold some resources might be a more sensible option. Balancing control objectives such as stability and safety with the efficient use of available resources is then an integral part of the implementation of control laws.

This dissertation is divided into two parts. The first part (Chapters 3–4) focuses on scenarios with both stability and safety objectives. We formulate both objectives as control constraints and use them to design feedback controllers. In our treatment, we pay special attention to the continuity/smoothness properties of the resulting feedback controllers. The second part (Chapters 5–7) investigates the implementation of the control laws with a focus on balancing resource usage and system performance, either in term of stability or safety.

1.1 Literature Review

1.1.1 Smooth Safe Stabilization

Broadly speaking, safety is the concept of avoiding undesirable states, and it is directly related to the concept of set invariance. There is a vast literature surrounding the concept of set invariance in the context of optimization. Barrier methods, more popularly known as interior-point methods [FGW02, BV04], were developed to ensure that solutions remain in a feasible set. Although there was already a sizeable literature of set invariance in control [Bla99], the introduction

of barrier functions to the field has greatly expanded its literature. Barrier Lyapunov Functions (BLFs) [NMJ05,TGT09] integrate the ideas of barrier methods into Lyapunov theory. BLFs enforce safety by specifying a Lyapunov function with each sublevel set contained in the safe set. Safety is then a byproduct of stabilization, as the decrease of the Lyapunov function (i.e., stabilization) leads to containment within sequentially nested sublevel sets. Nevertheless, the coupling between safety and stabilization, in principle, makes the analysis more difficult. Another approach is to isolate the safety analysis from stability. The notion of barrier certificate [PJ04] focuses solely on safety. The certificate is defined to be positive for states that are safe, and it guarantees set invariance through monotonicity of its time evolution. Control Barrier Functions [WA07] (CBF) build on the notion, and are used to find choices of control inputs that makes the certificate increase, guaranteeing forward invariance of a desired set. The CBF idea is refined further by using Nagumo theorem [BM07] as the basis for set invariance, see [ACE+19]. This refined version introduces the concept of letting the certificate also decrease, depending on the level of safety. CBFs are commonly used in tandem with control Lyapunov functions (CLFs) in order to design a controller for both safety and stability objectives.

There are two general approaches to exploiting the available CLFs and CBFs for safe stabilization. The first is optimization-based, where a control input is selected through a pointwise optimization over a feasible control set given by the conditions from CLFs and CBFs (see e.g., [MPA13, AXGT17, XTGA15]). The drawback of this approach, as pointed out in [MPA15] with a counterexample, is that the resulting feedback controller can be non-Lipschitz. To remedy this, [AXGT17, XTGA15] propose relaxing the condition for stability in order to guarantee Lipschitzness of the feedback controller. Another approach, as introduced by [RJ16], is to combine a CLF and a CBF into a CLBF and then use Sontag's universal formula for stabilization [Son89].

Although this method will guarantee smoothness of the controller, as long as CLFs and CBFs are sufficiently smooth, the constructed CLBF will only work when it meets the restrictive criteria laid out by [RJ16], which is discussed further in [BK17].

Nonsmooth Control Barrier Functions (NCBFs) [GCE17a] are a generalization of CBFs to consider certificate that are not differentiable. However, care must be taken when using NCBFs as there are many results that rely on the smoothness of the certificate function. Particularly, there is no longer a guarantee on the continuity when using optimization-based controllers. The paper [MPA15] uses perturbation theory to study the smoothness properties of optimization-based controllers with CBFs. Nevertheless, the result is only applicable to continuously differentiable CBFs. In Chapter 3, we apply NCBFs to the problem of connectivity maintenance to highlight the continuity issue mentioned here.

Connectivity Maintenance is an important concept for multi-robot systems. In order to be able to interchange information across the network, the interaction graph must be connected. The concept of algebraic connectivity [GR01] of a graph, also known as Fiedler eigenvalue [Fie73], characterizes the connectivity of a network graph by transforming it into an eigenvalue computation problem. For multi-robot systems, the network graph is dynamically changing as the robots' states evolves and they navigate through their tasks. Typically, robot network graphs are determined via proximity graphs [BCM09, ZP15], where the degree of connectivity changes along the robots' trajectories. Connectivity maintenance of dynamic graphs can be categorized into two approaches, local and global, depending on how connectivity is enforced. In the local approach, connectivity is maintained by reasoning over the connections present in the initial graph. This includes the direct method of preserving all initial connections, see e.g., [JE07], which limits the graph to one arrangement. This method can be improved by considering instead multiple-hops neighbors and

allowing rearrangements in the edges [ZP05, SC12], but its flexibility is still limited by the initial robot configuration. The global approach reasons more broadly over network connectivity using network-wide metrics such as algebraic connectivity. Under this approach, we find works that pose connectivity as a problem of maximizing algebraic connectivity [Boy06, KM06]. The idea is to find a robot motion that will increase the algebraic connectivity. A decentralized implementation of this idea is explored in [dGJ06]. Nevertheless, maximizing the algebraic connectivity in all scenarios can be overly restrictive. In this regard, [SCS13, SC09] introduce more flexibility by allowing algebraic connectivity to decrease when its value is large.

We rely on set-valued theory to address continuity of the feedback controller. In safe stabilization, the feasible control set, given jointly by a CLF and a CBF, can be viewed as a set-valued map that changes across states. The idea behind set-valued map selection is to pick a control from each feasible control set at each system state to construct a single-valued function. One important selection theorem to note is Michael's theorem [Mic56], which shows the existence of a continuous feedback under mild conditions. Unfortunately, the proof is not constructive. One simple way to find a continuous feedback is to use a minimum-norm controller [FK96], i.e., an optimization-based controller where the objective function is a norm. Another notable work [YJ06] generalizes minimum-norm controllers with a guide function. For a more general objective function, Berge Maximum Theorem [AB99, Thm. 17.31] is a well-known result in parametric optimization that guarantees continuity of the controller based on continuity of its constraint map. These tools mentioned so far only guarantee continuity of the selection function.

Not many works deal with the problem of smooth selection. The book [FK96] discusses Lipschitz selections, but again does not give a constructive function. Smooth selection for polytope set-valued functions is explored in [LD97, Luc97], which suggests using the vertex of the polytope,

but smoothness is only guaranteed almost everywhere, and if the result is applied in the context of controls, the control signal can be unnecessarily large. In controls, the idea of smooth stabilization begins with [Art83], which shows that there exists a continuous control feedback for a control-affine system with a differentiable CLF. As briefly mentioned in [Son89], the idea can be extended to guarantee a smooth control feedback. However, the proof does not provide a construction of such feedback. This motivates [Son89] to use the analyticity of the root of a quadratic function with respect to its parameters to formulate the famous Sontag's universal formula for stabilization. As its name suggests, the formula does not take into account safety, which would further limit the feasible control option. It should be noted that there are extension to the universal formula in [LS91, LS95], which consider constrained inputs of $u \in [0, \infty)$ and $u \in [0, 1]$. However, these constraints are static, unlike constraints from CBF that change across states. Chapter 4 explores a new way to find a smooth selection function using the weighted centroid of the feasible set.

1.1.2 Event-Triggered Control

The event-triggered framework [Tab07, HJT12, HFO+17] seeks to determine criterions to employ opportunistically the available control resources (e.g., actuation, sensing, communication) in order to produce efficient implementations on digital systems. Such criterions, called triggers, are commonly obtained by examining the evolution under aperiodic sample-and-hold executions of the Lyapunov certificates valid for their continuous-time counterparts. This can be done in a derivative-based fashion, i.e., by monitoring the time derivative of the certificate, see e.g., [Tab07, APDN16, PTNA15, HDT11, KAH17], or in a function-based fashion, i.e., by directly monitoring the value of the certificate, see e.g., [VMB09, DMGC11, MAT09]. Both approaches

are widely applicable. However, derivative-based approaches tend to be conservative because they are evaluated at the current system state without taking into account how much the certificate has decreased since the last update. This is tackled in dynamic event-triggering [Gir15] by introducing an extra variable to store an estimate of this decrease and incorporate it into the trigger design. On the other hand, function-based designs suffer from the lack of robustness to disturbances in the value of the certificate. The work [SP11] uses both frameworks to mitigate these drawbacks by estimating how much the certificate will decrease after each trigger, which constitutes another source of conservatism, together with its reliance on time triggering.

Chapter 5 takes a different approach to combine the derivative- and function-based design methodologies inspired by the concept of control barrier functions, and particularly, Nagumo's Theorem, see e.g., [WA07, AXGT17, BM07, ACE+19]. The basic insight is to incorporate into the trigger design the performance residual, i.e., how well the system is doing in regards to a prescribed performance specification. This specification plays the role of the "barrier" that the system should not exceed. This makes it possible to allow the certificate to deviate from monotonically decreasing at all times, with the amount of deviation allowed specified as a function of the size of the performance residual. Interestingly, the dynamic event-triggered approach [Gir15] mentioned above can be naturally interpreted within the framework proposed in this dissertation.

Our technical approach also builds on the literature of event-triggered approaches applied to the distributed control of network systems, see e.g., [WL09, MT11, TC14, BDH16, DFJ12, NGC19, BN21] and references therein. One known issue in this context is that Zeno behavior may arise as a result of the partial availability of information to individual agents, despite it being ruled out for its centralized counterpart. In such scenarios, it is common to use time regularization [MT11, TC14, BDH16], i.e., preventing by design any update before certain fixed time (usually

the minimum inter-event time from the centralized design) has elapsed. This requires an offline computation and the resulting executions may behave like periodic time-triggered ones. An alternative way of avoiding Zeno behavior is to allow for the violation of the monotonic decrease of the certificate at all times, see e.g., [DH12, GLM+12], at the cost of only achieving practical stability. Other works avoid Zeno behavior by either requiring stronger system assumptions on the type of certificates [WL11, GA12] or their solutions are problem-specific [CKSD14, BN21]. Here, we combine the performance-barrier-based framework with dynamic average consensus [KSC+19] to synthesize a Zeno-free distributed design that ensure asymptotic convergence for a general class of nonlinear systems.

As investigated in Chapter 6, Zeno behavior also appears when event-triggered control is used in the context of safety. One challenge in developing event-triggered control approaches for safety is ensuring that the time between events, or *interevent times*, are lower bounded [BH14, PSH19, BN21]. Such bounds ensure that the resulting controller and trigger law is free from Zeno behavior. The notion of Input-to-State Stability (ISS) has been used to prove the existence of these bounds in the context of stabilization [LWL12, Tab07]. In contrast to the task of stabilization, in the context of safety the dynamics of the system, and thus the error dynamics, are not required to vanish as the quantity dictating the triggering of events vanishes. This can lead to events occurring in rapid succession. Similar problem has been observed in the context of output-based event-triggered control [DH12]. To combat the issue, a mixed event-triggered mechanism [BH14, HJT12] is employed to address the problem. The idea is to add a buffer constant to define the trigger condition in order to guarantee a minimum interevent time. Following suit, Chapter 6 uses the same strategy together with the notion of an *input-to-state safe set* (ISSf) [KA18] to propose a Zeno-free event-triggered control law for safety certificate.

Chapter 7 demonstrates the versatility of the event-triggered control by using it in the context of human-robot interaction for multiobjective optimization. Interactive approaches in multiobjective optimization involve an algorithmic strategy that "interacts" with a human supervisor to determine an acceptable solution to the problem. A comprehensive survey on interactive approaches can be found in [MRW08], which groups different techniques into three main categories: the trade-off approach, the reference points approach, and the classification method. Algorithmic solutions often combine elements of several of these categories. In the scenarios considered here, the optimization problems arise as the human-robot system explore the world, and hence global information is not available a priori. Most works in the trade-off approach focus on finding local information, usually related to the gradient of an implicit preference function at each iteration that ranks different outcomes [GDF72, Sak82, Yan99]. The implicit preference function is well-studied and utilized in utility theory. For example, its existence is proven in an important result in [Deb54] under mild assumptions. Using an implicit preference function is common for solving a multiobjective problem, see e.g., [GDF72, LYW09, MRW08]. The above reference list is relatively old because newer methods often require global information, such as the knowledge of the optimizer of each objective function or the knowledge of the Pareto front. For a list of newer methods with brief summaries of them, we refer the readers to [XCC+18]. Event-triggered design approach has been used successfully in the context of optimization, usually in distributed settings where communication among agents is viewed as a limited resource, see e.g., [WL11, WAJ12, KCM15, RC16]. The application of resource-aware ideas to a multiobjective optimization setting, with the human as the resource available to the robot whose use should be minimized, are novel aspects of Chapter 7.

The literature of human-robot interaction has become vast with the accelerated pace of development in robotics. A good overview is captured by the survey [GS07]. Our work can be classi-

fied under the category of supervisory control. One of the many human factors often explored in the human-robot interaction is the amount of workload on the human, see e.g., [PST+15, SFK+06]. An important concept is neglect time and interaction time as presented in [CGDRON05]. Closely related is the concept of response and reaction time, which becomes important when the human needs to work and respond to robots in the real world [HA17]. We employ techniques from event-triggered control to accommodate these time factors. In dealing with time constraints, we utilize the flexibility of event-triggered control to deal with delay, see e.g., [DBH17, LWL12, HDI06, WRGL15]. Although the results there are not directly applicable to our presented problem because of the particular features of the human-robot setup, we follow the idea of bounding interevent times to deal with delay, cf. [Tab07, LWL12]. In addition, we also employ the novel idea in event-triggered control of allowing the certificate function to increase along trajectories. This idea can be found in dynamic triggering [DBH14, Gir15] and in designs based on performance-based-barrier triggering in Chapter 5, and allows us to consider more general constraints on human performance.

1.2 Statement of Contributions

The main focus of this dissertation is to explore how different objectives of control systems can be put together. These objectives include stability, trajectory safety, and resource conservation. Our contribution are structured in two blocks. We begin with the exploration on how the concept stability and safety together can dictate how the safe stabilizing controllers are designed. In particular, we investigate the smoothness properties of the feedback controllers. Next, we turn our attention to event-triggered control as an implementation method to conserve resources in the context of both stability and safety. We also employ event-triggered control to accommodate human

performance limitations in the context multiobjective optimization.

1.2.1 Smooth Safe Stabilization

The first part of the dissertation studies smooth safe stabilization. We look at the problem of designing controllers that can satisfy simultaneously conditions for stability and safety, given by a control Lyapunov function and a control barrier function, respectively. Our focus when designing the controllers are on their its smoothness properties.

Chapter 3 considers the problem of global connectivity maintenance of the robot system that is operating under some nominal control constraint maps (e.g., CLFs and CBFs). We synthesize two different set-valued constraint maps for global connectivity maintenance using ideas based NCBFs, and then we use the optimization-based design forumlation to propose a feedback controller. In our treatment, we find that the nonsmoothness of the CBF can cause an issue in the continuity of the feedback controller, and thus we make conservative adjustments to avoid abrupt changes in the constraint sets where the algebraic connectivity has a higher multiplicity. We illustrates the effectiveness of our results in a resource gathering problem, both in simulations and an experiment. The problem consists a group of four robots trying to reach towards its assigned location, but cannot do so directly without losing connectivity. We show that using our proposed results, each robot in the network completes its tasks with continuous feedback control inputs, and the robot network remains connected throughout.

Unlike Chapter 3, Chapter 4 deals with CBFs that are smooth. Particularly, we consider the feedback controller design problem given a smooth CLF and a smooth CBF, or alternatively two control-affine constraint maps, to be respected. Noticing that an optimization-based feedback

controller does not guarantee regularity beyond continuity, we tackle the problem of designing a feedback controller that retains the smoothness properties of the given constraint maps. Our design is based on the idea of finding a weighted centroid of the admissible control set, using the weights given by a probability density function of a normal distribution. Using our results, we are able to achieve exactly what Sontag's universal formula does, guaranteeing smoothness in the feedback controller. Furthermore, our result has the additional flexibility to deal with two control-affine constraint maps. In other words, we provide a formula for smooth safe stabilization.

Our contributions for this part go beyond the main contributions discussed above. In our analyses, we provide a few important auxiliary results results, which may have useful applications beyond the scope of this dissertation. One such result is the generalization of Artstein's theorem. We use this result to analyze the well-posedness of the safe stabilization problem. One question that we answer is in regard to the existence of a smooth feedback control satisfying given constraint maps. We use our result to deduce a mild and verifiable condition that guarantees our problem is well-posed. This result extends beyond the problem we consider. For instance, the Artstein-like result proposes the existence of a smooth controller for any number of nonempty intersecting constraint maps, suggesting the possibility of future work in the area of systems with multiple control objectives. Another important results are the continuity results for any intersection of eigenspaces. We use this result to study the change in algebraic connectivity, since its generalized gradient is related to its associated eigenspace. However, not only does this result directly contribute to the literature on regularity of algebraic connectivity, it also has the potential to be useful in other eigenvalue problems where the smoothness of the eigenvalues are important.

1.2.2 Resource Aware Implementation of Control Laws

The second part of the dissertation shifts the focus from the design to the implementation of control laws, with an emphasis on resource conservation. We use event-triggered control as the tool to examine the tight coupling between physical and cyber processes, and to prescribe, in a principled way, when to use the available resources while still guaranteeing a desired quality of service. In this part, motivated by the concept of multi-robot systems operating with multiple objectives, we develop various novel event-triggered control frameworks.

In Chapter 5, we revisit event-triggered control in the context of stabilization. We address the problem of developing a digital feedback implementation that simultaneously retains the stability properties, opportunistically updates the controller, and meets a prescribed performance. The contributions of the chapter are threefold. The first contribution is the synthesis of a novel framework for event-triggered control termed performance-barrier-based design. We combine derivative- and function-based designs by incorporating into the trigger criterion both the time derivative and the value of the certificate. The flexibility of the proposed approach stems from allowing the certificate to deviate from having to monotonically decrease at all times. In our design, a larger performance residual, measured as the difference between the prescribed performance and the value of the certificate, results in a larger amount potential deviation allowed. By construction, at any given state, the performance-barrier-based design enjoys a longer inter-event time than the derivative-based approach, while still achieving the prescribed performance. Our second contribution is the characterization of the implementability and asymptotic stability properties of nonlinear systems under the proposed framework. We introduce the concept of class-$\mathcal{K}$ performance specification function and establish, for general nonlinear systems, a uniform lower bound in the inter-event

times of the proposed design, thereby ruling out the possibility of Zeno behavior. For the particular case of exponential performance specifications, which includes the case of linear control systems, we provide an explicit expression of an improved minimum inter-event time with respect to the derivative-based approach. Our third contribution builds on this characterization to develop distributed triggers for network systems using the performance-barrier-based approach that ensure asymptotic correctness. Our distributed design makes use of dynamic average consensus to estimate, with some tracking error, the terms in the trigger criterion that require global information to be evaluated. The guarantees on the design then rely on its ability to tolerate the tracking errors. This is where we leverage the flexibility provided by the performance-barrier-based approach to rule out Zeno behavior in the network executions without using any time regularization. We illustrate the effectiveness of the proposed framework in a vehicle platooning problem.

Chapter 6 examines event-triggered control in the context of safety. The problem considered in this chapter is the same as the one in Chapter 5, except that instead of stability, trajectory safety is desired. Despite the many similarities between the two concepts, event-triggered control ideas do not translate well to the new settings. We provide an example that demonstrates how a naive triggering scheme imitating the one for stability leads to Zeno-behavior. To fix the issue we introduce the concept of *strong Input-to-State Safety* and proposes a Zeno-free triggering mechanism. We note importantly that the trigger design proposed can be used in conjunction with another for stability, e.g., the triggers proposed in Chapter 5, to implement a safe stabilizing controller in an opportunistic fashion.

In Chapter 7, we look at the possibility of using a human to help out robot systems to balance many control objectives they may have. We consider a convex multiobjective optimization problem where a robot (a simplification from a multi-robot system) works alongside a human to

find a Pareto optimal solution. Based on its knowledge of the multiple objective functions, the robot presents outcomes to the human, who expresses her preference among them. The human cannot express in closed form the function she uses to evaluate the outcomes, but can provide its gradient (this is a convenient abstraction of the ability of the human to express preferences about an outcome being better than another). Throughout the chapter, we consider models of increasing complexity about how the human can interact with the robot. Our contributions are multiple fold. Our first contribution considers the ideal case, where the human can respond instantaneously to the robot queries. We propose an event-triggered design that allows the robot to interact with the human in an opportunistic fashion as required by the solution of the overarching multiobjective optimization problem, thereby reducing human workload. Our design is based on examining the evolution of the value of the outcomes along the robot trajectories and ensuring that it is decreasing. We next move on to consider timing constraints on human performance. Our second contribution considers the "need to rest" case, where the human needs some time after responding to a query before she can respond to a new one. In effect, this means that the robot might not get the information it requires if two queries are formulated in quick succession. We examine to what extent our original trigger design case can be made valid for this case by tuning a design parameter and characterize the human resting times that can be tolerated. To accommodate longer resting times, we propose an alternative trigger design that allows the certificate to increase at times during the evolution, as long as it decreases when evaluated at consecutive human's queries. To do this, our technical treatment introduces the important concepts of critical time and grace period. Critical time refers to how long without human input and by how much the robot can guarantee the monotonic decrease of the certificate. After the critical time, grace period refers to the amount of time the robot can still wait without querying the human while the certificate potentially increases, but not beyond the

value it had when information was last received from the human. We show that this design can accommodate longer resting times than our original design. Our third contribution considers the "need to think" case, where the human needs some time before responding to a query. Our design is based on the robot anticipating the evolution of the certificate for the period of time the human may take in responding, and using this information to query the human sufficiently in advance by tuning appropriately a design parameter in our original design. Finally, our last contribution considers the model of human performance that combines both "need to rest" and "need to think" timing constraints. For each model, we provide a complete analytical treatment of the proposed design that includes establishing the monotonic decrease of the certificate, a uniform lower bound on the minimum time between consecutive queries (thereby ruling out Zeno behavior), and the asymptotic correctness of the resulting algorithm to the desired optimal solution. Throughout the paper, we provide explicit expressions of the lower bounds on the minimum interevent time which, together with the characterization of the convergence rates of the dynamics, provide a mean to assess the trade-offs between the frequency of human queries and the algorithm performance. Simulations on an example in multiobjective robot motion planning show the reductions in human workload obtained by the proposed event-triggered design versus algorithms that require continuous human involvement. We also illustrate the trade-offs between design convergence rate, human workload, and human response time.

Chapter 2

Preliminaries

This chapter reviews key concepts in which we rely on. These include concepts in graph theory, convex analysis, set-valued theory, nonsmooth analysis, and nonsmooth control barrier functions. We begin by introducing our basic notation.

2.1 Notation

The symbols $\mathbb{N}$, $\mathbb{R}$, $\mathbb{R}_{\geq 0}$, and $\mathbb{R}_{>0}$ represent the set of natural, real, real nonnegative, and real positive numbers, respectively. We write Sym^n for the space of $n \times n$ symmetric matrices with real values. For $m, n \in \mathbb{N}$, we denote $[m : n] = \{m, \dots, n\}$, and we write $[1 : n]$ simply as $[n]$. Given a finite set $\mathcal{I}$, $|\mathcal{I}|$ is its cardinality. The convex closure of a set S is represented by $\mathrm{co}(S)$. Given $\mathbf{x} \in \mathbb{R}^N$, $\|\mathbf{x}\|$ denotes its Euclidean norm. We use the symbol $\mathbb{1}$ for the vector of all ones (of appropriate dimension). The unit sphere in $\mathbb{R}^n$ is denoted by $\mathbb{S}^n = \left\{\mathbf{v} \in \mathbb{R}^n \mid \|\mathbf{v}\| = 1\right\}$. The open ball of radius $\delta > 0$ centered at $\mathbf{x}^* \in \mathbb{R}^N$ is $\mathbb{B}_\delta(\mathbf{x}^*) = \left\{\mathbf{x} \in \mathbb{R}^N \mid \|\mathbf{x} - \mathbf{x}^*\| < \delta\right\}$. Given matrices $\mathbf{A}, \mathbf{B} \in \mathbb{R}^{n \times n}$, the Frobenius product is $\mathbf{A} \cdot \mathbf{B} = \sum_{i,j} \mathbf{A}_{ij} \mathbf{B}_{ij}$. We note the property that $\mathbf{v}\mathbf{v}^\top \cdot \mathbf{A} = \mathbf{v}^\top \mathbf{A} \mathbf{v}$,

for $\mathbf{v} \in \mathbb{R}^n$. The Frobenius norm is given by $\|\mathbf{A}\|_F = (\mathbf{A} \cdot \mathbf{A})^{1/2}$. A continuous function $\alpha : \mathbb{R} \to \mathbb{R}$ is of extended class $\mathcal{K}$ if α is strictly increasing, and $\alpha(0) = 0$. In addition, the function is class-$\mathcal{K}_\infty$ if it also satisfies $\lim_{r\to\infty} \alpha(r) = \infty$. A function $f : \mathbb{R}^n \to \mathbb{R}^n$ is locally Lipschitz if, for every compact set $S_0 \subset \mathbb{R}^n$, there exists $L > 0$ such that $\|f(x) - f(y)\| \leq L\|x - y\|$, for all $x, y \in S_0$. We write supp(f) for the support of the function f, i.e., the set of $\mathbf{x}$ where $f(\mathbf{x}) \neq 0$. For a continuously differentiable f, $J_f : \mathbb{R}^n \to \mathbb{R}^{m\times n}$ denotes its Jacobian matrix. For f and $g : \mathbb{R}^m \to \mathbb{R}$, the composition of functions is $g \circ f : \mathbb{R}^n \to \mathbb{R}$, i.e., $(g \circ f)(x) = g(f(x))$ for $x \in \mathbb{R}^n$.

2.2 Graph Theory

A graph is a triplet $\mathcal{G} = (V, E, \mathbf{A})$, where V is a set of vertices, $E \subseteq V \times V$ is a set of edges, and $\mathbf{A} \in \mathbb{R}^{|V|\times|V|}$ is the adjacency matrix, with $\mathbf{A}_{ij} > 0$ if $(i, j) \in E$, and $\mathbf{A}_{ij} = 0$ otherwise. The graph is undirected if $\mathbf{A}$ is symmetric. A path is an ordered sequence of vertices such that all pairs of consecutive vertices are elements of E. The graph is connected if there exists a path between any two vertices. The degree matrix $\mathbf{D} \in \mathbb{R}^{|V|\times|V|}$ is a diagonal matrix whose ith element is $D_{ii} = \sum_{j\in V} \mathbf{A}_{ij}$.

2.2.1 Laplacian Spectrum

The Laplacian matrix $\mathbf{L}$, defined by $\mathbf{L} := \mathbf{D} - \mathbf{A}$, is symmetric and positive semidefinite, and consequently has real and nonnegative eigenvalues. We denote these eigenvalues with $\phi_m \in \mathbb{R}_{\geq 0}$, ordering them in an increasing manner with the subscripts $m \in [|V|]$, i.e., $0 = \phi_1 \leq \phi_2 \leq \ldots \leq \phi_{|V|}$. The eigenvalue $\phi_1 = 0$ is simple (with associated eigenvector $\mathbb{1}$) if and only if the graph is connected. This justifies the terminology of ϕ_2 as the algebraic connectivity (also known as

Fiedler eigenvalue). For network systems, graphs are used to described the underlying interaction topology, and they can vary according to the system states. A state-dependent graph $\mathbf{x} \mapsto \mathcal{G}(\mathbf{x})$ is called a *proximity graph* [BCM09]. In such a case, the Laplacian matrix $\mathbf{x} \mapsto \mathbf{L}(\mathbf{x})$ is then also a function of state. We define the function $\lambda_m(\mathbf{x}) := (\phi_m \circ \mathbf{L})(\mathbf{x})$ to be the Laplacian's eigenvalues as a function of the state. Given a trajectory $t \mapsto \mathbf{x}(t)$, a graph remains robustly connected at all times if $\lambda_2(\mathbf{x}(t)) \geq \varepsilon$, where $\varepsilon \in \mathbb{R}_{>0}$ is a threshold parameter providing a robustness margin in ensuring connectivity.

2.2.2 Dynamic Average Consensus

Consider a group of N agents communicating over an undirected graph $\mathcal{G}$. Each agent $i \in \mathcal{V} = [N]$ has a continuously differentiable reference signal $W_i : [0, \infty) \to \mathbb{R}$. Dynamic average consensus aims at making the agents track asymptotically the average of the reference signals. For convenience, let $W = (W_1, \ldots, W_N)$. Here we employ the dynamic average consensus algorithm [KSC+19],

$$\dot{y} = \dot{W} - \rho \mathbf{L} y, \tag{2.1}$$

where each component of $y \in \mathbb{R}^N$ is the agents' estimate of the average, $\rho > 0$ is a rate of convergence parameter, and $\mathbf{L}$ is the Laplacian matrix of the graph. The following result shows that with the correct initialization and a suitable assumption on the evolution of W, each state y_i asymptotically tracks the average $\mathbb{1}^\top W(t)/N$ of the reference signal. The following result is a refinement of [KSC+19, Thm.2] to reference signals whose time derivative is bounded exponentially.

Lemma 2.2.1. ***(Tracking Error Bound).*** *Consider the dynamic average consensus dynamics* (2.1) *with a reference signal W whose time derivative is bounded exponentially, i.e., $\|\dot{W}(t)\| \leq$*

$c_{\dot{W}} \exp(-rt)$ *with a constant* $c_{\dot{W}} > 0$*, for time* $t \in [0, s)$*. Define the tracking error as* $\epsilon := y - \mathbb{1}\mathbb{1}^\top W/N$*. If the initialization of* y *is such that* $\mathbb{1}^\top y(0) = \mathbb{1}^\top W(0)/N$*, then the tracking error is also bounded for time* $t \in [0, s)$ *as*

$$\|\epsilon(t)\| \leq \frac{c_{\dot{W}}}{\rho\lambda_2 - r} \exp(-rt) + \left(\|\epsilon(0)\| - \frac{c_{\dot{W}}}{\rho\lambda_2 - r} \right) \exp(-\rho\lambda_2 t) \tag{2.2}$$

where λ_2 *is the second smallest eigenvalue of the Laplacian matrix* $\mathbf{L}$*.*

Proof. We begin the proof by writing the dynamics of the tracking error,

$$\begin{aligned}
\dot{\epsilon} &= \dot{y} - \mathbb{1}\mathbb{1}^\top \dot{W}/N \\
&= \dot{W} - \rho\mathbf{L}(\epsilon - \mathbb{1}\mathbb{1}^\top W/N) - \mathbb{1}\mathbb{1}^\top \dot{W}/n \\
&= -\rho\mathbf{L}\epsilon + (\mathbf{I} - \mathbb{1}\mathbb{1}^\top/N)\dot{W}
\end{aligned}$$

where we have used the fact that $\mathbf{L}\mathbb{1} = 0$. Note also that $\mathbb{1}^\top\dot{\epsilon} = 0$, so $\mathbb{1}^\top\epsilon = 0$ by construction. Hence, at all time, there is no component of ϵ along the eigenvector $\mathbb{1}$ associated with the eigenvalue 0 of the Laplacian matrix $\mathbf{L}$. Consequently, we can bound

$$\begin{aligned}
\frac{d}{dt}\|\epsilon\|^2 &= -\rho\epsilon^\top(\mathbf{L} + \mathbf{L}^\top)\epsilon + 2\epsilon^\top(\mathbf{I} - \mathbb{1}\mathbb{1}^\top/N)\dot{W} \\
&\leq -2\rho\lambda_2\|\epsilon\|^2 + 2\|\epsilon\|\|\mathbf{I} - \mathbb{1}\mathbb{1}^\top/N\|\|\dot{W}\| \\
&\leq -2\rho\lambda_2\|\epsilon\|^2 + 2c_{\dot{W}} \exp(-rt)\|\epsilon\|
\end{aligned}$$

for time $t \in [t_k, t_{k+1})$. It can be verified through substitution that the solution

$$v = \frac{c_{\dot{W}}}{\rho\lambda_2 - r} \exp(-rt) + \left(v(0) - \frac{c_{\dot{W}}}{\rho\lambda_2 - r} \right) \exp(-\rho\lambda_2 t).$$

satisfies the Bernoulli differential equation [Inc56]

$$v\frac{dv}{dt} = -\rho\lambda_2 v^2 + c_{\dot{W}} \exp(-rt)v.$$

Note that when $v \neq 0$, this reduces to

$$\frac{dv}{dt} = -\rho\lambda_2 v + c_{\dot{W}} \exp(-rt),$$

which is linear, with the right-hand side locally Lipschitz in v. Then, with $v(0) = \|\epsilon(0)\| \neq 0$, we can deduce $\|\epsilon\|^2 \leq v^2$ by applying the Comparison Lemma [Kha02, Lemma 3.4]. Whenever $\|\epsilon\| = 0$, it is possible (depending on $\dot{W}$) for ϵ to remain zero for some time interval. On such interval, the Comparison Lemma does not apply; however, the case is trivial, and the bound $\|e\|^2 \leq v^2$ still holds. Finally, by noting that $v \geq 0$ because $v(0) \geq 0$, we obtain $\|\epsilon\| \leq |v| = v$ as stated. □

2.3 Convex Analysis

For a twice continuously differentiable, scalar-valued function $g : \mathbb{R}^n \to \mathbb{R}$, we let $\nabla g : \mathbb{R}^n \to \mathbb{R}^n$ and $\nabla^2 g : \mathbb{R}^n \to \mathbb{R}^{n\times n}$ denote its gradient and Hessian functions. g is convex on $\mathcal{S}$ if, for all $x \in \mathcal{S}$, $\nabla^2 g(x) \succeq 0$; strictly convex if $\nabla^2 g(x) \succ 0$; and strongly convex if there exists $\mu > 0$ such that $\nabla^2 g(x) \succeq \mu I_n$. If g is strongly convex on $\mathcal{S}$, its sublevel sets contained in $\mathcal{S}$ are bounded.

This implies that there exists $M > 0$ such that $\nabla^2 g \preceq M I_n$ on S. In fact, if x^* is the minimizer of g, then

$$\frac{1}{2M}\|\nabla g(x)\|^2 \leq g(x) - g(x^*) \leq \frac{1}{2\mu}\|\nabla g(x)\|^2. \tag{2.3}$$

2.4 Set-Valued Theory

A set-valued map $\mathcal{U} : \mathbb{R}^n \rightrightarrows \mathbb{R}^m$ assigns a subset of $\mathbb{R}^m$ to each point in $\mathbb{R}^n$. A set-valued map $\mathcal{U}$ is closed-valued, convex-valued, compact-valued, and has a nonempty interior if its image at each point of its domain is closed, convex, compact, and has a nonempty interior, respectively. All set operations, e.g., union and intersection, between set-valued maps are performed pointwise. We consider set-valued maps arising from a single-valued function $g : \mathbb{R}^N \times \mathbb{R}^M \to \mathbb{R}^d$ as follows:

$$\mathcal{U}(\mathbf{x}) = \{\mathbf{u} \in \mathbb{R}^M \mid g(\mathbf{x}, \mathbf{u}) \leq 0\}. \tag{2.4}$$

Given $\mathbf{x}$, we say $\mathbf{u}$ strictly satisfies $\mathcal{U}(\mathbf{x})$ if $g(\mathbf{x}, \mathbf{u}) < 0$.

2.4.1 Continuity of Set-Valued Maps

The concept of continuity for set-valued maps is more intricate than the one for single-valued functions. Continuity of set-valued maps is often broken down into different types of hemicontinuity. Here we present the two that we rely on: upper and lower hemicontinuity[1].

Definition 2.4.1. ***(Set-Valued Map Continuity [Bor85]).*** *A set-valued map $\mathcal{U} : \mathbb{R}^N \rightrightarrows \mathbb{R}^M$ is*

- *upper hemicontinuous (UHC) at $\mathbf{x}$ if for any neighborhood $\bar{\mathcal{U}}$ of $\mathcal{U}(\mathbf{x})$, there exists $\delta > 0$*

[1] Sometimes referred to as semicontinuity, see e.g., [LS85].

such that, if $\|\mathbf{x} - \mathbf{x}'\| < \delta$, *then* $\mathcal{U}(\mathbf{x}') \subset \bar{\mathcal{U}}$;

- *lower hemicontinuous (LHC) at* $\mathbf{x}$ *if for each* $\mathbf{u} \in \mathcal{U}(\mathbf{x})$ *and for any sequence* $\{\mathbf{x}^k\}_{k\in\mathbb{N}}$ *converging to* $\mathbf{x}$, *there exists a sequence* $\{\mathbf{u}^k\}_{k\in\mathbb{N}}$ *converging to* $\mathbf{u}$ *with* $\mathbf{u}^k \in \mathcal{U}(\mathbf{x}^k)$;

- *continuous at* $\mathbf{x}$ *if it is both UHC and LHC at* $\mathbf{x}$.

Note here that UHC and LHC are equivalent for single-valued functions. For convenience, the map (hemi)continuous if it is (hemi)continuous for all $\mathbf{x}$. Interestingly for set-valued maps of the form (2.4), even g being continuous is not enough to ensure the map $\mathcal{U}$ is continuous. In fact, to ensure UHC and LHC, we will resort to the additional requirements stated in the following results.

Lemma 2.4.2. ***(UHC Requirements [Sti18, Lem 5.7]).*** *Assume* g *is continuous. If* g *is convex in* $\mathbf{u}$, *and* $\mathcal{U}(\mathbf{x})$ *is nonempty and compact at* $\mathbf{x}$, *then* $\mathcal{U}$ *is UHC at* $\mathbf{x}$. □

Lemma 2.4.3. ***(LHC Requirements [Sti18, Lem 5.2]).*** *Assume* g *is continuous. If* $\mathcal{U}$ *has a nonempty interior and is convex-valued, then* $\mathcal{U}$ *is LHC.* □

In our treatment, we also rely on various results on how hemicontinuity is preserved under set-valued map intersections.

Lemma 2.4.4. ***(Intersection of UHC maps [Bor85, 11.21a]).*** *Let the set-valued maps* $\mathcal{U}_1, \mathcal{U}_2 : \mathbb{R}^N \rightrightarrows \mathbb{R}^M$ *be UHC and closed-valued at* $\mathbf{x}$. *The intersection* $\mathcal{U}_1 \cap \mathcal{U}_2$ *is also UHC at* $\mathbf{x}$ *if it is nonempty at* $\mathbf{x}$. □

Lemma 2.4.5. ***(Intersection of LHC maps [LS85, Thm. B]).*** *Let the set-valued maps* $\mathcal{U}_1, \mathcal{U}_2 : \mathbb{R}^N \rightrightarrows \mathbb{R}^M$ *be LHC and locally convex-valued at* $\mathbf{x}$. *The intersection* $\mathcal{U}_1 \cap \mathcal{U}_2$ *is also LHC at* $\mathbf{x}$ *if it has a nonempty interior at* $\mathbf{x}$. □

2.5 Nonsmooth Analysis

Here we present basic notions of nonsmooth analysis following [Cla83]. Given a locally Lipschitz function $h : \mathbb{R}^N \to \mathbb{R}$, the generalized directional derivative of h at $\mathbf{x} \in \mathbb{R}^N$ in the direction $\mathbf{d} \in \mathbb{R}^N$ is

$$h^\circ(\mathbf{x}; \mathbf{d}) = \limsup_{\mathbf{x}' \to \mathbf{x}, s \downarrow 0} \frac{h(\mathbf{x}' + s\mathbf{d}) - h(\mathbf{x}')}{s}.$$

The generalized gradient of h at $\mathbf{x}$ is then given by

$$\partial h(\mathbf{x}) = \{\zeta \in \mathbb{R}^N \mid h^\circ(\mathbf{x}; \mathbf{d}) \geq \zeta^\top \mathbf{x},\ \forall \mathbf{d} \in \mathbb{R}^N\}.$$

If the function h is continuously differentiable at $\mathbf{x}$, the generalized gradient is a singleton, $\partial h(\mathbf{x}) = \{\nabla h(\mathbf{x})\}$.

In our analysis, we find it useful to describe how a nonsmooth function changes along the trajectories of a dynamical system. Consider the nonlinear system,

$$\dot{\mathbf{x}} = f(\mathbf{x}, \mathbf{u}), \tag{2.5}$$

with $f : \mathbb{R}^N \times \mathbb{R}^M \to \mathbb{R}^N$, where $\mathbf{x}$ is the state and $\mathbf{u}$ is the control input. The weak set-valued Lie derivative [GCE17b, SP94] is

$$\mathcal{L}_f h(\mathbf{x}, \mathbf{u}) = \left\{\zeta^\top f \in \mathbb{R} \mid \zeta \in \partial h(\mathbf{x})\right\}.$$

The Lie derivative describes the rate of change of h along a trajectory of the system. Let $t \to \mathbf{u}(t)$

be a control signal, and $t \to \mathbf{x}(t)$ be a Carathéodory solution[2] to the differential equation (2.5), then

$$\frac{d}{dt}h(\mathbf{x}(t)) \in \mathcal{L}_f h(\mathbf{x}(t), \mathbf{u}(t)), \ a.e.\ t \geq 0. \tag{2.6}$$

In essence, the weak set-valued Lie derivative contains all the possible rates of change of the function h along a solution of the dynamical system.

2.6 Nonsmooth Control Barrier Functions

We use Nonsmooth Control Barrier Functions (NCBF) [GCE17b] to establish forward invariance of a desired set. Consider the dynamical system (2.5) and a set $\mathcal{C} = \{\mathbf{x} \in \mathbb{R}^N \mid h(\mathbf{x}) \geq 0\}$ with a locally Lipschitz continuous $h : \mathbb{R}^N \to \mathbb{R}$, referred to as a nonsmooth control barrier function. Indeed, for a continuous trajectory $t \to \mathbf{x}(t)$, we can ensure h remains positive if we constrain h from decreasing whenever $h(\mathbf{x}(t)) = 0$. This can be done by imposing a constraint, as a function of network state $\mathbf{x}$, on our choice of the input $\mathbf{u}$ with a set-valued map

$$\mathcal{U}(\mathbf{x}) = \left\{\mathbf{u} \in \mathbb{R}^M \mid \min \mathcal{L}_F h(\mathbf{x}, \mathbf{u}) \geq -\alpha(h(\mathbf{x}))\right\},$$

where α is a locally Lipschitz extended class $\mathcal{K}$ function. Given (2.6), by taking the minimum element of the set-valued Lie derivative, the constraint map enforces the bound even for the worst-case rate of change of h. Note importantly that the above constraint map does not only limit the choice of $\mathbf{u}$ for $\mathbf{x}$ at the boundary of $\mathcal{C}$ where $h(\mathbf{x}) = 0$, but also in the interior where $h(\mathbf{x}) > 0$, even when it is not necessary. Rather than outright allowing any choice of $\mathbf{u}$, the constraint map gradually

[2] A Carathéodory solution is an absolutely continuous trajectory that satisfies the system dynamics at almost every time, in the sense of Lebesgue measure.

becomes stricter for states closer to the boundary. The idea here is to begin consider the necessary constraint as the trajectory approaches the boundary, and thereby provide some robustness to how the set $\mathcal{C}$ is rendered forward invariant.

Part I

Smooth Safe Stabilizing Feedback Controllers

Chapter 3

Connectivity Maintenance as a Safety Objective

This chapter considers the problem of maintaining global connectivity of a multi-robot system while executing a desired coordination task. Our approach builds on optimization-based feedback design formulations, where the nominal cost function and constraints encode desirable control objectives for the resulting input. We take advantage of the flexibility provided by control barrier functions to produce additional constraints that guarantee that the resulting optimization-based controller is continuous and maintains network connectivity. Our solution uses the algebraic connectivity of the multi-robot interconnection topology as a control barrier function and critically embraces its nonsmooth nature. The technical treatment combines elements from set-valued theory, nonsmooth analysis, and algebraic graph theory to imbue the proposed constraints with regularity properties so that they can be smoothly combined with other control constraints. We provide simulations and experimental results illustrating the effectiveness and continuity of the proposed approach in a resource gathering problem.

3.1 Problem Statement

Consider a group of n robots, evolving according to a single-integrator dynamics of the form

$$\dot{x}_r = u_r, \quad \forall r \in [n], \tag{3.1}$$

where $x_r \in \mathbb{R}^{d_r}$ and $u_r \in \mathbb{R}^{d_r}$ are the state and the control input associated with the r-th robot (note that the state dimensions of each robot might be different). For convenience, we define state and input variables for the network system as follows: let $N = \sum_{r\in[n]} d_r$ and denote $\mathbf{x} = \left[x_1^\top, \dots, x_n^\top\right]^\top \in \mathbb{R}^N$ and $\mathbf{u} = \left[u_1^\top, \dots, u_n^\top\right]^\top \in \mathbb{R}^N$. We use the shorthand notation $f_{\text{si}} : \mathbb{R}^N \times \mathbb{R}^N \to \mathbb{R}^N$ to refer compactly to the dynamics (3.1) for the whole group of agents. The underlying interaction topology is specified by a proximity graph $\mathbf{x} \mapsto \mathcal{G}(\mathbf{x}) = ([n], E(\mathbf{x}), \mathbf{A}(\mathbf{x}))$, for which we assume that the function $\mathbf{x} \mapsto \mathbf{A}(x)$ is continuously differentiable[1].

We are interested in designing a continuous controller $\mathbf{k} : \mathbb{R}^N \to \mathbb{R}^N$ such that the network system under feedback $\mathbf{u} = \mathbf{k}(\mathbf{x})$ enjoys some desirable performance and asymptotic guarantees. Continuity is an important property, both from a theoretical and practical viewpoint. Regarding the former, continuity guarantees the existence of Carathéodory solutions [Hal69, Thm. 5.1]. At the same time, continuity makes it easier for the desired feedback control signal to be implemented on digital platforms.

A commonly used design methodology to synthesize controllers is based on optimization

[1]This assumption is satisfied by commonly employed weight assignments [SC12, GSU17].

and takes the form

$$\mathbf{k}_{\text{opt}}(\mathbf{x}) = \underset{\mathbf{u}\in\mathcal{U}(x)}{\operatorname{argmin}}\ J(\mathbf{x},\mathbf{u}), \tag{3.2}$$

where $J : \mathbb{R}^N \times \mathbb{R}^N \to \mathbb{R}$ is a cost function encoding some desirable objective (e.g., minimum-energy control specifications) and $\mathcal{U} : \mathbb{R}^N \rightrightarrows \mathbb{R}^M$ is a set-valued map encoding constraints on the control input at each $\mathbf{x}$ (e.g., bounds on magnitude). This formulation is flexible as it allows to address simultaneously different performance requirements: the map $\mathcal{U}$ can be itself an intersection of multiple set-valued maps, each representing a different control constraint from a performance aspect (input boundedness, infinitesimal decrease of certificate).

We consider the scenario where the robot group has a nominal control constraint map $\mathbf{x} \mapsto \mathcal{U}_{\text{nom}}(\mathbf{x})$, defined via a function $g_{\text{nom}} : \mathbb{R}^N \times \mathbb{R}^N \to \mathbb{R}^{d_{\text{nom}}}$ as

$$\mathcal{U}_{\text{nom}}(\mathbf{x}) = \{\mathbf{u} \in \mathbb{R}^N \mid g_{\text{nom}}(\mathbf{x},\mathbf{u}) \leq 0\}.$$

The components of g_{nom} here represent constraints that the robot group must respect to achieve different control performances and goals. This nominal constraint map, however, does not encode any network connectivity constraint. We are then interested in solving the following problem.

Problem 1. ***(Continuous Connectivity Controller Design Problem).*** Consider the multi-robot system (3.1) operating with the optimization-based controller (3.2). Design the constraint map $\mathcal{U}$ so that:

- the controller $\mathbf{k}_{\text{opt}}$ is continuous;
- the nominal constraint map is respected, i.e., $\mathcal{U} \subseteq \mathcal{U}_{\text{nom}}$;

- the underlying graph $\mathcal{G}$ remains connected at all time. •

In order to make Problem 1 tractable, we make some assumptions on the cost function J and the nominal constraint map $\mathcal{U}_{\text{nom}}$. As one can expect, continuity of $\mathbf{k}_{\text{opt}}$ is related to continuity of the cost function J and the constraint map $\mathcal{U}$. In this regard, Berge Maximum Theorem [AB99, Thm. 17.31] states that, if J and $\mathcal{U}$ are continuous, $\mathcal{U}$ is compact-valued, and the resulting $\mathbf{k}_{\text{opt}}$ is single-valued, then $\mathbf{k}_{\text{opt}}$ is continuous. Based on this result, we make the following continuity assumption.

Assumption 3.1.1. ***(Continuity Assumption on Cost and Nominal Constraint).*** *The functions J and g_{nom} are continuous.* •

We do not make a direct assumption on the continuity of $\mathcal{U}_{\text{nom}}$ for greater generality. In fact, such assumption would rule out many commonly used constraint maps (e.g., control affine constraint maps are typically not UHC). As such, we rely instead on the following assumption.

Assumption 3.1.2. ***(Convexity Assumption on Cost and Nominal Constraint).*** *The function J is strictly convex in $\mathbf{u}$ and g_{nom} is convex in $\mathbf{u}$.* •

Although convexity is not required by Berge Maximum Theorem, the above assumption is justified by several reasons. First, the assumption helps us make the optimization problem that defines the controller a convex program, which opens the way to employing available convex optimization methods to compute the controller. In addition, the strict convexity assumption also ensures that the controller is single-valued for any given $\mathbf{x}$, which is a requirement of Berge Maximum Theorem. More importantly, the convexity assumption also opens up the possibility of $\mathcal{U}$ being defined by unbounded constraints, despite the compact-valued requirement in Berge Maximum Theorem. To reconcile this, we consider the sublevel sets of J. Suppose for each $\mathbf{x}$, there

exists a control $\mathbf{x} \mapsto \bar{\mathbf{u}}(\mathbf{x})$ such that $\bar{\mathbf{u}}(\mathbf{x}) \in \mathcal{U}(\mathbf{x})$, and define

$$\mathcal{J}_{\bar{\mathbf{u}}}(\mathbf{x}) = \{\mathbf{u} \in \mathbb{R}^N \mid \|J(\mathbf{x}, \mathbf{u})\| \leq \|J(\mathbf{x}, \bar{\mathbf{u}}(\mathbf{x}))\| + \delta_J\} \tag{3.3}$$

with $\delta_J \in \mathbb{R}_{>0}$. Note that this set-valued map is compact-valued due to strict convexity of J. In addition, when $\mathcal{J}_{\bar{\mathbf{u}}}$ is considered in conjunction with $\mathcal{U}$, it is always inactive at the optimizer because $\bar{\mathbf{u}}$ is a feasible point. Consequently, for a properly designed $\mathcal{U}$, even if it is not compact-valued, we may consider $\mathcal{U} \cap \mathcal{J}_{\bar{\mathbf{u}}}$ as the constraint map without changing the optimizer at each $\mathbf{x}$ and apply Berge Maximum Theorem.

3.2 Discontinuity in the Naive Connectivity Maintenance Solution

In this section we make a first attempt at solving Problem 1 using algebraic connectivity as a nonsmooth control barrier function. We show that the proposed solution falls short because the resulting feedback controller is discontinuous. This exercise serves two purposes. On the one hand, it motivates the technical refinement pursued in our exposition later. On the other, it helps us pinpoint the obstructions associated with solving Problem 1, providing the necessary exposition for the rationale behind our solutions.

For maintaining connectivity, it seems natural to use the algebraic connectivity as a NCBF to guarantee λ_2 remains positive along the trajectory. This is essentially the approach taken in [CS20] (with the difference that we explicitly account for the nonsmoothness of λ_2 in the exposition here).

Consider the safe set of connected robot configurations

$$\mathcal{C}_\varepsilon := \left\{ \mathbf{x} \in \mathbb{R}^N \mid \lambda_2(\mathbf{x}) \geq \varepsilon \right\},$$

with $\varepsilon \in \mathbb{R}_{>0}$. Let $\mathbf{x} \mapsto h(\mathbf{x}) = \lambda_2(\mathbf{x}) - \varepsilon$ be our candidate NCBF. Resorting to the discussion of Section 2.6, we specify a constraint map for the purpose of connectivity maintenance.

Lemma 3.2.1. ***(Connectivity Maintenance Constraint Map).*** *Consider the multi-robot system* (3.1) *operating with a controller* $\mathbf{x} \mapsto \mathbf{k}(\mathbf{x})$. *Given a locally Lipschitz extended class* $\mathcal{K}$ *function* α*, define the constraint map*

$$\mathcal{U}_{\text{cm}}(\mathbf{x}) := \left\{ \mathbf{u} \in \mathbb{R}^N \mid \min \mathcal{L}_{f_{\text{si}}} \lambda_2(\mathbf{x}, \mathbf{u}) \geq -\alpha(\lambda_2(\mathbf{x}) - \varepsilon) \right\}.$$

If $\mathbf{k}(\mathbf{x}) \in \mathcal{U}_{\text{cm}}(\mathbf{x})$ *for all* $\mathbf{x} \in \mathcal{C}_\varepsilon$*, then for any initial connected network configuration* $\mathbf{x}_0 \in \mathcal{C}_\varepsilon$*,* $\lambda_2(\mathbf{x}(t)) \geq \varepsilon$ *along all Carathéodory solutions of the closed-loop system under* $\mathbf{u} = \mathbf{k}(\mathbf{x})$*, ensuring that network connectivity is maintained.*

Lemma 3.2.1 is a direct result of using $h(\mathbf{x}) = \lambda_2(\mathbf{x}) - \varepsilon$ as a NCBF, cf. [GCE17b, Thm.3]. Note importantly that connectivity maintenance is only guaranteed along Carathéodory solutions, which may not exist if $\mathbf{k}$ is not continuous. In particular, consider an optimization-based controller (3.2) naively defined with the connectivity maintenance constraint map,

$$\mathbf{k}_{\text{dis}}(\mathbf{x}) := \underset{u \in \mathcal{U}_{\text{cm}}(\mathbf{x})}{\operatorname{argmin}} J(\mathbf{x}, \mathbf{u}). \tag{3.4}$$

Unfortunately, this controller is not continuous. Indeed, this is because $\mathcal{U}_{\text{cm}}$ itself is not continuous

and does not meet the requirement of Berge Maximum Theorem.

To pinpoint the root cause of the discontinuity of $\mathcal{U}_{\text{cm}}$, we review the generalized gradient of the Laplacian's eigenvalues. Each eigenvalue function ϕ_m is globally Lipschitz with respect to the entries of the Laplacian matrix (cf., [SC12, Lem. 1] and [Lew96, Thm. 2.4]). As a result, if $\mathbf{L}$ is a continuously differentiable function of the network state, then $\lambda_m = \phi_m \circ \mathbf{L}$ is also Lipschitz. Therefore, generalized gradients are well-defined for the eigenvalue functions. Mathematically, the generalized gradient of ϕ is given by, cf. [SC12, Thm. 1],

$$\partial\phi_m(\mathbf{L}) = \text{co}\big\{\mathbf{v}_m\mathbf{v}_m^\top \mid \mathbf{v}_m \in \mathcal{V}_m(\mathbf{L})\big\}, \tag{3.5}$$

where $\mathcal{V}_m(\mathbf{L}) := \{\mathbf{v}_m \in \mathbb{S}^n \mid \mathbf{L}\mathbf{v}_m = \phi_m(\mathbf{L})\mathbf{v}_m\}$ is the set of normalized eigenvectors associated with ϕ_m. Using the nonsmooth chain rule [Cla83, Thm. 2.3.10], the expression for the weak set-valued Lie derivative [GCE17b, Rmk. 2.1] of λ_m with respect to the system (3.1) is

$$\mathcal{L}_{f_{\text{si}}}\lambda_m(\mathbf{x},\mathbf{u}) = \partial\phi_m(\mathbf{L}(\mathbf{x})) \cdot \Big(\sum_{i\in[N]} \frac{\partial \mathbf{L}}{\partial \mathbf{x}_i}\mathbf{u}_i\Big).$$

In the constraint map $\mathcal{U}_{\text{cm}}$, we use the minimal value of this set to bound the worst-case rate of change of λ_m along the control choice $\mathbf{u}$. Unfortunately, this minimal value is not a continuous function of $\mathbf{x}$. The following result helps us understand why.

Lemma 3.2.2. ***(Equivalent Minimization of the Eigenvalue's Set-Valued Lie Derivative).*** *Consider the multi-robot system* (3.1)*. For $m \in [N]$, let $(\mathbf{x},\mathbf{u}) \mapsto \mu_m(\mathbf{x},\mathbf{u})$,*

$$\mu_m(\mathbf{x},\mathbf{u}) := \min_{\mathbf{v}\in\mathcal{V}_m(\mathbf{x})} \mathbf{v}^\top\Big(\sum_{i\in[N]} \frac{\partial \mathbf{L}}{\partial \mathbf{x}_i}\mathbf{u}_i\Big)\mathbf{v}. \tag{3.6}$$

Then $\min \mathcal{L}_{f_{\text{si}}} \lambda_m(\mathbf{x}, \mathbf{u}) = \mu_m(\mathbf{x}, \mathbf{u})$ *for any* $\mathbf{x}$ *and* $\mathbf{u}$.

Proof. Let $D \in \partial \phi_m(\mathbf{L}(\mathbf{x}))$ be the element of the generalized gradient (3.5) corresponding to the minimum value in $\mathcal{L}_{f_{\text{si}}} \lambda_m(\mathbf{x}, \mathbf{u})$, i.e.,

$$\min \mathcal{L}_{f_{\text{si}}} \lambda_m(\mathbf{x}, \mathbf{u}) = D \cdot \Big(\sum_{i \in [N]} \frac{\partial \mathbf{L}}{\partial \mathbf{x}_i} \mathbf{u}_i \Big).$$

Since $D \in \mathbb{R}^{n \times n}$, there exists $n^2 + 1$ points $\{D_i\}_{i=1}^{n^2+1}$ (cf. Carathéodory theorem on convex hulls [Roc70, Thm. 17.1]) in $\left\{ \mathbf{v}_m \mathbf{v}_m^\top \mid \mathbf{v}_m \in \mathcal{V}_m(\mathbf{L}(\mathbf{x})) \right\}$ such that $D = \sum_{i=1}^{n^2+1} \sigma_i D_i$, with $\sum_{i=1}^{n^2+1} \sigma_i = 1$. Therefore,

$$\min \mathcal{L}_{f_{\text{si}}} \lambda_m(\mathbf{x}, \mathbf{u}) = (\sum_{i=1}^{n^2+1} \sigma_i D_i) \cdot \Big(\sum_{i \in [N]} \frac{\partial \mathbf{L}}{\partial \mathbf{x}_i} \mathbf{u}_i \Big).$$

Because of the minimization, we can reason by way of contradiction that $D = D_i$ for all $i \in [n^2 + 1]$. Hence, $D \in \left\{ \mathbf{v}_m \mathbf{v}_m^\top \mid \mathbf{v}_m \in \mathcal{V}_m(x) \right\}$, and $\mu_m(\mathbf{x}, \mathbf{u}) = \min \mathcal{L}_{f_{\text{si}}} \lambda_m(\mathbf{x}, \mathbf{u})$, concluding the proof. □

Lemma 3.2.2 transforms the minimization of the set-valued Lie derivative into an equivalent one with respect to eigenvectors. From this perspective, it is easy to identify the reason for the discontinuity in the minimum value. Whenever the multiplicity of an eigenvalue changes, so does the dimension of its eigenspace. Consequently, the minimization may abruptly drop in value. We rely on this key insight to synthesize our design in the next section.

3.3 Continuous Connectivity Maintenance Constraint Maps

In this section, we propose our solution to Problem 1. We construct two constraint maps for the purpose of connectivity maintenance. The first solution directly addresses the discontinuity issue in the naive solution. This is done by adjusting conservatively the discontinuous term discussed in Section 3.2. Our second solution refines the first to reduce its conservatism. For clarity of exposition, here we just explain the proposed solutions, and delay the formal technical analysis to Section 3.4 below.

We first design a connectivity maintenance constraint map by replacing the discontinuous term μ_m. The discontinuity in μ_m is due to the abrupt change in the eigenspace being considered in the minimization (3.6). One possible fix is to augment the eigenspace preemptively so that there is no abrupt expansion. For $\mathcal{I} \subseteq [n]$, consider

$$\mathcal{V}_{\mathcal{I}}(\mathbf{x}) := \operatorname{span}\Big\{ \bigcup_{p\in\mathcal{I}} \mathcal{V}_p(\mathbf{x}) \Big\} \cap \mathbb{S}^n,$$

the normalized span of eigenspaces corresponding to the eigenvalues $\{\lambda_p\}_{p\in\mathcal{I}}$ at $\mathbf{x}$. We refer to the set-valued map $\mathcal{V}_{\mathcal{I}}$ as the *normalized merged eigenspace*. We use this set-valued map to define

$$\mu_{\mathcal{I}}(\mathbf{x},\mathbf{u}) := \min_{\mathbf{v}\in\mathcal{V}_{\mathcal{I}}(\mathbf{x})} \mathbf{v}^\top \Big(\sum_{i\in[N]} \frac{\partial \mathbf{L}}{\partial \mathbf{x}_i}\mathbf{u}_i \Big)\mathbf{v}, \tag{3.7}$$

which we refer to as the *merged lower bound (of the eigenvalues' rate of change)* as it bounds the rate of change of all the eigenvalues $\{\lambda_p\}_{p\in\mathcal{I}}$ at $\mathbf{x}$ for a given $\mathbf{u}$.

We are interested in using the merged lower bound to replace the discontinuous function μ_2 used in $\mathcal{U}_{\text{cm}}$, in order to avoid sudden changes in its value. For instance, noticing how the eigenspace

$\mathcal{V}_2$ expands into $\mathcal{V}_{[2:3]}$ when $\lambda_2 = \lambda_3$, we want to replace μ_2 with $\mu_{[2:3]}$. This way, we avoid the abrupt change in the connectivity maintenance constraint map that occurs when $\lambda_2 = \lambda_3$. However, with this approach, a discontinuity would still arise when $\lambda_3 = \lambda_4$ since the eigenspace of λ_4 is not considered in the merged eigenspace. To address this, we can indeed use $\mu_{[2:n]}$, corresponding to the merged eigenspace of all nonzero eigenvalues, as stated in the following result.

Theorem 3.3.1. ***(Strict Connectivity Constraint Map for Continuous Controller).*** *Consider the multi-robot system* (3.1)*. Given a locally Lipschitz extended class $\mathcal{K}$ function α, define the constraint map*

$$\mathcal{U}_{\text{str}}(\mathbf{x}) := \left\{\mathbf{u} \in \mathbb{R}^N \mid \mu_{[2:n]}(\mathbf{x}, \mathbf{u}) \geq -\alpha(\lambda_2(\mathbf{x}) - \varepsilon)\right\}. \tag{3.8}$$

If, for each $\mathbf{x}$, there exists a control input $\mathbf{u} \in \mathbb{R}^N$ that strictly satisfies the constraint map $\mathcal{U}_{\text{str}} \cap \mathcal{U}_{\text{nom}}(\mathbf{x})$, then under Assumptions 3.1.1 and 3.1.2, the optimization-based controller

$$\mathbf{k}_{\text{str}}(\mathbf{x}) := \underset{\mathbf{u} \in \mathcal{U}_{\text{str}} \cap \mathcal{U}_{\text{nom}}(\mathbf{x})}{\operatorname{argmin}} \; J(\mathbf{x}, \mathbf{u}) \tag{3.9}$$

is continuous on $\mathcal{C}_\varepsilon$, and the closed-loop feedback $\mathbf{u} = \mathbf{k}_{\text{str}}(\mathbf{x})$ renders $\lambda_2(\mathbf{x}(t)) \geq \varepsilon$ at all time, ensuring that network connectivity is maintained, for any given initial condition $\mathbf{x}_0 \in \mathcal{C}_\varepsilon$.

While Theorem 3.3.1 provides a solution to Problem 1, it is undoubtedly conservative. By design, the constraint map $\mathcal{U}_{\text{str}}$ bounds the rate of change of λ_2 as if it always has the highest possible multiplicity of $n - 1$ for a connected robot configuration. As a result, in the situation when the multiplicity of λ_2 is unlikely to change, e.g., when λ_2 is far apart from λ_3, the design is conservative. This conservatism is also illustrated later in our simulations of Section 3.5.

To be less conservative, our next design takes into account how far the multiplicity of the

eigenvalues is from changing. Instead of defining a NCBF constraint map for only λ_2, the design considers NCBFs for all the nonzero eigenvalues. We then replace each μ_m with the merged lower bound $\mu_{[2:m]}$. Formally, for each $m \in [2 : n]$, consider the constraint maps,

$$\mathcal{U}_{[2:m]}(\mathbf{x}) := \left\{\mathbf{u} \in \mathbb{R}^N \mid \mu_{[2:m]}(\mathbf{x}, \mathbf{u}) \geq -\alpha(\lambda_m(\mathbf{x}) - \varepsilon)\right\}$$

with a locally Lipschitz extended class $\mathcal{K}$ function α and a constant $\varepsilon \in \mathbb{R}_{>0}$. The aggregations of the constraint maps of this form gives rise to our design for connectivity maintenance.

Theorem 3.3.2. ***(Aggregate Connectivity Constraint Map for Continuous Controller).*** *Consider the multi-robot system* (3.1)*. Given a locally Lipschitz extended class $\mathcal{K}$ function α, define the constraint map*

$$\mathcal{U}_{\text{agg}}(\mathbf{x}) := \bigcap_{m \in [2:n]} \mathcal{U}_{[2:m]}(\mathbf{x}). \tag{3.10}$$

If, for each $\mathbf{x}$, there exists a control input $\mathbf{u} \in \mathbb{R}^N$ that strictly satisfies $\mathcal{U}_{\text{agg}} \cap \mathcal{U}_{\text{nom}}(\mathbf{x})$, then under Assumptions 3.1.1 and 3.1.2, the optimization-based controller

$$\mathbf{k}_{\text{agg}}(\mathbf{x}) := \underset{\mathbf{u} \in \mathcal{U}_{\text{agg}}(\mathbf{x}) \cap \mathcal{U}_{\text{nom}}(\mathbf{x})}{\operatorname{argmin}} J(\mathbf{x}, \mathbf{u}) \tag{3.11}$$

is continuous on $\mathcal{C}_\varepsilon$, and the closed-loop feedback $\mathbf{u} = \mathbf{k}_{\text{agg}}(\mathbf{x})$ renders $\lambda_2(\mathbf{x}(t)) \geq \varepsilon$ at all time, ensuring that network connectivity is maintained, for any given initial condition $\mathbf{x}_0 \in \mathcal{C}_\varepsilon$.

The idea behind the design of the aggregate constraint (3.10) is as follows. Consider a state $\mathbf{x}$ where $\lambda_{m-1}(\mathbf{x}) = \lambda_m(\mathbf{x})$. At this state, $\mathcal{U}_{[2:m-1](\mathbf{x})}$ abruptly shrinks to $\mathcal{U}_{[2:m-1](\mathbf{x})}$ due to the value of the merged lower bound $\mu_{[2:m-1]}(\mathbf{x}, \mathbf{u})$ dropping to that of $\mu_{[2:m]}(\mathbf{x}, \mathbf{u})$, for any given $\mathbf{u}$. Nevertheless,

the constraint map $\mathcal{U}_{[2:m]}$ is also considered in the aggregate constraint map $\mathcal{U}_{\text{agg}}$, and the fact that it experiences no abrupt change there is enough to prevent $\mathcal{U}_{\text{agg}}$ from changing abruptly at that state.

Both constraint maps (3.8) and (3.10) ensure continuity of the corresponding optimization-based controller and solve Problem 1. In general, for $m \in [2 : n]$, one has $\mathcal{U}_{\text{str}} \subseteq \mathcal{U}_{[2:m]}$ because $\mu_{[2:n]} \leq \mu_{[2:m]}$ and $\lambda_m \geq \lambda_2$. Therefore, $\mathcal{U}_{\text{str}} \subseteq \mathcal{U}_{\text{agg}}$, with equality holding on those states where $\lambda_n(\mathbf{x}) = \lambda_2(\mathbf{x})$. Consequently, $\mathcal{U}_{\text{agg}}$ imposes less conservative constraints than $\mathcal{U}_{\text{str}}$. This is because the aggregate constraint $\mathcal{U}_{\text{agg}}$ only gradually becomes stricter as the gap between each eigenvalue and the lowest $\lambda_m - \lambda_2$ gets smaller, unlike the strict constraint $\mathcal{U}_{\text{str}}$ that is agnostic to the gap.

Remark 3.3.3. ***(Strictly Satisfying Feasible Controls Requirement).*** We note that both Theorems 3.3.1 and 3.3.2 require the existence, at each $\mathbf{x}$, of a control $\mathbf{u}$ strictly satisfying the corresponding constraint map. This is our conceptualization of the fact that, in order for Problem 1 to be solvable, there must exist at each state a control that can simultaneously maintain connectivity and satisfy the nominal constraints. The choice of class $\mathcal{K}$ function also provides flexibility in this regard because, if a control exists that satisfies the constraints at $\mathbf{x}$ for α_1, then the same control strictly satisfies the constraints for α_2 with $\alpha_1 < \alpha_2$, as long as $\lambda_2(\mathbf{x}) \neq \varepsilon$. Finally, as we show later in our analysis (cf. Lemma 3.4.4), the existence of strictly satisfying feasible control at each state is enough to guarantee the existence of a continuous controller. While this latter condition would be enough to establish Theorems 3.3.1 and 3.3.2, the existence of strictly satisfying feasible control is easier to check as it consists of a pointwise condition at each network state $\mathbf{x}$, instead of the analysis across the states required to ensure continuity. •

Remark 3.3.4. ***(Computation of Proposed Controllers).*** For each $\mathbf{x}$, the computation of the controllers $\mathbf{k}_{\text{str}}$ and $\mathbf{k}_{\text{agg}}$ are convex optimization problems (as we show later, the constraint maps are

convex-valued, and the cost function J is convex by assumption). This means that one can utilize the wide variety of existing methods and solvers available for convex optimization, cf. [BV09, Roc70], to compute the controllers. In implementing these methods, one must pay attention to the fact that obtaining the value of each merged lower bound function $\mu_{[2:m]}$ is itself an optimization problem. Nevertheless, this can be addressed by casting the computation of the merged lower bounds as an eigenvalue problem. To see why this is so, note the following relationship

$$\begin{aligned}
\mu_{[2:m]}(\mathbf{x},\mathbf{u}) &= \min_{\mathbf{v}\in\mathcal{V}_{[2:m]}(\mathbf{x})} \mathbf{v}^\top \Big(\sum_{i\in[N]} \frac{\partial \mathbf{L}}{\partial \mathbf{x}_i}\mathbf{u}_i \Big)\mathbf{v} \\
&= \min_{\xi\in\mathcal{S}^{m-1}} \xi^\top [\mathbf{v}]_{2:m}^\top(\mathbf{x}) \Big(\sum_{i\in[N]} \frac{\partial \mathbf{L}}{\partial \mathbf{x}_i}\mathbf{u}_i \Big)[\mathbf{v}]_{2:m}(\mathbf{x})\xi \\
&:= \min_{\xi\in\mathbb{S}^{m-1}} \xi^\top Z_m(\mathbf{x},\mathbf{u})\xi,
\end{aligned}$$

where $[\mathbf{v}]_{2:m}(\mathbf{x})$ is the matrix created by concatenating orthonormal eigenvectors of $\{\lambda_p\}_{p\in[2:m]}$. It then follows that $\mu_{[2:m]}(\mathbf{x},\mathbf{u})$ is the minimum eigenvalue of the matrix $Z_m(\mathbf{x},\mathbf{u})$ defined above. This formulation as eigenvalue problem is advantageous for two reasons: it makes the evaluation of the function easy using standard linear algebraic routines and, for gradient-based optimization methods, it facilitates the computation of the generalized gradient of the merged lower bound. •

3.4 Technical Analysis of the Proposed Solutions

This section provides the proofs of the results presented in Section 3.3. Before presenting them, we establish a number of auxiliary results that characterize the properties of the merged lower bounds involved in the construction of the constraint set-valued maps.

3.4.1 Properties of Merged Lower Bounds

We first examine the properties of functions $\mu_{\mathcal{I}}$ of the form (3.7) defining our proposed constraint sets. The definition of such functions relies critically on the normalized merged eigenspace $\mathcal{V}_{\mathcal{I}}$. The following result characterizes the continuity properties of the latter.

Theorem 3.4.1. ***(Continuity of Normalized Merged Eigenspaces).*** *Let* $\mathbf{L} : \mathbb{R}^N \to \mathrm{Sym}_n$ *be a continuous function. Given* $\mathcal{I} \subset [n]$*, the normalized merged eigenspace* $\mathcal{V}_{\mathcal{I}}$ *is continuous at any* $\mathbf{x}$ *such that* $\lambda_i(\mathbf{x}) \neq \lambda_j(\mathbf{x})$ *for all* $i \in \mathcal{I}$ *and* $j \notin \mathcal{I}$*, i.e., where none of the considered eigenvalues is equal to any of the remaining eigenvalues.*

Due to its length, the proof of this result is provided in the Appendix. Building on this result, the continuity of the merged lower bounds follows from a direct application of the Berge Maximum Theorem [AB99, Thm. 17.31].

Corollary 3.4.2. ***(Continuity of Merged Lower Bounds).*** *Given* $\mathcal{I} \subset [n]$*, the function* $\mu_{\mathcal{I}}$ *is continuous at any* $(\mathbf{x}, \mathbf{u})$ *such that* $\lambda_i(\mathbf{x}) \neq \lambda_j(\mathbf{x})$ *for all* $i \in \mathcal{I}$ *and* $j \notin \mathcal{I}$*.* □

In particular, we consider indices $\mathcal{I} = [2 : m]$ of ordered eigenvalues on the domain where the graph remains connected $\mathcal{C}_\varepsilon$ (i.e., where $\lambda_1(\mathbf{x}) \neq \lambda_2(\mathbf{x})$). Thus, $\mu_{[2:m]}$ is continuous at any $\mathbf{x}$ such that $\lambda_m(\mathbf{x}) \neq \lambda_{m+1}(\mathbf{x})$, and $\mu_{[2:n]}$ is continuous everywhere on $\mathcal{C}_\varepsilon \times \mathbb{R}^N$.

Besides continuity of $\mu_{[2:m]}$, another crucial property to show is convexity of the constraint maps $\mathcal{U}_{\mathrm{str}}$ and $\mathcal{U}_{\mathrm{agg}}$. To this end, we establish the concavity property of the merged lower bounds.

Lemma 3.4.3. ***(Concavity of Merged Lower Bounds).*** *For any* $\mathcal{I} \subseteq [n]$*,* $\mu_{\mathcal{I}}$ *is concave in* $\mathbf{u}$*. Consequently, the constraint maps* $\mathcal{U}_{\mathrm{str}}$ *and* $\mathcal{U}_{\mathrm{agg}}$ *are convex-valued.*

Proof. Given any $\mathbf{u}^1, \mathbf{u}^2 \in \mathbb{R}^N$ and $0 \leq \gamma \leq 1$, we have

$$\begin{aligned}
&\mu_{\mathcal{I}}(\mathbf{x}, \gamma\mathbf{u}^1 + (1-\gamma)\mathbf{u}^2) \\
&\quad = \min_{\mathbf{v} \in \mathcal{V}_{\mathcal{I}}(x)} \mathbf{v}^\top \Big(\sum_{i \in [N]} \frac{\partial \mathbf{L}}{\partial \mathbf{x}_i} (\gamma \mathbf{u}_i^1 + (1-\gamma)\mathbf{u}_i^2) \Big) \mathbf{v} \\
&\quad \geq \min_{\mathbf{v} \in \mathcal{V}_{\mathcal{I}}(\mathbf{x})} \Big(\gamma \mathbf{v}^\top \Big(\sum_{i \in [N]} \frac{\partial \mathbf{L}}{\partial \mathbf{x}_i} \mathbf{u}_i^1 \Big) \mathbf{v} \Big) \\
&\qquad + \min_{\mathbf{v} \in \mathcal{V}_{\mathcal{I}}(\mathbf{x})} \Big((1-\gamma) \mathbf{v}^\top \Big(\sum_{i \in [N]} \frac{\partial \mathbf{L}}{\partial \mathbf{x}_i} \mathbf{u}_i^2 \Big) \mathbf{v} \Big) \\
&\quad = \gamma \mu_{\mathcal{I}}(\mathbf{x}, \mathbf{u}^1) + (1-\gamma) \mu_{\mathcal{I}}(\mathbf{x}, \mathbf{u}^2).
\end{aligned}$$

Therefore, $\mu_{\mathcal{I}}$ is concave in $\mathbf{u}$. □

Having established the continuity and concavity properties of the merged lower bounds $\mu_{\mathcal{I}}$, we next turn our attention to characterize the properties of the constraint maps.

3.4.2 Equivalent Constraint Maps

In general, the constraint maps $\mathcal{U}_{\text{str}}$ and $\mathcal{U}_{\text{agg}}$ might not be UHC because they are unbounded. To make sure the requirements of Lemma 2.4.2 as well as Berge Maximum Theorem are met, we explain here how to consider, following Section 3.1, equivalent constraint maps that are compact-valued. This procedure involves using sublevel sets of the cost function J, which are compact due to Assumption 3.1.2. In order to do so, we require a feasible control function $\mathbf{x} \mapsto \bar{\mathbf{u}}(\mathbf{x})$ to define $\mathcal{J}_{\bar{\mathbf{u}}}$ as in (3.3). Note, importantly for our purposes, that the function $\bar{\mathbf{u}}$ must be continuous so that $\mathcal{J}_{\bar{\mathbf{u}}}$ is also continuous. The next result shows that, under the assumptions of Theorems 3.3.1 and 3.3.2, such continuous feasible control function always exists.

Lemma 3.4.4. ***(Generalization of Artstein's Theorem).*** *Consider a set-valued map* $\mathcal{U} : \mathbb{R}^N \rightrightarrows \mathbb{R}^M$ *defined with a vector-valued function* $g : \mathbb{R}^N \to \mathbb{R}^M$ *as*

$$\mathcal{U}(\mathbf{x}) = \{\mathbf{u} \in \mathbb{R}^M \mid g(\mathbf{x}, \mathbf{u}) \leq 0\}.$$

If g *is continuous and* $\mathcal{U}$ *is convex-valued, and, for each* $\mathbf{x}$*, there exists a control input* $\mathbf{u}$ *that strictly satisfies* $\mathcal{U}(\mathbf{x})$*, then there exists a* $\mathcal{C}^\infty$ *function* $\bar{\mathbf{u}} : \mathbb{R}^N \to \mathbb{R}^M$ *such that* $\bar{\mathbf{u}}(\mathbf{x}) \in \mathcal{U}(\mathbf{x})$*.*

Proof. For each $\mathbf{x}$, let $\mathbf{u}_{\text{int}}(\mathbf{x})$ denote the control input such that $g(\mathbf{x}, \mathbf{u}_{\text{int}}(\mathbf{x})) > 0$. Due to continuity of g, there exists a neighborhood of $\mathbf{x}$, denoted by $\mathcal{W}(\mathbf{x})$, such that $\mathbf{u}_{\text{int}}(\mathbf{x}) \in \mathcal{U}(\mathbf{x})$. The collection of $\{\mathcal{W}(\mathbf{x})\}_{\mathbf{x}\in\mathbb{R}^N}$ is an open cover for $\mathbb{R}^N$. Then, because we deal with a Euclidean space that is a differentiable manifold, there exists a countable partition of unity $\{\psi_j\}$ subordinate to the cover, cf. [War89a, Theorem 1.11]. In other words, for each j, there exists an $\mathbf{x}$ such that $\text{supp}(\psi_j)$ is a subset of $\mathcal{W}(\mathbf{x})$, each of which has an associated control $\mathbf{u}^j_{\text{int}} \in \mathcal{U}(\mathbf{x})$ for $\mathbf{x} \in \text{supp}(\psi_j)$. Then we define $\bar{\mathbf{u}}(\mathbf{x}) = \sum_j \psi_j(\mathbf{x})\mathbf{u}^j_{\text{int}}$, which satisfies the statement due to convexity of the map $\mathcal{U}$. □

Lemma 3.4.4 is a generalization of Artstein's Theorem [Art83, Thm. 4.1] on the existence of a continuous controller given a control Lyapunov function. The proof of the result, included here for completeness, is also a slight modification of the original proof. For each of the cases $\mathcal{U} = \mathcal{U}_{\text{str}}$ and $\mathcal{U} = \mathcal{U}_{\text{agg}}$, Lemma 3.4.4 provides a continuous feasible control function $\bar{\mathbf{u}}$, which we use to define the corresponding set-valued map $\mathcal{J}_{\bar{\mathbf{u}}}$. This map is convex-valued and compact-valued due to it being a sublevel set of a strictly convex function J, cf. Assumption 3.1.2. Then according to Lemmas 2.4.2 and 2.4.3, it is also continuous due to the functions $\bar{\mathbf{u}}$ and J being continuous, cf. Assumption 3.1.1. We then consider the intersections $\mathcal{U}_{\text{str}} \cap \mathcal{U}_{\text{nom}} \cap \mathcal{J}_{\bar{\mathbf{u}}}$ and $\mathcal{U}_{\text{agg}} \cap \mathcal{U}_{\text{nom}} \cap \mathcal{J}_{\bar{\mathbf{u}}}$, where

the inclusion of $\mathcal{J}_{\bar{\mathbf{u}}}$ make these constraint maps compact-valued. For the purpose of our analysis, we equivalently define $\mathbf{k}_{\text{str}}$ and $\mathbf{k}_{\text{agg}}$ with these constraint maps as the constraint to the optimization.

3.4.3 Continuity of the Connectivity Maintenance Controllers

With the preparations from prior sections, we are now ready to prove our results on continuity of $\mathbf{k}_{\text{str}}$ and $\mathbf{k}_{\text{agg}}$.

Proof of Theorem 3.3.1. Consider the constraint set $\mathcal{U}_{\text{str}} \cap \mathcal{U}_{\text{nom}} \cap \mathcal{J}_{\bar{\mathbf{u}}}$. We note the followings: (i) all the functions defining the constraint map are continuous due to Assumption 3.1.1 and $\mu_{[2:n]}$ being continuous everywhere (on $\mathcal{C}_\varepsilon \times \mathbb{R}^N$); (ii) the map is convex-valued because all intersecting maps are convex-valued; (iii) the map has a nonempty interior by assumption; (iv) the map is compact-valued because the intersecting maps are closed-valued and $\mathcal{J}_{\mathbf{u}}$ is compact-valued. Thus, we may apply Lemmas 2.4.2 and 2.4.3, to show continuity of this constraint map. By Berge Maximum Theorem [AB99, Thm. 17.31], $\mathbf{k}_{\text{str}}$ is a continuous function as stated. Lastly, from the relationship

$$\min \mathcal{L}_{f_{\text{si}}} \lambda_2(\mathbf{x}, \mathbf{u}) = \mu_2(\mathbf{x}, \mathbf{u}) \geq \mu_{[2:n]}(\mathbf{x}, \mathbf{u}),$$

it follows that $\mathbf{k}_{\text{str}}(\mathbf{x}) \in \mathcal{U}_{\text{str}}(\mathbf{x}) \subseteq \mathcal{U}_{\text{cm}}(\mathbf{x})$. As a result, Lemma 3.2.1 ensures $\lambda_2(\mathbf{x}(t)) \geq \varepsilon$, and the proof concludes. □

We next prove the continuity result for $\mathbf{k}_{\text{agg}}$, which is more complicated due to the merged lower bounds used not being continuous everywhere.

Proof of Theorem 3.3.2. Consider the constraint map $\mathcal{U}_{\text{agg}} \cap \mathcal{U}_{\text{nom}} \cap \mathcal{J}_{\bar{u}}$. Because each $\mu_{[2:m]}$ is not continuous everywhere, we can only conclude continuity using Lemmas 2.4.2 and 2.4.3 wherever

$\mu_{[2:m]}$ are continuous for all $m \in [2 : n]$. For the remaining states, we show continuity of the constraint map by proving separately below that it is UHC and LHC. Note that once we prove continuity, the theorem statements are established analogously as we did in the proof of Theorem 3.3.1

Upper Hemicontinuity: We begin by consider the partial constraint map $\mathcal{J}_{\bar{\mathbf{u}}} \cap \mathcal{U}_{[2:n]}$. This set-valued map is continuous on $\mathcal{C}_\varepsilon$ because of the continuity of $\mu_{[2:n]}$. Consider its intersection with $\mathcal{J}_{\bar{\mathbf{u}}} \cap \mathcal{U}_{[2:n-1]}$. At the states where $\lambda_n(\mathbf{x}) = \lambda_{n-1}(\mathbf{x})$, notice that $\mathcal{U}_{[2:n]}(\mathbf{x}) = \mathcal{U}_{[2:n-1]}(\mathbf{x})$, so the intersection $\mathcal{J}_{\bar{\mathbf{u}}} \cap (\bigcap_{m\in[n-1:n]} \mathcal{U}_{[2:m]})(\mathbf{x})$ is exactly the same set as $\mathcal{J}_{\bar{\mathbf{u}}} \cap \mathcal{U}_{[2:n]}(\mathbf{x})$ at those $\mathbf{x}$. For other states $\mathbf{x}$, we know that the former map is a subset of the latter. Then, directly from the definition of UHC for $\mathcal{J}_{\bar{\mathbf{u}}} \cap \mathcal{U}_{[2:n]}$, we can conclude UHC for the intersection $\mathcal{J}_{\bar{\mathbf{u}}} \cap (\bigcap_{m\in[n-1:n]} \mathcal{U}_{[2:m]})$ at $\mathbf{x}$ where $\lambda_n(\mathbf{x}) = \lambda_{n-1}(\mathbf{x})$. Elsewhere, the intersection can be proven UHC directly via Lemma 2.4.2, so it is continuous everywhere on $\mathcal{C}_\varepsilon$. With the same line of reasoning, we can continue to show by induction that $\mathcal{J}_{\bar{u}} \cap \mathcal{U}_{\text{agg}}$ is UHC on $\mathcal{C}_\varepsilon$. Then intersecting with $\mathcal{U}_{\text{nom}}$, we conclude the set-valued map $\mathcal{U}_{\text{agg}} \cap \mathcal{U}_{\text{nom}} \cap \mathcal{J}_{\bar{u}}$ is UHC from Lemma 2.4.4.

Lower Hemicontinuity: We begin by defining the following auxiliary set-valued maps for $m \in [2 : n]$,

$$\mathcal{H}_m(\mathbf{x}) = \left\{\mathbf{u} \in \mathbb{R}^n \mid \mu_{[2:m]}(\mathbf{x}, \mathbf{u}) \geq -\alpha(\lambda_{m-1}(\mathbf{x}) - \varepsilon)\right\}.$$

By definition, $\mathcal{H}_m(\mathbf{x}) \subseteq \mathcal{U}_{[2:m-1]}(\mathbf{x})$ because $\mu_{[2:m]}(\mathbf{x}, \mathbf{u}) \leq \mu_{[2:m-1]}(\mathbf{x}, \mathbf{u})$, and $\mathcal{H}_m(\mathbf{x}) \subseteq \mathcal{U}_{[2:m]}(\mathbf{x})$ because $\lambda_{m-1}(\mathbf{x}) \leq \lambda_m(\mathbf{x})$. In addition, note that $\mathcal{H}_m$ is convex-valued because the merged lower bound $\mu_{[2:m]}$ in concave in $\mathbf{u}$, cf. Lemma 3.4.3, and it has a nonempty interior as it is a subset of $\mathcal{U}_{[2:m]}$, which has a nonempty interior by assumption. Then, by Lemma 2.4.3 it is LHC for all $\mathbf{x} \in \mathcal{C}_\varepsilon$ where $\lambda_m(\mathbf{x}) \neq \lambda_{m+1}(\mathbf{x})$ (with $\mathcal{H}_n$ continuous everywhere on $\mathcal{C}_\varepsilon$).

We prove LHC of $\mathcal{U}_{\text{agg}}$ by induction. We start by considering the maps $\mathcal{U}_{[2:n]}$ and $\mathcal{H}_n$, which

are both LHC on $\mathcal{C}_\varepsilon$. We then consider the intersection $\mathcal{U}_{[2:n]}$ with $\mathcal{U}_{[2:n-1]}$. For $\mathbf{x}$ where $\lambda_n(\mathbf{x}) = \lambda_{n-1}(\mathbf{x})$, the two eigenvalues share the same eigenspaces. Thus, it is also the case that $\mu_{[2:n]}(\mathbf{x}) = \mu_{[2:n-1]}(\mathbf{x})$, and we find that $\mathcal{H}_n(\mathbf{x}) = \mathcal{U}_{[2:n]}(\mathbf{x}) \cap \mathcal{U}_{[2:n-1]}(\mathbf{x})$ for all $\mathbf{x}$ where the two eigenvalues are equal. From this and the fact that $\mathcal{H}_n$ is a subset of $\mathcal{U}_{[2:n]} \cap \mathcal{U}_{[2:n-1]}$ in general, we can use the LHC of $\mathcal{H}_n$, at $\mathbf{x}$ where $\lambda_n(\mathbf{x}) = \lambda_{n-1}(\mathbf{x})$ to deduce LHC for $\mathcal{U}_{[2:n]} \cap \mathcal{U}_{[2:n-1]}$ there. Elsewhere, the set $\mathcal{U}_{[2:n]} \cap \mathcal{U}_{[2:n-1]}$ can be proven continuous directly from Lemma 2.4.3, so it is LHC everywhere on $\mathcal{C}_\varepsilon$. Then using Lemma 2.4.5, we also deduce that the intersection $\mathcal{H}_n \cap (\mathcal{U}_{[2:n]} \cap \mathcal{U}_{[2:n-1]})$ is LHC on $\mathcal{C}_\varepsilon$.

To continue with the induction proof, assume the set-valued maps

$$\bigcap_{m \le p \le n} \mathcal{U}_{[2:p]} \text{ and } \mathcal{H}_m \cap \bigcap_{m \le p \le n} \mathcal{U}_{[2:p]}$$

are LHC. Then we can follow the arguments above to also deduce that their intersections with $\mathcal{U}_{[2:m-1]}$ are also LHC. Hence, $\mathcal{U}_{\text{agg}}$ is LHC. Then the LHC of the intersection $\mathcal{U}_{\text{agg}} \cap \mathcal{U}_{\text{nom}} \cap \mathcal{J}_{\bar{\mathbf{u}}}$ follows via Lemma 2.4.5, concluding the proof. □

3.5 Simulations and Experimental Validation

In this section we report the simulations and the experiment we have carried out to verify the effectiveness of the proposed controller. We consider a resource gathering problem with a group of four ($n = 4$) robots, moving in a two-dimensional space ($d_r = 2$ for all agents). Each robot is tasked with visiting its own target region. If the robots were to individually move directly to their targets, the network will be disconnected. Therefore, we prioritize the order in which the robots

reach their targets and use our proposed controller to maintain the connectivity among them. We consider the mission accomplished when the target location is visited by the corresponding robot, and we change the task prioritization to the next robot.

The nominal controller carries each robot towards the corresponding target with a conical potential field:

$$u_{\text{nom},r}(x_r) = v_{\text{nom}} \frac{e_r(x_r)}{\|e_r(x_r)\|}, \ \forall \in [n], \tag{3.12}$$

where x_r is the position of the r-th robot and $e_r(x_r) = x_{\text{target},r} - x_r$ is the error between the center of the robot's target region and its position, and $v_{\text{nom}} \in \mathbb{R}_{>0}$ is a constant velocity parameter. By denoting $\mathbf{k}_{\text{nom}}(\mathbf{x}) = [u_{\text{nom},1}(x_1)^\top, \dots, u_{\text{nom},4}(x_4)^\top]^\top$, our cost function

$$J(\mathbf{x}, \mathbf{u}) = \|\mathbf{u} - \mathbf{k}_{\text{nom}}(\mathbf{x})\|^2 \tag{3.13}$$

measures the deviation of the control decision from the nominal controller. In order to ensure that our prioritized robot, indexed P, makes progress towards its target, we enforce the following constraint map,

$$\mathcal{U}_{\text{nom}}(\mathbf{x}) = \left\{\mathbf{u} \in \mathbb{R}^N \mid k v_{\text{nom}}^2 - u_{\text{nom},P}^\top u_P \leq 0\right\}, \tag{3.14}$$

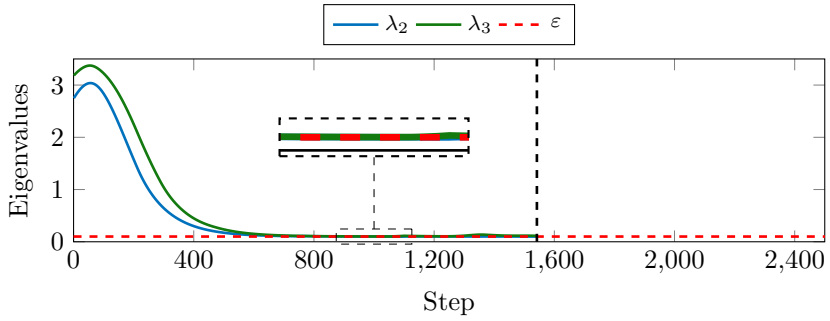
where $k \in \mathbb{R}_{>0}$ is a constant parameter to restrict how much u_P should point in the direction of $u_{\text{nom},P}$. Once the robot reaches its assigned target region, its $u_{\text{nom},P}$ is set equal to zero. This represents the fact that after having accomplished its task, the robot is relieved from its mission, and prefers to conserve energy by not moving. Note that it continues to collaborate at maintaining the connectivity. Also after the prioritized robot achieves its mission, we adjust $\mathcal{U}_{\text{nom}}$ by changing the index to correspond to the next robot that has yet to achieve its goal.

Note that the objective (3.13) and the nominal constraint (3.14) verify both Assumptions 3.1.1 and 3.1.2 for any priority robot (we disregard the jumps in $\mathcal{U}_{\text{nom}}$ due to the transitions when a robot reaches its target region and the identity of the priority robot changes). We show that our proposed controllers from Theorem 3.3.1 and 3.3.2 are continuous for the duration between events when the prioritized robots achieve its goal. For both our simulations and our experiment, we use projected saddle-point dynamics [CGC17] to solve the convex optimization problems and compute our controllers in MATLAB®.

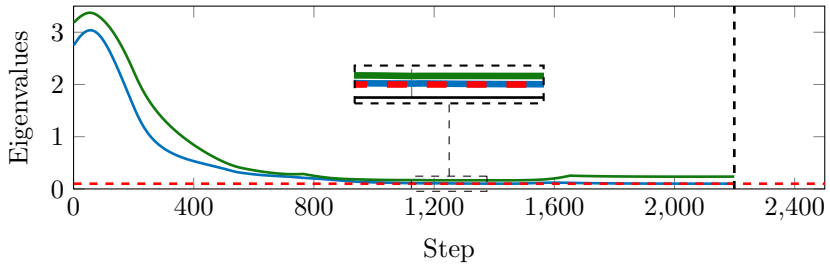
3.5.1 Simulations

Our simulations highlight the differences among the different controllers: $\mathbf{k}_{\text{dis}}$, defined in (3.4), $\mathbf{k}_{\text{str}}$, defined in (3.9), and $\mathbf{k}_{\text{agg}}$, defined in (3.11). The initial positions, the robots' targets, and the parameters ($v_{\text{nom}} = 0.5$, $k = 0.75$, $\varepsilon = 0.1$) are the same in each simulation. Fig. 3.1 reports the eigenvalues of the Laplacian matrix during the simulations. It is clear how both the aggregate (Fig. 3.1a) and the strict controller (Fig. 3.1b) maintain the connectivity constraint, unlike the discontinuous one that leads to disconnection (Fig. 3.1c). Regarding overall performance, the aggregate controller (1542 steps) outperforms the strict one (2199 steps). This corroborates the hypothesis that the strict controller over-constrains the robots' motion, hence resulting in a worse performance in terms of the total time it takes for the network to complete its goals. Figs. 3.2a and 3.2b show the continuous input produced by the aggregate and the strict controllers, and Fig. 3.2c shows the discontinuous one generated by the discontinuous controller. Fig. 3.3 reports the evolution of the function defining the nominal constraint map $\mathcal{U}_{\text{nom}}$ under the strict and the aggregate controllers. In the corresponding slot of time, the robot that has the target with the highest priority

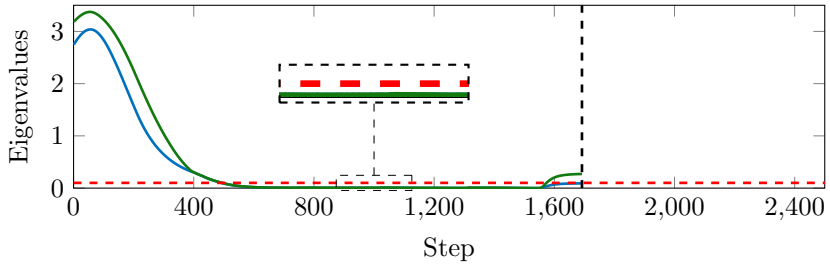
respects the constraint, while the others cooperate to maintain connectivity, minimally changing their nominal control law. We do not report the plot for the discontinuous controller as it is highly jittering, confirming what is already displayed in Fig. 3.2c.



(a) Aggregate controller

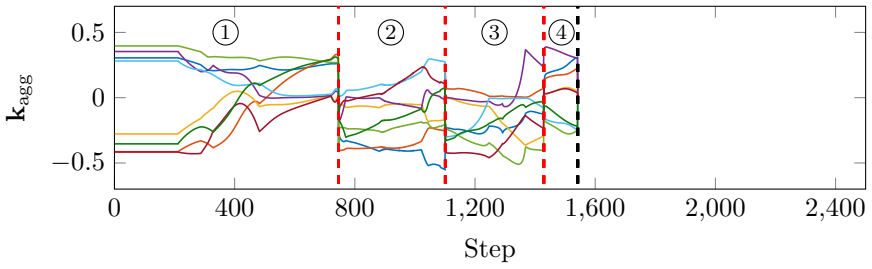


(b) Strict controller

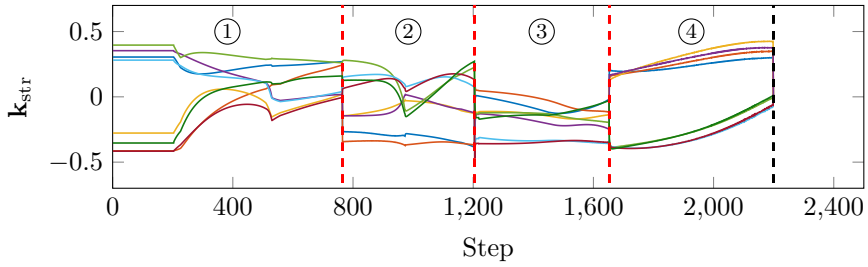


(c) Discontinuous controller

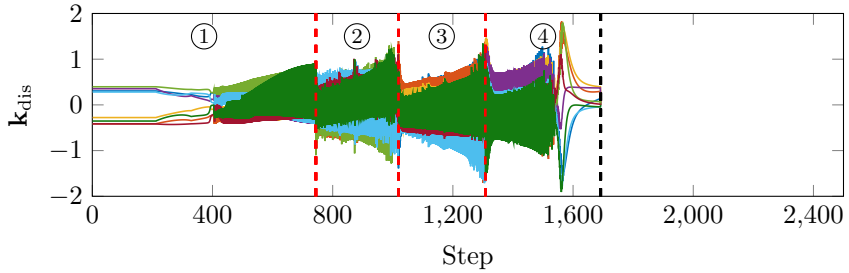
Figure 3.1: Eigenvalue evolution during the simulations under the different controllers. The *dashed black* lines represent the end of the network task.



(a) Aggregate controller

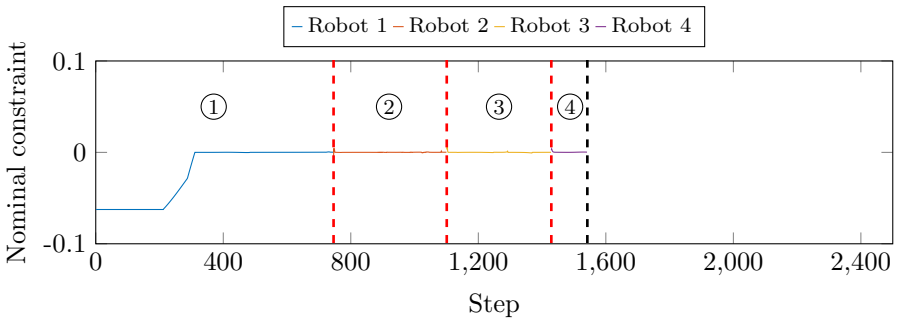


(b) Strict controller

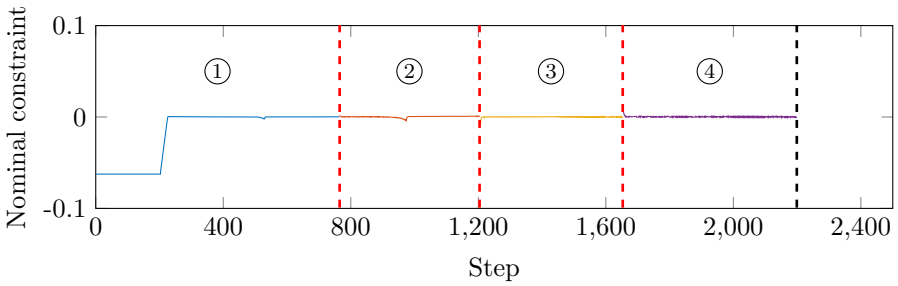


(c) Discontinuous controller

Figure 3.2: Control inputs during the simulations of different controllers. We report all the components of the control input for each robot.



(a) Aggregate controller



(b) Strict controller

Figure 3.3: Nominal constraint (3.14) during the simulations. The *dashed red* lines represent the instant in which the robot priority changes, due to the fact that a target has been reached. The number for each time slot corresponds to the robot with the highest priority.

3.5.2 Experimental Validation

Figure 3.4: Experimental setup: 4 ePucks and their corresponding targets.

We also carry out an experiment for the same resource gathering problem, cf Fig. 3.4. We use four small wheeled robots (ePucks) that are controlled via Bluetooth from a central unit that performs the calculations. The central unit is also connected to an Optitrack system, which provides the position of the robots. In order to transform the input calculated for the single-integrator dynamics to the unicycle dynamics of the robots, we use a simple input-output state-feedback linearization [OLV02]. We tested only the proposed controller $\mathbf{k}_{\text{agg}}$, as the simulations in Section 3.5.1 verified that it is the best both in terms of performance and connectivity maintenance. We set the main parameters as $v_{\text{nom}} = 0.1$, $k = 0.75$, and $\varepsilon = 0.3$. We report an example of the experiments in an accompanying video.

Fig. 3.5 reports the eigenvalue evolution during the experiment, further confirming the effectiveness of the proposed method in maintaining connectivity. Fig. 3.6 shows the trajectories followed by the robots, accomplishing the gathering task. It is evident how each target had been reached by the corresponding robot, and in the final positions (reported with *triangles*) it is possible

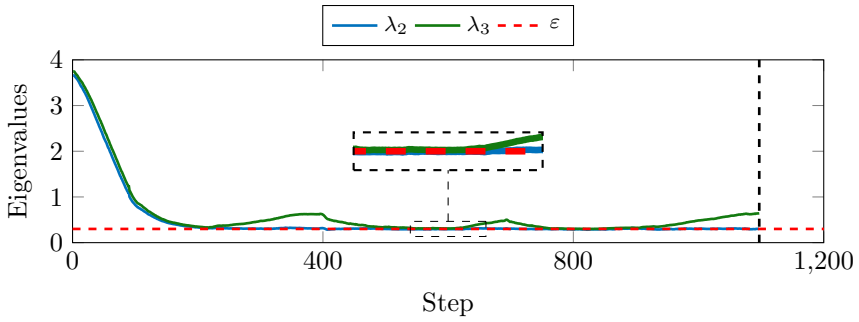


Figure 3.5: Eigenvalue evolution during the experiment. The *dashed black* lines represents the end of the task.

to see how the robots that have already reached their target cooperate to connectivity. Fig. 3.7 shows the applied control inputs: here, the jittering is due both to the non-idealities introduced while using wheeled robots, which hardly instantaneously follow an omnidirectional dynamics, and the time needed for the calculation, which sometimes introduces a small delay. In fact, the time required to let the saddle-point dynamics converge is longer than the time needed to update the control input of the robots, which run at 10 Hz. Despite the limitations of the calculation and of the input of the robots, we achieve good performance also in satisfying the nominal constraint, cf. Fig. 3.8.

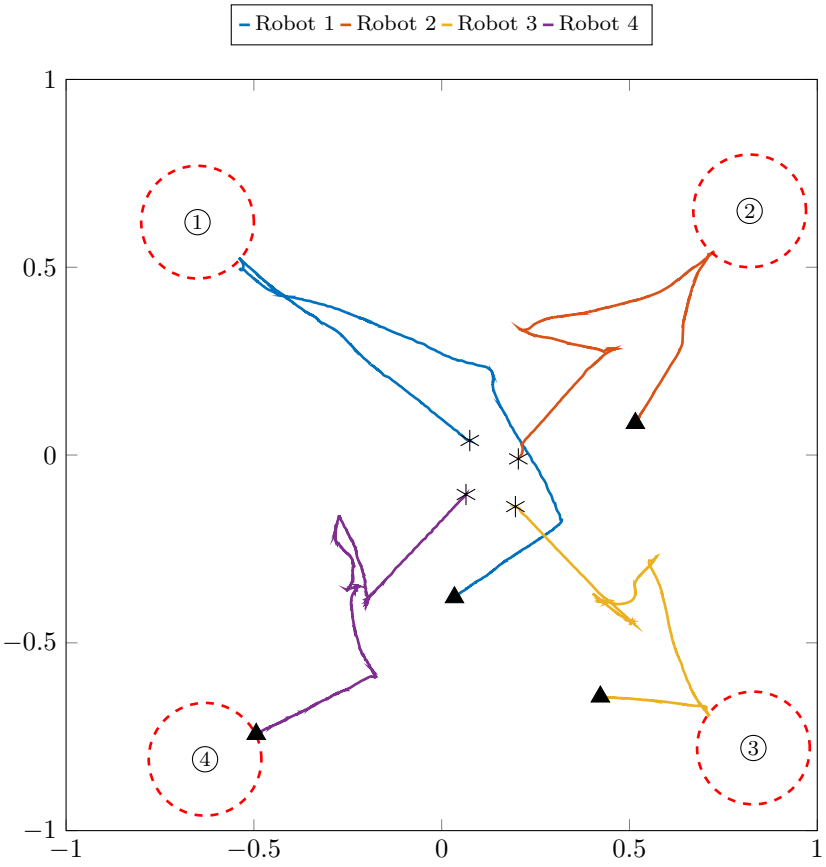


Figure 3.6: Trajectories followed by the robots during the experiment. The *dotted red* circles represent the region where we consider the target reached (circle of 15 cm of radius around the target). The numbers represent the order of priority of the targets. The initial and final positions are reported with *asterisks* and *triangles*, respectively.

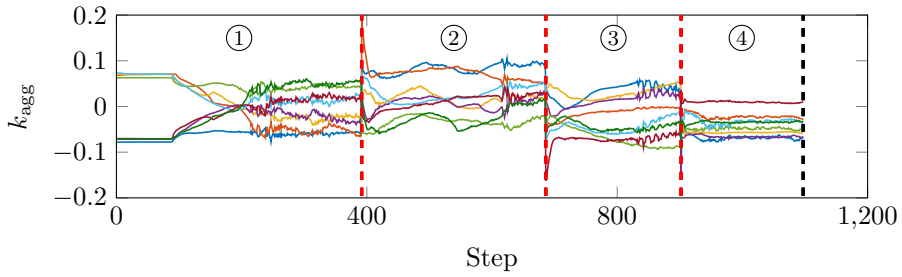


Figure 3.7: Control input applied to the robots in the experiment. We report all the components of the control input for each robot. These inputs have been transformed via input-output state-feedback linearization to be executed by the robots.

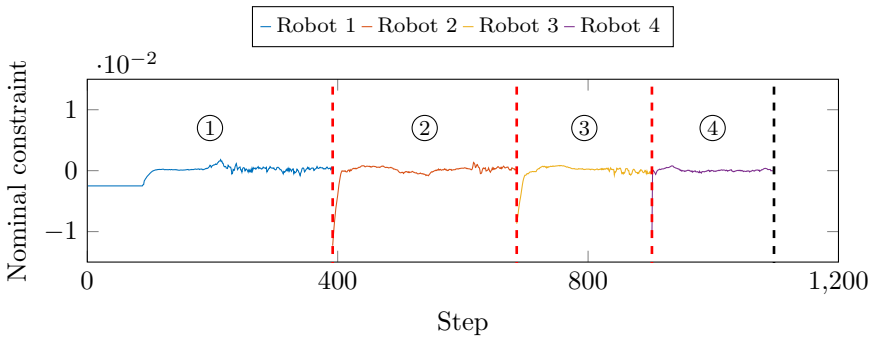


Figure 3.8: Nominal constraint (3.14) during the experiment. The *dashed red* lines represent the instant in which the robot priority changes, due to the fact that a target has been reached. The number for each time slot corresponds to the robot with the highest priority.

3.6 Chapter Appendix

Here we provide several results on the continuity properties of eigenspaces, with the ultimate goal of establishing that the merged eigenspaces are a continuous function of the state as long as its dimension remains constant, cf. Theorem 3.4.1[2].

Given indices $\mathcal{I} \in [n]$, consider the merged eigenspaces $\mathcal{V}_{\mathcal{I}}$. For the purpose of analysis, instead of writing $\mathcal{V}_{\mathcal{I}}$ as a span, we write out the full set definition as follows

$$\mathcal{V}_{\mathcal{I}}(\mathbf{x}) = \left\{ \mathbf{v} \in \mathbb{R}^n \mid (\mathbf{L}(x) - \lambda_i(x)\mathbf{I})\xi_i = 0,\ \forall i \in \mathcal{I}, \mathbf{v} = \sum_{i \in \mathcal{I}} \mathbf{c}_i \xi_i,\ \mathbf{c} \in \mathbb{R}^{|\mathcal{I}|},\ \|\mathbf{v}\| = 1 \right\}. \tag{3.15}$$

For this set-valued map, we will show UHC and LHC separately.

3.6.1 Upper Hemicontinuity of Merged Eigenspaces

For the analysis of (3.15), it is convenient to use the eigenbasis as the coordinate system. Given a state $\mathbf{x}^* \in \mathbb{R}^N$ at which we seek to prove continuity, let the matrix $\mathbf{T} \in \mathbb{R}^{n \times n}$ be an orthonormal eigenbasis of the symmetric matrix $\mathbf{L}(\mathbf{x}^*)$. Furthermore, for each eigenvalue $\lambda_i(\mathbf{x}^*)$, we define $\mathbf{T}_i \in \mathbb{R}^{n \times n}$ with a permutation so that the eigenvectors associated with $\lambda_i(\mathbf{x}^*)$ appear in the last columns of the matrix. As a consequence, we can define the similar matrix

$$\mathbf{D}^{(i)}(\mathbf{x}) := \mathbf{T}_i^\top \mathbf{L}(\mathbf{x}) \mathbf{T}_i.$$

[2] Although this result is seemingly intuitive, we have not found it in the literature. There are results (e.g., [Kat76, Ch. 2.5.3]) that study the continuity properties of eigenvectors when their eigenvalues have multiplicity of one, a case where the eigenvectors can be viewed as a single-valued function. Instead, we investigate eigenspaces of eigenvalues with higher multiplicity, which requires set-valued analysis.

Note importantly that the matrix $\mathbf{T}_i$ is defined in relation to the state $\mathbf{x}^*$ and is constant for all $\mathbf{x}$, so $\mathbf{D}^{(i)}$ is continuous. On the other hand, $\mathbf{T}_i$ being constant does not guarantee that $\mathbf{D}^{(i)}$ will be diagonal at the states other than $\mathbf{x}^*$. Furthermore, by defining the matrix $\mathbf{B}^{(i)}(\mathbf{x}) := \mathbf{D}^{(i)}(\mathbf{x}) - \lambda_i(\mathbf{x})\mathbf{I}$, we can equivalently write each eigenequation with

$$\begin{bmatrix} \mathbf{B}^{(i)}_{aa}(x) & \mathbf{B}^{(i)}_{ab}(x) \\ \mathbf{B}^{(i)}_{ab}(x)^\top & \mathbf{B}^{(i)}_{bb}(x) \end{bmatrix} \begin{bmatrix} \mathbf{w}_{i,a} \\ \mathbf{w}_{i,b} \end{bmatrix} = 0, \tag{3.16}$$

where $\mathbf{w}_i$ is the vector ξ_i in the coordinate system $\mathbf{T}_i$, i.e., $\xi_i = \mathbf{T}_i\mathbf{w}_i$. Above, we partition the matrix $\mathbf{B}^{(i)}$ and the vector $\mathbf{w}_i$ so that $\mathbf{w}_{i,b}$ has the same dimension as the eigenspace associated with λ_i at $\mathbf{x}^*$. The next result shows that each individual eigenspace, when normalized, is already UHC.

Lemma 3.6.1. ***(UHC of individual eigenspaces).*** *Consider a continuous function* $\mathbf{L} : \mathbb{R}^N \to \mathrm{Sym}_n$. *Given a state* $\mathbf{x}^*$ *and* $\delta_{\mathbf{w}} > 0$, *there exists* $\delta_{\mathbf{x}} > 0$ *small enough such that if* $\mathbf{x} \in \mathbb{B}_{\delta_{\mathbf{x}}}(\mathbf{x}^*)$, *then for any* $\mathbf{w}_i$ *satisfying* $\mathbf{B}^{(i)}(\mathbf{x})\mathbf{w}_i = 0$, *there exists* $\mathbf{w}^*$ *satisfying* $\mathbf{B}^{(i)}(\mathbf{x}^*)\mathbf{w}^* = 0$ *with* $\|\mathbf{w}_i - \mathbf{w}^*\| < \delta_{\mathbf{w}}\|\mathbf{w}_i\|$.

Proof. Because $\mathbf{B}^{(i)}_{aa}(\mathbf{x}^*)$ is invertible, there exists $\bar{\delta}_{\mathbf{x}} > 0$ such that $\mathbf{B}^{(i)}_{aa}(\mathbf{x})$ remains invertible for each $\mathbf{x} \in \mathbb{B}_{\bar{\delta}_{\mathbf{x}}}(\mathbf{x}^*)$. From (3.16),

$$\mathbf{w}_{i,a} = \mathbf{B}^{(i)}_{aa}(\mathbf{x})^{-1}\mathbf{B}^{(i)}_{ab}(\mathbf{x})\mathbf{w}_{i,b}.$$

Because $\mathbf{B}^{(i)}_{aa}(\mathbf{x})^{-1}\mathbf{B}^{(i)}_{ab}(\mathbf{x})$ is continuous on $\mathbb{B}_{\bar{\delta}_{\mathbf{x}}}(\mathbf{x}^*)$, given $\delta_{\mathbf{w}}$, there exists $0 < \delta_{\mathbf{x}} < \bar{\delta}_{\mathbf{x}}$ such that $\|\mathbf{B}^{(i)}_{aa}(\mathbf{x})^{-1}\mathbf{B}^{(i)}_{ab}(\mathbf{x})\|_F < \delta_{\mathbf{w}}/\sqrt{2}$ for all $\mathbf{x} \in \mathbb{B}_{\delta_{\mathbf{x}}}(\mathbf{x}^*)$. Then,

$$\|\mathbf{w}_{i,a}\| \le \|\mathbf{B}^{(i)}_{aa}(\mathbf{x})^{-1}\mathbf{B}^{(i)}_{ab}(\mathbf{x})\|_F\|\mathbf{w}_{i,b}\| < \delta_{\mathbf{w}}\|\mathbf{w}_i\|/\sqrt{2}.$$

This also implies $\|\mathbf{w}_{i,b}\| > \|\mathbf{w}_i\|(1 - \delta_{\mathbf{w}}/\sqrt{2})$. Let $\mathbf{w}^* = \begin{bmatrix} 0 & \mathbf{w}_{i,b}^\top \|\mathbf{w}_i\|/\|\mathbf{w}_{i,b}\| \end{bmatrix}^\top$ (and $\mathbf{w}^* = 0$ if $\|\mathbf{w}_b\| = 0$), then $\mathbf{B}^{(i)}(\mathbf{x}^*)\mathbf{w}^* = 0$ because $\mathbf{B}_{ab}^{(i)}(\mathbf{x}^*)$ and $\mathbf{B}_{bb}^{(i)}(\mathbf{x}^*)$ are zero by construction. Also, we can bound the distance

$$\begin{aligned}
\|\mathbf{w}_i - \mathbf{w}^*\| &= \left\| \begin{bmatrix} \mathbf{w}_{i,a} \\ \mathbf{w}_{i,b} \end{bmatrix} - \begin{bmatrix} 0 \\ \mathbf{w}_{i,b}\|\mathbf{w}_i\|/\|\mathbf{w}_{i,b}\| \end{bmatrix} \right\| \\
&= (\|\mathbf{w}_{i,a}\|^2 + \|\mathbf{w}_{i,b}\|^2(1 - \|\mathbf{w}_i\|/\|\mathbf{w}_{i,b}\|)^2)^{1/2} \\
&= (\|\mathbf{w}_{i,a}\|^2 + (\|\mathbf{w}_{i,b}\| - \|\mathbf{w}_i\|)^2)^{1/2} \\
&< \delta_{\mathbf{w}}\|\mathbf{w}_i\|,
\end{aligned}$$

and the proof concludes. □

From this result for individual eigenspaces, we can deduce further that any merged eigenspace is UHC.

Theorem 3.6.2. ***(UHC of Merged Eigenspaces).*** *Consider a continuous function* $\mathbf{L} : \mathbb{R}^N \to \mathrm{Sym}_n$. *For any* $\mathcal{I} \subseteq [n]$, *the merged eigenspace* $\mathcal{V}_\mathcal{I}$ *is UHC.*

Proof. Given any $\mathbf{v} \in \mathcal{V}_\mathcal{I}(\mathbf{x})$ in (3.15), we assume, without loss of generality, that if $\lambda_j(\mathbf{x}) = \lambda_i(\mathbf{x})$ for some $i > j$, the associated eigenvector ξ_j is zero. This way, there is only one nonzero vector ξ_i from each eigenspace. In addition, by scaling ξ_i, we can assume $\mathbf{c} = \mathbb{1}$. Using these simplifications, $\|\xi_i\| \leq 1$ because of the orthogonality of eigenspaces and the fact $\|\mathbf{v}\| = 1$. Thus, when we transform the coordinate frame $\mathbf{w}_i = \mathbf{T}_i^\top \xi_i$, we also guarantee $\|\mathbf{w}_i\| \leq 1$. This is particularly useful when we apply Lemma 3.6.1 as follows.

Consider any arbitrary $\mathbf{x}^*$ at which we wish to prove UHC for $\mathcal{V}_\mathcal{I}$. Lemma 3.6.1 guarantees

for any given $\delta_{\mathbf{w}} > 0$ the existence of a small enough neighborhood $\mathbb{B}_{\delta_{\mathbf{x}}}(\mathbf{x}^*)$ such that for every $\mathbf{x} \in \mathbb{B}_{\delta_{\mathbf{x}}}(\mathbf{x}^*)$, any $\mathbf{w}_i$ satisfying $\mathbf{B}^{(i)}(\mathbf{x})\mathbf{w}_i = 0$ has a corresponding $\mathbf{w}_i^* \in \mathbb{B}_{\delta_{\mathbf{w}}}(\mathbf{w})$ satisfying $\mathbf{B}^{(i)}(\mathbf{x}^*)\mathbf{w}_i^* = 0$. Through coordinate transformation $\xi_i^* = \mathbf{T}_i\mathbf{w}_i^*$, we deduce that given the set of vectors $\{\xi_i\}_{i\in\mathcal{I}}$ defining $\mathbf{v}$, there exists a corresponding set of vectors $\{\xi_i^*\}_{i\in\mathcal{I}}$ such that $\xi_i \in \mathbb{B}_{\delta_{\mathbf{w}}}(\xi_i)$ and $(\mathbf{L}(\mathbf{x}) - \lambda_i(\mathbf{x})\mathbf{I})\xi_i^* = 0$. We then define $\mathbf{v}^* = (\sum \xi_i^*)/\|\sum \xi_i^*\|$, which is an element of the set $\mathcal{V}_{\mathcal{I}}(\mathbf{x}^*)$ by definition.

We next prove that $\mathbf{v}^*$ is close to $\mathbf{v}$ for a small enough $\delta_{\mathbf{w}}$. From the condition $1 = \|\mathbf{v}\| = \|\sum(\xi_i^* + (\xi_i - \xi_i^*))\|$, we can bound the norm $\|\sum \xi_i^*\| \in (1 - n\delta_{\mathbf{w}}, 1 + n\delta_{\mathbf{w}})$. With these facts, we bound the distance

$$\begin{aligned}
\|\mathbf{v} - \mathbf{v}^*\| &= \left\| \sum \xi_i - \frac{\sum \xi_i^*}{\|\sum \xi_i^*\|} \right\| \\
&\leq \|\sum(\xi_i - \xi_i^*)\| + \left\| \sum \left(\xi_i^* - \frac{\xi_i^*}{\|\sum \xi_i^*\|} \right) \right\| \\
&\leq n\delta_{\mathbf{w}} + (1 + n\delta_{\mathbf{w}})n\delta_{\mathbf{w}}/(1 - n\delta_{\mathbf{w}}) \\
&= 2n\delta_{\mathbf{w}}/(1 - n\delta_{\mathbf{w}}).
\end{aligned}$$

Given any $\delta_{\mathbf{v}}$, we can pick $\delta_{\mathbf{w}}$ small enough so that $\|\mathbf{v} - \mathbf{v}^*\| < \delta_{\mathbf{v}}$, i.e., $\mathbf{v} \in \mathbb{B}_{\delta_{\mathbf{v}}}(\mathbf{v}^*)$.

We have shown that given any $\delta_{\mathbf{v}} > 0$, there exists $\delta_{\mathbf{x}} > 0$ such that any $\mathbf{v} \in \mathcal{V}_{\mathcal{I}}(\mathbf{x})$, for $\mathbf{x} \in \mathbb{B}_{\delta_{\mathbf{x}}}(\mathbf{x}^*)$, has a corresponding $\mathbf{v}^* \in \mathcal{V}_{\mathcal{I}}(\mathbf{x}^*)$ such that $\mathbf{v} \in \mathbb{B}_{\delta_{\mathbf{v}}}(\mathbf{v}^*)$. In other words, $\mathcal{V}_{\mathcal{I}}(\mathbf{x})$, for $\mathbf{x} \in \mathbb{B}_{\delta_{\mathbf{x}}}(\mathbf{x}^*)$, is a subset of a $\delta_{\mathbf{v}}$ neighborhood of $\mathcal{V}_{\mathcal{I}}(\mathbf{x}^*)$, which is precisely the definition of UHC, concluding the proof. □ □

3.6.2 Lower Hemicontinuity of Merged Eigenspaces

Unlike the case of UHC, individual normalized eigenspaces are not LHC everywhere. Therefore, we proceed directly to the analysis of the merged eigenspaces. We define, for an index set $\mathcal{I} \subseteq [n]$, an orthonormal eigenbasis matrix $\mathbf{T}_{\mathcal{I}} \in \mathbb{R}^{n\times n}$, with the eigenvectors associated with $\lambda_i(\mathbf{x}^*)$ for $i \in \mathcal{I}$ showing up in the last columns of the matrix. Then, we define the matrix

$$\mathbf{D}^{\mathcal{I}}(\mathbf{x}) = \mathbf{T}_{\mathcal{I}}^{\top}\mathbf{L}(\mathbf{x})\mathbf{T}_{\mathcal{I}}.$$

The next result establishes the LHC property of the merged eigenspaces.

Theorem 3.6.3. ***(LHC of Merged Eigenspaces).*** *Consider a continuous function* $\mathbf{L} : \mathbb{R}^N \to \mathrm{Sym}_n$. *For any* $\mathcal{I} \subseteq [n]$, *the merged eigenspace is LHC at* $\mathbf{x}$ *where* $\lambda_i(\mathbf{x}) \neq \lambda_j(\mathbf{x})$ *for all* $i \in \mathcal{I}$ *and* $j \notin \mathcal{I}$, *i.e., where none of the eigenvalues considered in the span is equal to any of the remaining eigenvalue.*

Proof. Consider the change of coordinate frame $\xi_i = \mathbf{T}_{\mathcal{I}}\mathbf{w}_i$, for each $i \in [n]$. The merged eigenspace given by (3.15) can be rewritten as

$$\mathcal{V}_{\mathcal{I}}(\mathbf{x}) = \left\{ \mathbf{v} \in \mathbb{R}^n \mid (\mathbf{D}^{\mathcal{I}}(\mathbf{x}) - \lambda_i(\mathbf{x})\mathbf{I})\mathbf{w}_i = 0,\ \forall i \in \mathcal{I}, \mathbf{v} = \mathbf{T}_{\mathcal{I}}\mathbf{W}\mathbf{c},\ \mathbf{c} \in \mathbb{R}^{|\mathcal{I}|},\ \|\mathbf{v}\| = 1 \right\},$$

where $\mathbf{W} \in \mathbb{R}^{n\times|\mathcal{I}|}$ is a matrix constructed by stacking $\mathbf{w}_i$ together. By construction, given an element $\mathbf{v}^* \in \mathcal{V}_{\mathcal{I}}(\mathbf{x}^*)$, it must take the form $\mathbf{v}^* = \mathbf{T}_{\mathcal{I}}\begin{bmatrix} 0 \\ \psi \end{bmatrix}$ for some $\psi \in \mathbb{R}^{|\mathcal{I}|}$.

Consider $\mathbf{x}^*$ at which we wish to prove LHC for $\mathcal{V}_{\mathcal{I}}$. We next show the existence of $\mathbf{v} \in \mathcal{V}_{\mathcal{I}}(\mathbf{x})$

close enough to $\mathbf{v}^*$ for all $\mathbf{x}$ close enough to $\mathbf{x}^*$. First, we partition the eigenequations,

$$\left(\begin{bmatrix} \mathbf{D}^{\mathcal{I}}_{aa}(\mathbf{x}) & \mathbf{D}^{\mathcal{I}}_{ab}(\mathbf{x}) \\ \mathbf{D}^{\mathcal{I}}_{ab}(\mathbf{x})^\top & \mathbf{D}^{\mathcal{I}}_{bb}(\mathbf{x}) \end{bmatrix} - \lambda_i(\mathbf{x})\mathbf{I}\right)\begin{bmatrix} \mathbf{w}_{i,a} \\ \mathbf{w}_{i,b} \end{bmatrix} = 0,$$

so that $\mathbf{w}_{i,b}$ has the dimension of $|\mathcal{I}|$. The matrix $\mathbf{D}^{\mathcal{I}}_{aa}(\mathbf{x}^*)$ is a diagonal matrix of eigenvalues $\lambda_j(\mathbf{x}^*)$ for $j \notin \mathcal{I}$. Because $\lambda_j(\mathbf{x}^*) \neq \lambda_i(\mathbf{x}^*)$ for any $i \in \mathcal{I}$ and $j \notin \mathcal{I}$, the matrix $\mathbf{D}^{\mathcal{I}}_{aa}(\mathbf{x}) - \lambda_i(\mathbf{x})\mathbf{I}$ is invertible at $\mathbf{x} = \mathbf{x}^*$. Then due to continuity of the matrix, there exists $\bar{\delta}_{\mathbf{x}}$ such that it remains invertible for $\mathbf{x} \in \mathbb{B}_{\bar{\delta}_{\mathbf{x}}}(\mathbf{x}^*)$, and we can find the following relationship,

$$\mathbf{w}_{i,a} = (\mathbf{D}^{\mathcal{I}}_{aa}(\mathbf{x}) - \lambda_i(\mathbf{x})\mathbf{I})^{-1}\mathbf{D}^{\mathcal{I}}_{ab}(\mathbf{x})\mathbf{w}_{i,b}.$$

Due to continuity of the matrix $\mathbf{D}^{\mathcal{I}}$ and the fact that $\mathbf{D}^{\mathcal{I}}_{ab}(\mathbf{x})$ is a zero matrix at $\mathbf{x} = \mathbf{x}^*$, we can further find that given $\delta_{\mathbf{w}}$, there exists $0 < \delta_{\mathbf{x}} \leq \bar{\delta}_{\mathbf{x}}$ such that $\|\mathbf{w}_{i,a}\| \leq \delta_{\mathbf{w}}\|\mathbf{w}_{i,b}\|$ for all $\mathbf{x} \in \mathbb{B}_{\delta_{\mathbf{x}}}(\mathbf{x}^*)$. With this property, we construct $\mathbf{v} \in \mathcal{V}_{\mathcal{I}}(\mathbf{x})$ with the following procedure.

We begin by selecting the set of eigenvectors $\{\mathbf{w}_i\}_{i\in\mathcal{I}}$ to be orthonormal to one another. This set of eigenvectors must exist because $\mathbf{D}^{\mathcal{I}}(\mathbf{x})$ is symmetric. With this choice, we can show that when we partition the matrix $\mathbf{W} = \begin{bmatrix} \mathbf{W}_a \\ \mathbf{W}_b \end{bmatrix}$, $\mathbf{W}_b$ is an invertible matrix for $\mathbf{x} \in \mathbb{B}_{\delta_{\mathbf{x}}}(\mathbf{x}^*)$. We prove this statement by contradiction. Assume that $\mathbf{W}_b$ is not full rank, then there exists a vector $0 \neq \mathbf{c} \in \mathbb{R}^{|\mathcal{I}|}$ such that $\mathbf{W}_b\mathbf{c} = 0$. In addition, $\mathbf{W}^\top\mathbf{W} = \mathbf{I}$ because $\mathbf{w}_i$ are orthogonal to each other. Thus,

$$\|\mathbf{c}\| = \|\mathbf{W}^\top\mathbf{W}\mathbf{c}\| = \|\mathbf{W}_a^\top\mathbf{W}_a\mathbf{c}\| \leq \delta_{\mathbf{w}}^2|\mathcal{I}|^2\|\mathbf{c}\|,$$

which is a contradiction for small $\delta_{\mathbf{w}}$. Since $\mathbf{W}_b$ is invertible, we can define the vector

$$\bar{\mathbf{v}} = \mathbf{T}_{\mathcal{I}} \mathbf{W} \mathbf{W}_b^{-1} \psi = \mathbf{T}_{\mathcal{I}} \begin{bmatrix} \mathbf{W}_a \mathbf{W}_b^{-1} \psi \\ \psi \end{bmatrix},$$

which we use to construct $\mathbf{v} \in \mathcal{V}_{\mathcal{I}}(\mathbf{x})$ Before doing so, we upper bound $\mathbf{W}_a \mathbf{W}_b^{-1} \psi$. Note that

$$\begin{aligned} \|\mathbf{W}_a \mathbf{W}_b^{-1} \psi\| &= \|\mathbf{W}_a (W_b^\top \mathbf{W}_b)^{-1} \mathbf{W}_b^\top \psi\| \\ &\leq \|\mathbf{W}_a\| \|(\mathbf{W}_b^\top \mathbf{W}_b)^{-1}\| \|\mathbf{W}_b\|. \end{aligned}$$

Here, we can bound $\|\mathbf{W}_b\| \leq |\mathcal{I}|$ due to normality of each $\mathbf{w}_i$. Also from the earlier fact $\|\mathbf{w}_{i,a}\| \leq \delta_{\mathbf{w}} \|\mathbf{w}_{i,b}\| \leq \delta_{\mathbf{w}}$, we bound $\|\mathbf{W}_a\| \leq \delta_{\mathbf{w}} |\mathcal{I}|$. As for the $\|(\mathbf{W}_b^\top \mathbf{W}_b)^{-1}\|$, we investigate the smallest eigenvalue of $(\mathbf{W}_b^\top \mathbf{W}_b)$. Due to orthonormality,

$$\mathbf{w}_{i,b}^\top \mathbf{w}_{j,b} = \begin{cases} -\mathbf{w}_{i,a}^\top \mathbf{w}_{j,a} & i \neq j, \\ 1 - \mathbf{w}_{i,a}^\top \mathbf{w}_{j,a} & i = j. \end{cases}$$

Combined with the fact $\|\mathbf{w}_{i,a}\| \leq \delta_{\mathbf{w}}$, we upper bound the off-diagonal entries of $\mathbf{W}_b^\top \mathbf{W}_b$ with $\delta_{\mathbf{w}}^2$, and we lower bound the diagonal entries with $1 - \delta_{\mathbf{w}}^2$. Using the Gershgorin circle theorem [BCM09, Thm. 1.3], the smallest eigenvalue of $\mathbf{W}_b^\top \mathbf{W}_b$ is lower bounded by $1 - |\mathcal{I}| \delta_{\mathbf{w}}^2$. Using these bounds, we find

$$\|\mathbf{W}_a \mathbf{W}_b^{-1} \psi\| \leq \frac{\delta_{\mathbf{w}} |\mathcal{I}|}{1 - \delta_{\mathbf{w}}^2} := \delta_{\mathbf{v}}.$$

Note importantly that smaller $\delta_{\mathbf{v}}$ corresponds to small $\delta_{\mathbf{w}} < 1$.

Finally, we select $\mathbf{c} = \mathbf{w}_b^{-1}\psi/\|\bar{\mathbf{v}}\|$ to construct $\mathbf{v} = \bar{\mathbf{v}}/\|\bar{\mathbf{v}}\|$, which is an element of $\mathcal{V}_{\mathcal{I}}(\mathbf{x})$. Let θ be the angle between the unit vectors $\mathbf{v}$ and $\mathbf{v}^*$. Then, we bound

$$\|\mathbf{v} - \mathbf{v}^*\| \leq \theta \leq \tan\theta = \frac{\|\mathbf{W}_a\mathbf{W}_b^{-1}\psi\|}{\|\psi\|} \leq \delta_{\mathbf{v}}.$$

Thus, we have proven that given any $\delta_{\mathbf{v}} > 0$, there exists $\delta_{\mathbf{x}} > 0$ such that if $\mathbf{x} \in \mathbb{B}_{\delta_{\mathbf{x}}}(\mathbf{x}^*)$, then there exists $\mathbf{v} \in \mathcal{V}_{\mathcal{I}}(\mathbf{x})$ where $\mathbf{v} \in \mathbb{B}_{\delta_{\mathbf{v}}}(\mathbf{v}^*)$. This is sufficient to prove that given any sequence $\{\mathbf{x}^k\}_{k\in\mathbb{N}}$ converging to $\mathbf{x}^*$, there exists a sequence $\{\mathbf{v}^k\}_{k\in\mathbb{N}}$, with $\mathbf{v}^k \in \mathcal{V}_{\mathcal{I}}(\mathbf{x}^k)$, converging to $\mathbf{v}^*$, concluding the proof. □

Acknowledgements

This chapter, in part, has been submitted for publication of the material [OCSC21] as it may appear as 'Nonsmooth Control Barrier Function Design of Continuous Constraints for Network Connectivity Maintenance' by P. Ong, B. Capelli, L. Sabattini, and J. Cortés, in The International Journal of Robotics Research. The dissertation author was the primary investigator and author of this paper. The work in this chapter was partially supported by NSF Award ECCS-1917177.

Chapter 4

Smooth Safe Stabilization Formula

This chapter formulates a safe and stabilizing control state feedback law for a control affine nonlinear system. We assume that there exist a known control Lyapunov function (CLF) and a control barrier function (CBF) that are compatible, i.e., there exists a control choice satisfying the conditions given by both the CLF and CBF at each given state. In contrast to the approach in the literature of finding a minimum-norm control using optimization on the feasible control set, we take a different approach by finding and combining different weighted centroids of the feasible control set. As a result, we can propose a control feedback law with guaranteed smoothness everywhere except at the origin, and with guaranteed continuity at the origin if small control property holds.

4.1 Problem Statement

Consider a nonlinear control-affine system of the form

$$\dot{x} = f(x) + g(x)u, \tag{4.1}$$

where $x \in \mathbb{R}^n$ is the state and $u \in \mathbb{R}^m$ is the input. The system vector fields $f : \mathbb{R}^n \to \mathbb{R}^n$ and $g : \mathbb{R}^n \to \mathbb{R}^{n\times m}$ are assumed to be smooth, and $f(0) = 0$ so that the origin is an equilibrium of the unforced system. The goal of this paper is to find a control feedback $k : \mathbb{R}^m \to \mathbb{R}^n$ such that $u = k(x)$ guarantees both the global asymptotic stability of the origin and the safety of the trajectories for the closed-loop system. We address the problem of asymptotic stability and safety with a control Lyapunov function and a control barrier function, resp., whose definitions we recall next.

Definition 4.1.1. ***(Control Lyapunov Function).*** *Given a function $\delta : \mathbb{R}^n \to \mathbb{R}_{\geq 0}$, a continuously differentiable function $V : \mathbb{R}^n \to \mathbb{R}$ is a* ***δ-relaxed Control Lyapunov Function (δ-CLF)*** *for the system* (4.1) *if*

(i) V is proper, i.e., $\{x \in \mathbb{R}^n \mid V(x) \leq c\}$ is a compact set for all $c > 0$;

(ii) V is positive definite;

(iii) For each $x \in \mathbb{R}^n \setminus \{0\}$, $\exists u \in \mathbb{R}^m$ such that

$$L_f V(x) + L_g V(x) u < \delta(x). \qquad \diamond \tag{4.2}$$

The standard notion of CLF (cf. [FK96, Section 3.3.1]) corresponds to $\delta(x) = 0$ in this definition when applied specifically to the system (4.1), and we refer to it by 0-CLF. One can indeed find in the literature different variations of this notion: for example, the right-hand side of the inequality (4.2) may include a negative definite function [Son99] or the function V might only be continuous, in which case the monotonic property in (iii) is expressed in terms of directional

derivatives [SS96]. Throughout the paper, we focus the definition above, even though our results are generalizable to other cases as well. Similarly, we define a CBF as follows.

Definition 4.1.2. ***(Control Barrier Function [AXGT17, XTGA15]).*** *Given a open set $\mathcal{D} \subset \mathbb{R}^n$, a function $h : \mathbb{R}^n \to \mathbb{R}$, continuously differentiable on $\mathbb{R}^n \setminus \mathcal{D}$ is a* ***control barrier function (CBF)*** *for the system* (4.1) *if*

(i) $h(x) = 0$ *for all* $x \in \partial\mathcal{D}$*;*

(ii) $h(x) < 0$ *for all* $x \in \mathbb{R}^n \setminus \mathcal{D}$*;*

(iii) For each $x \in \mathbb{R}^n \setminus \mathcal{D}$, $\exists u \in \mathbb{R}^m$ *such that*

$$L_f h(x) + L_g h(x) u \leq \beta(-h(x)) \tag{4.3}$$

where β is a Lipschitz class-$\mathcal{K}$ function. ◇

The set $\mathcal{D}$ is referred to as the *unsafe* set. The purpose of a CBF is to guarantee that all trajectories with an initial condition outside of $\mathcal{D}$ will not enter it. For instance, when the right-hand side of inequality (4.3) is zero, then the condition guarantees that the value of the function h does not increase along the trajectory. Therefore, if the initial condition is outside $\mathcal{D}$, where the value of h is negative, then the function will remain negative, and the trajectory will not enter the set $\mathcal{D}$. The function β in the right-hand side allows for the value of h to actually increase on points in the interior of $\mathbb{R}^n \setminus \mathcal{D}$. As a result, the trajectory still avoids the set $\mathcal{D}$.

We assume for the rest of the paper that the system (4.1) admits a 0-CLF and a CBF. With these functions, one can deduce particularly that, if there exists a feedback $u = k(x)$ satisfying

inequalities (4.2) and (4.3), global asymptotic stability of the origin and safety of the trajectories can be guaranteed. This motivates us to introduce the concept of compatibility.

Definition 4.1.3. ***(Compatibility).*** *We refer to a collection of inequalities of the form* $a(x)+b(x)u < 0$ *or* $a(x) + b(x)u \leq 0$ *as* ***(strictly) compatible on*** $\mathcal{X} \subset \mathbb{R}^n$ *if, for each* $x \in \mathcal{X}$*, there exists a corresponding* $u \in \mathbb{R}^m$ *satisfying all inequalities (strictly). We call a* δ*-CLF* V *and a CBF* h *compatible if their inequalities* (4.2) *and* (4.3) *are compatible on* $\mathbb{R}^n \setminus (\mathcal{D} \cup \{0\})$. ⋄

Given a 0-CLF V and a CBF h, we want to emphasize the importance of their compatibility. If they are not compatible, this means that there exists at least a state x where there is no control u that can satisfy inequalities (4.2) and (4.3) simultaneously, and either stability or safety needs to be sacrificed. In the literature, it is common to sacrifice stability by allowing $\delta(x) \neq 0$ to ensure compatibility. We come back to this point later in Section 4.5.

On top of compatibility, it is also important that the control as a feedback function is at least Lipschitz continuous, to guarantee the existence and uniqueness of solutions. This motivates the formulation of the main problem.

Problem 2. ***(Feedback Safe Stabilization).*** *Assume we are given a* 0*-CLF* $V : \mathbb{R}^n \to \mathbb{R}$ *and a CBF* $h : \mathbb{R}^n \to \mathbb{R}$ *for the system* (4.1) *which are compatible. Find a smooth control feedback* $k : \mathbb{R}^m \to \mathbb{R}^n$ *such that* $u = k(x)$ *satisfies both inequalities* (4.2) *and* (4.3) *for all* $x \in \mathbb{R}^n \setminus (\mathcal{D} \cup \{0\})$.

In what follows, we show that the problem above has a solution when the 0-CLF and a CBF are strictly compatible, and provide a constructive formula for the feedback.

4.2 Existence of a Smooth Control Feedback

In this section, we examine the existence of the solution to Problem 2. The work [Son89] shows that it is possible to construct a smooth control feedback when stability is our only concern. Prior to providing its formula for universal stabilization, the work suggests that the existence of a smooth feedback controller can be derived through extending Artstein's Theorem for the existence of a continuous feedback controller (cf. [Art83]) by considering a partition of unity. Incidentally, using the same concept, we can extend Artstein's Theorem even further to when more than one control-affine inequalities are considered.

Proposition 4.2.1. ***(Extension of Artstein's Theorem).*** *Consider a collection of n inequalities of the form* $a(x)+b(x)u < 0$ *or* $a(x)+b(x)u \leq 0$ *each defined on the domain* $\mathcal{X}_i \subseteq \mathbb{R}^n$ *with* $a_i : \mathcal{X}_i \to \mathbb{R}$ *and* $b_i : \mathcal{X}_i \to \mathbb{R}^m$ *continuous. If the inequalities are strictly compatible on* $\cap_{i\in[n]}\mathcal{X}_i$*, then there exists a* C^∞ *selection function* $k : \cap_{i\in[n]}\mathcal{X}_i \to \mathbb{R}^m$ *such that* $u = k(x)$ *satisfy all the inequalities for all* $x \in \cap_{i\in[n]}\mathcal{X}_i$. □

Proof. For each x, let $v(x)$ denote the given u that strictly satisfies all inequalities. Due to continuity of a_i and b_i, there exists a neighborhood of x, denoted by $\mathcal{W}(x)$, where $v(x)$ strictly satisfies all the inequalities for all $y \in \mathcal{W}(x)$. The collection of $\{\mathcal{W}(x)\}_{x\in\cap_{i\in[n]}\mathcal{X}_i}$ is an open cover for $\cap_{i\in[n]}\mathcal{X}_i$. Then, because we deal with a Euclidean space which is a differentiable manifold, there exists a countable partition of unity $\{\psi_j\}$ subordinate to the cover, cf. [War89b, Theorem 1.11]. In other words, for each j, there exists an x such that $\text{supp}(\psi_j)$ is a subset of $\mathcal{W}(x)$, each of which has an associated control $v(x)$ satisfying all inequalities, which we now label as v_j. Then we define $k(x) = \sum_j \psi_j(x)v_j$ which satisfies the statement because of convexity. □

In the context of safe stabilization, Proposition 4.2.1 provides a non-constructive statement

suggesting that there exists a smooth control feedback for a given 0-CLF and any number of CBFs as long as they are all strictly compatible.

Corollary 4.2.2. ***(Existence of Smooth Safe Stabilizing Control Feedback).*** *For the system* (4.1) *with continuous f and g, Let the system admit a* 0*-CLF and a CBF that are compatible. If $L_g h(x) = 0_m \implies L_f h(x) < \beta(-h(x))$, then there exists a control feedback $k : \mathbb{R}^n \setminus (\mathcal{D} \cup \{0\}) \to \mathbb{R}^m$, C^∞ on $\mathbb{R}^n \setminus (\mathcal{D} \cup \{0\})$ such that $u = k(x)$ satisfies both* (4.2) *and* (4.3) *for each $x \in \mathbb{R}^n \setminus (\mathcal{D} \cup \{0\})$, and hence, global asymptotically safely stabilizes the closed-loop system* (4.1) *when $\delta = 0$.* □

Proof. By the definition of compatibility of the δ-CLF and the CBF, for each $x \in \mathbb{R}^n \setminus (\mathcal{D} \cup \{0\})$, there exists a corresponding $u \in \mathbb{R}^m$ satisfying (4.2) and (4.3). When $L_g h(x) \neq 0_m$, if the given u satisfy (4.3) with an equality, there must exist a u' that satisfy the inequality strictly because the inequality in (4.2) is strict. Under the assumption that $L_g h(x) = 0_m \implies L_f h(x) < \beta(-h(x))$, inequalities (4.2) and (4.3) are strictly compatible. With $L_f V$, $L_g V$, $L_f h$, $L_g h$, and β all being continuous, the assumptions of Proposition 4.2.1 are met, and the proof concludes. □

This result shows that there is a solution to Problem 2 under the additional assumption that $L_g h(x) = 0_m \implies L_f h(x) < \beta(-h(x))$. Although we have not yet worked out a counterexample, we suspect there is one when the condition does not hold. Therefore, we require this condition to hold. Note that in the literature (e.g., minimum-norm controller [AXGT17, XTGA15]), it is common to require that $L_g h(x) \neq 0$, a condition stricter than what we require. The result above does not give a constructive formula. Here, we follow [Son89] and shall construct a formula based on the available 0-CLF and CBF. As a result, the smoothness of the controller will rely heavily on the smoothness of the mentioned functions. As opposed to finding a C^∞ function given only a continuously differentiable 0-CLF and a continuously differentiable CBF, much like Sontag's

universal formula in [Son89], we will require that the two functions are C^{l+1} for which we will give a C^l control feedback formula.

Sontag's universal formula provides, in addition to the smoothness of the controller, continuity at the origin under the small control property of the admissible control set. We shall do the same here for our formula, so we provide the following definition.

Definition 4.2.3. ***(Small Control Property).*** *Given an admissible control set-valued function $\mathcal{U} : \mathbb{R}^n \rightrightarrows \mathbb{R}^m$.* ***Small control property holds with $\mathcal{U}$*** *if for every $\epsilon > 0$, there exists $\omega > 0$ so that there exists $\|u\| < \epsilon$ such that $u \in \mathcal{U}(x)$ for all $\|x\| < \omega$.* ⋄

Note that the usual definition of small control property is tied directly to the control set associated with inequality (4.2) of a 0-CLF. However, since we deal in this paper not only with a 0-CLF, but also an additional condition from a CBF, we give above definition to suit our purpose.

4.3 Alternative Universal Formula for Smooth Stabilization

In finding a constructive control feedback formula, one might want to begin by building on Sontag's universal formula for stabilization (cf. [Son89]) because it already handles smooth stabilization by satisfying inequality (4.2). We will briefly mention here the trouble in generalizing the formula to the case with multiple inequalities. Sontag's formula relies on the fact that the roots of a quadratic function behave analytically with respect to the function's parameters and the fact that function must be decreasing at one of the roots. To guarantee the satisfaction of inequality (4.2), a function quadratic in u is constructed so that the condition for the function decreasing at the root is precisely inequality (4.2). The root is the desired input, and can be computed using the quadratic formula. To develop a universal formula for the case with two inequalities (4.2) and (4.3)

to satisfy, one would have to consider a 2-dimensional quadratic function, and find a formula to the root at which its function's derivative describes both inequalities. Unlike the one-dimensional case, finding such a root can be problematic and its analyticity with respect to the function's parameters is unclear. This motivates the alternative formula proposed here.

Assume $V : \mathbb{R}^n \to \mathbb{R}$ is a 0-CLF for the system (4.1). The inequality (4.2) limits the choice of admissible controls for stabilization. Because of the dependency on x, we write the admissible inputs as a set-valued function $\mathcal{U}_1 : \mathbb{R}^n \rightrightarrows \mathbb{R}^m$,

$$\mathcal{U}_1(x) = \{u \in \mathbb{R}^m \mid L_f V(x) + L_g V(x) u < 0\}.$$

Clearly, if we select $u = k(x) \in \mathcal{U}_1(x)$ for each $x \in \mathbb{R}^n \setminus \{0\}$, then (4.2) is satisfied. However, this is not enough to guarantee stabilization, because the continuity properties of the control might not be enough to guarantee the existence and uniqueness of solutions. To do so, we need a control feedback that is at least Lipschitz continuous. To define this, we rely on the function $\mu : \mathcal{P}(\mathbb{R}^m) \times \mathbb{R}_{\geq 0} \to \mathbb{R}^m$,

$$\mu(S, \sigma) = \frac{\int_S u \exp(-u^\top u/(2\sigma^2)) du}{\int_S \exp(-u^\top u/(2\sigma^2)) du}. \tag{4.4}$$

The function can be interpreted as the mean of a set S with weights from a zero-mean, σ^2-variance Gaussian probability density distribution. We are now ready to propose our alternative universal formula for stabilization.

Proposition 4.3.1. ***(Alternative Universal Formula for Smooth Stabilization).*** *Let $V : \mathbb{R}^n \to \mathbb{R}$ be a 0-CLF for the system (4.1) and $\sigma : \mathbb{R}^n \to \mathbb{R}$ be a C^l positive definite function. If $L_f V$ and $L_g V$ are C^l, then $\mu \circ (\mathcal{U}_1 \times \sigma)$ is C^l on $\mathbb{R}^n \setminus \{0\}$. In addition, the feedback control $u(x) =$*

$\mu(\mathcal{U}_1(x), \sigma(x))$ satisfies (4.2) *for each $x \in \mathbb{R}^n \setminus \{0\}$ and hence globally asymptotically stabilizes the origin. Furthermore, if the small control property holds with $\mathcal{U}_1$, the function k is also continuous at the origin.* □

Proof. We begin by noting that because $\mathcal{U}_1$ is a halfspace, all the components of the weighted centroid along the direction perpendicular to $L_g V$ must evaluate to zero due to symmetry. As such, the solution to the integral must be

$$\mu(\mathcal{U}_1(x), \sigma(x)) = \frac{L_g V(x)^\top}{\|L_g V(x)\|} v(x)$$

where $v : \mathbb{R}^n \to \mathbb{R}$ is a scalar function resulting from an integration along the $L_g V(x)$ direction. $v(x)$ can be directly obtained through calculating a weighted centroid in one dimension. As such, we simply need to show that the result in single input case $m = 1$ is smooth.

Consider a halfspace $\mathcal{H}(a, b) = \{v \in \mathbb{R} \mid a + bv < 0\}$ where $a < 0$ when $b = 0$. Through direct integration, one can find

$$\mu(\mathcal{H}(a,b), \sigma) = \begin{cases} \frac{\sigma p(-a/(b\sigma))}{1-\Phi(-a/(b\sigma))}, & b < 0 \\ 0, & b = 0 \\ \frac{-\sigma p(-a/(b\sigma))}{\Phi(-a/(b\sigma))}, & b > 0 \end{cases}$$

where p and Φ are the probability and cumulative density function of the standard normal distribution, resp. One can check that $\mu(\mathcal{H}(a, b), \sigma)$ and all its derivative are identically zero as $b \to 0$, and therefore it is C^∞ with respect to a, b and σ. With $a = L_f V(x)$ and $b = L_g V(x)$ and the fact that

$L_f V(x) < 0$ when $L_g V(x) = 0$ from condition (iii) in Definition 4.1.1 of a 0-CLF, we deduce that $\mu \circ (\mathcal{U}_1 \times \sigma)$ is C^l.

Next, if small control property holds, we need to show that the control feedback is continuous at $x = 0$. We first note that σ approaches zero near the origin because it is a positive definite function. As a result, the weight determined by a Gaussian probability density is higher at the states closer to the origin. As such, when $x \to 0$, the mean $\mu \circ (\mathcal{U}_1 \times \sigma)$ is in a small neighborhood of the control in the feasible control set with the minimum norm, i.e., $\text{argmin}_{u \in \mathcal{U}_1(x)} \|u\|$. From small control property, we have that for all $\epsilon > 0$ there exists a neighborhood of the origin such that there exists a $u \in \mathcal{U}_1(x)$ such that $\|u\| < \epsilon$. In other words, the minimum-norm control also converges to zero. Hence, continuity holds.

Lastly, the satisfaction of inequality (4.2) is derived from the fact that $\mu \circ (\mathcal{U}_1 \times \sigma)(x)$ is the weighted centroid of $\bar{\mathcal{U}}_1(x)$. Because $\mathcal{U}_1(x)$ is convex, $\mu \circ (\mathcal{U}_1 \times \sigma)(x) \in \text{int}(\mathcal{U}_1(x))$, which concludes the proof. □

Proposition 4.3.1 provides an alternative design to Sontag's universal formula for exploiting the existence of a 0-CLF for global feedback stabilization. Furthermore, as we show in the forthcoming discussion, the construction behind the proposed formula can be extended to accommodate satisfaction of an additional inequality, particularly inequality (4.3) corresponding to a CBF.

4.4 Universal Formula for Smooth Safe Stabilization

In this section we build on the developments of Section 4.3 to deal with an additional inequality from a CBF. Before we move on to give our formula, we first define here a useful auxiliary

function. The following function $s : \mathbb{R} \to [0, 1]$ is C^∞, (cf. [War89b, Eq (3) in Lemma 1.10]),

$$s(t) = \begin{cases} 0, & t \leq 0 \\ \left(1 + \frac{e^{1/t}}{e^{1/(1-t)}}\right)^{-1}, & 0 < t < 1 \\ 1, & t \geq 1 \end{cases} \tag{4.5}$$

The function above is strictly increasing from 0 to 1 in the interval $(0, 1)$. Notably, the function is *flat* (derivatives with respect to t of all order are zeros) at $t = 0$ and $t = 1$. As a result, the function is particularly useful for smoothly transitioning from one function to another in a convex set. Also, we denote with a shorthand notation the following

$$\mu_{[1,2]}(x) = \mu(\mathcal{U}_1(x) \cap \mathcal{U}_2(x), \sigma(x)),$$

$$\mu_1(x) = \mu(\mathcal{U}_1(x), \sigma(x)),$$

$$\mu_2(x) = \mu(\mathcal{U}_2(x), \sigma(x)).$$

where $\mathcal{U}_1 : \mathbb{R}^n \rightrightarrows \mathbb{R}^m$ and $\mathcal{U}_2 : \mathbb{R}^n \rightrightarrows \mathbb{R}^m$ are defined as

$$\mathcal{U}_1(x) = \{u \in \mathbb{R}^m \mid \text{Ineq. (4.2) holds}\},$$

$$\mathcal{U}_2(x) = \{u \in \mathbb{R}^m \mid \text{Ineq. (4.3) holds}\}.$$

With above, we are ready to give our main result.

Theorem 4.4.1. ***(Universal Formula for Smooth Safe Stabilization).*** *Let $V : \mathbb{R}^n \to \mathbb{R}$ and*

$h : \mathbb{R}^n \to \mathbb{R}$ be a 0-CLF and a CBF that are compatible for the system (4.1). *Also let $\sigma : \mathbb{R}^n \to \mathbb{R}$ be a C^l positive definite function. Define*

$$k(x) = s(\rho(x))(\mu_1(x) + \mu_2(x)) + [1 - s(\rho(x))]\mu_{[1,2]}(x) \tag{4.6}$$

where $\rho(x) = \frac{L_g V(x) L_g h(x)^\top}{\|L_g V(x)\| \|L_g h(x)\|}$. If the following hold,

(i) $L_f V$, $L_g V$, $L_f h$, $L_g h$, and β are C^l on $\mathbb{R}^n \setminus (\mathcal{D} \cup \{0\})$;

(ii) $L_g h(x) = 0_m \implies L_f h(x) < \beta(-h(x))$,

then k is C^l on $\mathbb{R}^n \setminus (\mathcal{D} \cup \{0\})$. In addition, the feedback control $u = k(x)$ satisfies both (4.2) *and* (4.3) *for each $x \in \mathbb{R}^n \setminus (\mathcal{D} \cup \{0\})$ and hence, global asymptotically safely stabilizes the closed-loop system* (4.1). *Furthermore, if small control property holds with $\mathcal{U}_1 \cap \mathcal{U}_2$, the function k is also continuous at the origin.* □

Proof. For this proof, we omit the dependency on x for each function for brevity when it is clear . Also, for convenience of the proof, when $\|L_g V\| \neq 0$ and $\|L_g h\| \neq 0$, we rewrite the inequalities (4.2) and (4.3) as

$$c_1^\top u > d_1,\ c_2^\top u \geq d_2$$

where now the coefficients of u are unit vectors

$$c_1 = -\frac{L_g V}{\|L_g V\|},\ c_2 = -\frac{L_g h}{\|L_g V\|}$$

and d_1 and d_2 are the bounds scaled appropriately,

$$d_1 = \frac{L_f V}{\|L_g V\|},\ d_2(x) = \frac{L_f h - \beta(-h)}{\|L_g V\|}.$$

We begin the proof of smoothness by considering the single-input case, $m = 1$. In this case, $\rho = \text{sign}(c_1 c_2)$ can only evaluate to ± 1 or 0. Note that $k = \mu_1 + \mu_2$ when $\rho = 1$, which is smooth as proven in Proposition 4.3.1. For $\rho = -1$,

$$k = \mu_{[1,2]} = \sigma c_1 \frac{p(d_1/\sigma) - p(d_2/\sigma)}{\Phi(d_2/\sigma) - \Phi(d_1/\sigma)}$$

where p and Φ are the Gaussian probability and cumulative density function. Therefore, k is also smooth on $\rho = -1$. Next, we check for the case $\rho = 0$. This implies that either $\|L_g V\|$ or $\|L_g h\|$ is zero, so either $\mathcal{U}_1 = \mathbb{R}$ or $\mathcal{U}_2 = \mathbb{R}$. As a result, k is either μ_1 or μ_2 on $\rho = 0$, both of which are smooth. Finally we check the transition between these 3 cases. One can use $L_g V \to 0 \implies L_f V < 0$ from compatibility, and verify that $\mu_1 \to 0$ and $\mu_{[1,2]} \to \mu_2$ as a function. In addition, all derivatives with respect to c_1 and d_1 approach zero because p and Φ are flat at $d_1 = -\infty$. Similar argument for smoothness of k applies when $L_g h \to 0$. As a result, k is C^l with respect to x for the single-input case.

Now we prove for the cases $m \geq 2$. We note again that μ_1 and μ_2 are smooth, and we will be focusing on showing the smoothness of $\mu_{[1,2]}$. We consider different regions in $\mathbb{R}^n \setminus (\mathcal{D} \cup \{0\})$ with the following conditions:

- $\|L_g V(x)\| = 0$,
- $\|L_g h(x)\| = 0$,

- $\rho(x) = -1$,
- $\rho(x) \neq 0$, $L_g V(x) \neq 0$, and $L_g h(x) \neq 0$.

For the first three items, the function $\mu_{[1,2]}$ can be given with the closed form solution in the single input case. As shown earlier, $\mu_{[1,2]}$ is smooth for all x in those regions. Next, for the last item on the list, we write $C = \begin{bmatrix} c_1 & c_2 \end{bmatrix}$, and have the following closed form solution (cf. [Tal61, Tal65]),

$$\mu_{[1,2]} = \frac{\sigma C}{m} \begin{bmatrix} p(d_1/\sigma)(1 - \Phi(q_{12})) \\ p(d_2/\sigma)(1 - \Phi(q_{21})) \end{bmatrix},$$

$$q_{12} = \frac{d_2 - \rho d_1}{\sigma(1-\rho^2)},\ q_{21} = \frac{d_1 - \rho d_2}{\sigma(1-\rho^2)},$$

where m is the total mass,

$$\begin{aligned} m &= \int_{\mathcal{V}_1 \cap \mathcal{V}_2} \exp(-u^\top u/(2\sigma^2)) du \\ &= 1 - \Phi\left(\frac{d_1}{\sigma}\right) - \Phi\left(\frac{d_2}{\sigma}\right) + \Phi^{(2)}\left(\frac{d_1}{\sigma}, \frac{d_2}{\sigma}\right) \end{aligned}$$

with $\Phi^{(2)}$ is the joint cumulative function with zero mean and the covariance matrix $C^\top C = \begin{bmatrix} 1 & \rho \\ \rho & 1 \end{bmatrix}$. From the given solution, we deduce that $\mu_{[1,2]}$ is smooth for all x inside the fourth region.

We examine next the smoothness at x on the boundary of the listed regions, i.e., where one condition changes to another. The transition between the first three items is equivalent to the transition in the single input cases which we have already shown the smoothness. What remains to

consider is the transition between the fourth item to each of the rest. Consider first when $\|L_g V\| \to 0$, then $d_1 \to -\infty$ due to property of CLF. Because p and Φ are flat at $\pm\infty$, it is clear that at x where this transition occurs,

$$\lim_{\|L_g V\| \to 0} \mu_{[1,2]} \equiv \frac{\sigma c_2}{1 - \Phi(d_1/\sigma)} p(d_2/\sigma) = \mu_2.$$

Similarly, because of the assumption $L_g h(x) = 0_m \implies L_f h(x) < \beta(-h(x))$, we can reason that $\|L_g h\| \to 0 \implies \mu_{[1,2]} \equiv \mu_1$. Next, we look at the case $\rho \to -1$. This implies $c_2 \to -c_1$, and from compatibility condition, we can derive that $d_1 < -d_2$. As a result, q_{12} and q_{21} approach $-\infty$. We then find that

$$\lim_{\rho \to -1} \mu_{[1,2]} \equiv \sigma c_1 \frac{p(d_1/\sigma) - p(d_2/\sigma)}{\Phi(d_2/\sigma) - \Phi(d_1/\sigma)},$$

which is precisely the closed form formula for the single input case $m = 1$.

To summarize, we have shown so far the smoothness of $\mu_{[1,2]}$ at x such that $\rho(x) \neq 1$. Because μ_1 and μ_2 are smooth, k is then smooth where $\rho(x) \neq 1$. For $\rho = 1$, we note that $k = \mu_1 + \mu_2$, so the function is smooth there. Last, we show the smoothness of k at x where $\rho \neq 1$ transits to $\rho = 1$. In this case, q_{12} and q_{21} have opposite signs and approach $\pm\infty$. $\mu_{[1,2]}$ then approach either μ_1 or μ_2 which are smooth and all the derivative are bounded. Consequently, $[1 - s(\rho)]\mu_{[1,2]} \equiv 0$, so the smoothness of k holds. In conclusion, we have shown for all cases and all transition points that k is smooth.

We move on to proving the satisfaction of (4.2) and (4.3). First, we note that $\mu_{[1,2]}(x)$ satisfies both inequalities for all x due to it being a weighted centroid of the convex set $\mathcal{U}_1(x) \cup \mathcal{U}_2(x)$, and $k = \mu_{[1,2]}$ when $\rho < 0$. For $\rho \geq 0$, we write out the solution,

$$\mu_i = \sigma c_i \frac{p(d_i/\sigma)}{1 - \Phi(d_i/\sigma)}$$

when $\|L_g V\| \neq 0$, $\|L_g h\| \neq 0$ for $i = 1, 2$ resp., and otherwise $\mu_i = 0$. Because the fraction is always positive, we have $c_1\mu_2 \geq 0$ and $c_2\mu_1 \geq 0$ when $\rho = c_1^\top c_2 \geq 0$. As a consequence, (4.2) and (4.3) hold with $u = \mu_1(x) + \mu_2(x)$ for $\rho(x) \geq 0$. Because the set $\mathcal{U}_1(x) \cup \mathcal{U}_2(x)$ is convex, $u = k(x)$ satisfies both inequalities at all x.

Finally, the continuity of the controller at the origin under small control property is reasoned the same way as in the proof of Proposition 4.3.1. □

Theorem 4.4.1 gives a constructive formula for a strict compatible pair of 0-CLF and CBF. The control feedback given by (4.6) will achieve smooth safe stabilization. The main idea behind the construction of the controller is to exploit weighted centroid $\mu_{[1,2]}$, which is already smooth almost everywhere. Note this is done by examining the closed form solution (cf. [Tal61, Tal65]). For the place where it is not smooth, we can conveniently transition the controller into $\mu_1 + \mu_2$ because both $\mu_{[1,2]}$ and $\mu_1 + \mu_2$ become either μ_1 or μ_2 there. Using the property of the weighted centroid, the proposed controller can satisfy both control inequality constraints.

Remark 4.4.2. ***(Generalization to Control-Affine Inequality Constraints).*** *We stated Theorem 4.4.1 for the control inequality constraints given by a* 0*-CLF and a CBF. However, the proposed controller also applies for satisfying two different control-affine inequality constraints as long as strict compatibility holds. Application to a pair of a* 0*-CLF and a CBF is simply a special yet very important case.* •.

4.5 Discussion on δ and compatibility

In this section, we have a further discussion on compatibility and how we can obtain it through using a δ-CLF. Our results rely heavily on the compatibility of the given 0-CLF and CBF.

Here, we consider the scenario where compatibility does not hold for the 0-CLF and the CBF with the property $L_g h(x) = 0 \implies L_f h < 0$, i.e., there exists an x where there exists no u satisfying both inequalities (4.2) and (4.3). Particularly, we can reason with contraposition that this scenario must occur when the unsafe set $\mathcal{D}$ is bounded because there can be no smooth feedback control as proven in [BK17]. In any case, the question remains: what can be done when compatibility does not hold.

To answer the question, we review the literature to see how the problem of incompatibility is dealt. We recall from the literature the minimum-norm controller (cf. [AXGT17, XTGA15]). Given a 0-CLF and a CBF, a minimum-norm controller $k_{\min} : \mathbb{R}^m \to \mathbb{R}^n$ is computed through solving pointwise the following quadratic programming

$$\begin{aligned}\begin{bmatrix} k_{\min}(x) & \delta(x) \end{bmatrix}^\top = \underset{\tilde{u}=[u^\top, w]^\top \in \mathbb{R}^{m+1}}{\operatorname{argmin}} \quad & \tilde{u}^\top \tilde{u} \\ \text{s.t. } L_f V(x) + \begin{bmatrix} L_g V(x) & -1 \end{bmatrix} \tilde{u} < 0, \\ L_g h(x) + \begin{bmatrix} L_g h(x) & 0 \end{bmatrix} \tilde{u} \leq \beta(-h(x)).\end{aligned}$$

Notice that the first inequality constraint is no longer inequality (4.2) from the 0-CLF because of the relaxation input w. Instead, it represents the inequality (4.2) associated with a δ-CLF. Thus, stability of the origin is no longer guaranteed. The reason behind this sacrifice is to guarantee the feasibility of the quadratic programming. By introducing a relaxation input w, there exists a $\tilde{u}$ satisfying the inequality constraints for each x because the coefficient of $\tilde{u}$ from the two inequalities are always linearly independent.

We can integrate the idea of introducing a relaxation input to obtain compatibility into our

universal formula. First, we redefine $\mathcal{U}_1$ and $\mathcal{U}_2$ appropriately with the relaxation input

$$\mathcal{U}_1(x) = \{\tilde{u} \mid L_f V(x) + \begin{bmatrix} L_g V(x) & -1 \end{bmatrix} \tilde{u} < 0\},$$
$$\mathcal{U}_2(x) = \{\tilde{u} \mid L_f h(x) + \begin{bmatrix} L_g h(x) & 0 \end{bmatrix} \tilde{u} \leq \beta(-h(x))\}.$$

where $\tilde{u} = \begin{bmatrix} u & w \end{bmatrix}^\top \in \mathbb{R}^{m+1}$. With $\mu_{[1,2]}$, μ_1, and μ_2 redefined with the above set-valued functions, we find a smooth control feedback,

Proposition 4.5.1. ***(Exploiting δ for Compatibility).*** *Let the system* (4.1) $V : \mathbb{R}^n \to \mathbb{R}$ *and* $h : \mathbb{R}^n \to \mathbb{R}$ *be a* $0 - CLF$ *and a CBF that are not necessarily compatible. If the following assumptions hold,*

(i) $L_f V$, $L_g V$, $L_f h$, $L_g h$, *and* β *are* C^l*;*

(ii) $L_g h(x) = 0 \implies L_f h(x) < \beta(-h(x))$*,*

then with a C^l *positive definite function* $\sigma : \mathbb{R}^n \to \mathbb{R}$*, and* ρ *redefined as*

$$\rho(x) = \frac{\begin{bmatrix} L_g V(x) & -1 \end{bmatrix} \begin{bmatrix} L_g h(x) & 0 \end{bmatrix}^\top}{\| \begin{bmatrix} L_g V(x) & -1 \end{bmatrix} \| \| \begin{bmatrix} L_g h(x) & 0 \end{bmatrix} \|},$$

the function k *defined as in* (4.6) *is* C^l *on* $\mathbb{R}^n \setminus (\mathcal{D} \cup \{0\})$*. In addition, the control feedback* $u = \begin{bmatrix} I_m & 0_m \end{bmatrix} k(x)$ *satisfies the inequality* (4.3) *for each* $x \in \mathbb{R}^n \setminus (\mathcal{D} \cup \{0\})$ *and hence, guarantees the safety of the trajectories for the closed-loop system* (4.1)*. Also, the control feedback* $u = \begin{bmatrix} I_m & 0_m \end{bmatrix} k(x)$ *satisfies the inequality* (4.3) *associated with the* δ*-CLF where the relaxation function given by* $\delta(x) = \begin{bmatrix} 0_m^\top & 1 \end{bmatrix} k(x)$*, for each* $x \in \mathbb{R}^n \setminus (\mathcal{D} \cup \{0\})$*. Furthermore, if small control*

property holds with $\mathcal{U}_1 \cap \mathcal{U}_2$, the function k is also continuous at the origin. □

Proof. We begin by proving the compatibility of the inequalities associated with the set $\mathcal{U}_1$ and $\mathcal{U}_2$. Note that $\begin{bmatrix} L_g V(x) & -1 \end{bmatrix}$ and $\begin{bmatrix} L_g h(x) & 0 \end{bmatrix}$ are linearly independent when $L_g h(x) \neq 0$, so there must exist a $\tilde{u}$ satisfying both inequalities. In addition, when $L_g h(x) = 0$, because we have that $L_f h < \beta(-h(x))$, any $\tilde{u}$ in the set $\mathcal{U}_1$ holds for both inequalities. As such, compatibility on $\mathbb{R}^n \setminus \mathcal{D}$ holds. Next, because compatibility holds and $L_g h(x) = 0 \implies L_f h(x) < \beta(-h(x))$, we can show smoothness of k by following the proof of Theorem 4.4.1. Doing so, we can also show that $\tilde{u} = k(x)$ belongs to $\mathcal{U}_2(x)$ for each $x \in \mathbb{R}^n \setminus (\mathcal{D} \cup \{0\})$. We can deduce the following,

$$L_f h(x) + \begin{bmatrix} L_g h(x) & 0 \end{bmatrix} k(x) \leq \beta(-h(x))$$
$$L_f h(x) + L_g h(x) \begin{bmatrix} I_m & 0_m \end{bmatrix} k(x) \leq \beta(-h(x)).$$

As such, $u = \begin{bmatrix} I_m & 0_m \end{bmatrix} k(x)$ satisfies (4.3) for each $x \in \mathbb{R}^n \setminus (\mathcal{D} \cup \{0\})$ and the proof concludes. □

Proposition 4.5.1 provides a feedback controller for when the system (4.1) admits a CLF and a CBF that are not compatible. The idea is to introduce a relaxation input w and then apply the universal formula. As a result, we can find that the new set of inequalities are compatible. The downside of introducing a relaxation input is that we no longer guarantee the stability of the origin. We satisfy instead the condition for a δ-CLF. However, because δ is a smooth function, it will be upper bounded on any compact domain. Regardless, it must be noted again that stability is no longer guaranteed. Nevertheless, this method eliminates the need to search for a compatible δ-CLF and has been proven useful in the literature.

4.6 Numerical Example

In this section, we apply our results to an example. Consider a unicycle dynamics subjected to a drift with the following dynamics,

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} 0 \\ -y \\ 0 \end{bmatrix} + \begin{bmatrix} \cos\theta & 0 \\ \sin\theta & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix}.$$

For the states, we write $z = \begin{bmatrix} x & y & \theta \end{bmatrix}^\top$. One can check that $V(z) = \frac{1}{2}\|z\|^2$ is a 0-CLF. Next, suppose unsafe states are given by the set $\mathcal{D} = \left\{ z \in \mathbb{R}^3 \mid y > (2x+1)^2 + 1 \right\}$. We use the following CBF candidate.

$$h(z) = y - (2x+1)^2 - 1$$

We find that $h(z) = 0$ on the boundary of the unsafe set and $h(z) < 0$ on $\mathbb{R}^n \setminus \mathcal{D}$. Next we check if the function is a CBF by checking if there exists a u satisfying (4.3) for each x. We pick the simplest class-$\mathcal{K}$ function for $\beta(-h(z)) = -kh(z)$ where k is a positive constant. Now we evaluate,

$$L_f h(z) + L_g h(z) u = -y - 4(2x+1)u\cos\theta + u\sin\theta.$$

For $k \geq 1$, we find that

$$
\begin{aligned}
-y &\leq \begin{cases} -ky, & y < 0 \\ 0, & 0 \leq y < (2x+1)^2 + 1 \end{cases} \\
&< \begin{cases} -ky + k((2x+1)^2 + 1), & y < 0 \\ -k(y - (2x+1)^2 - 1), & 0 \leq y < (2x+1)^2 + 1 \end{cases} \\
&< -kh(z), \ \forall z \in \mathbb{R}^3 \setminus \mathcal{D}.
\end{aligned}
$$

Therefore, for $k \geq 1$, $u = 0$ satisfies (4.3) on $\mathbb{R}^3 \setminus \mathcal{D}$, and h is a CBF. We pick $k = 5$ for the simulation. Also note here that with $L_f h(z) < -kh(z)$, we can immediately satisfy the assumption $L_g h(z) = 0 \implies L_f h(z) < -kh(z)$.

Next, we examine the compatibility of the 0-CLF and CBF. First, it is clear that $u = 0$ satisfy both (4.2) and (4.3) for $y \neq 0, \ y \notin \mathcal{D}$. Then for $y = 0$, we can find that

$$
\begin{aligned}
L_g V(z) &= \begin{bmatrix} x\cos\theta & \theta \end{bmatrix} \\
L_g h(z) &= \begin{bmatrix} -(8x+4)\cos\theta + \sin\theta & 0 \end{bmatrix}.
\end{aligned}
$$

We only need to consider when these two vectors are linear dependent because otherwise there always exists a control that can satisfy both (4.2) and (4.3). As such, we consider when $\theta = 0$. For compatibility, we need a u satisfying

$$
xu < 0, \ -(8x+4)u \leq 0
$$

In other words, we want $0 < (7x - 4)u$. Clearly, a u with the same sign as $7x - 4$ exists, and we can pick it arbitrarily small, so that small control property holds.

With initial conditions of $\begin{bmatrix} -1 & 2 & \pi \end{bmatrix}^\top$ and $\begin{bmatrix} 0 & 2 & \pi \end{bmatrix}^\top$, we simulate our proposed controller given by (4.6) with $\sigma(z) = 1 - \exp(z^\top z)$. The resulting trajectory is shown in Figure 4.1 in a thin solid curve. In comparison to Sontag's universal formula, given in dashed curve, our proposed controller results in a trajectory that avoids the unsafe states as predicted. In addition, the minimum-norm controller discussed Section 4.5 in is plotted in a dotted line for comparison. For this controller, a relaxation input is introduced to guarantee Lipschitzness of the controller. Also, we add a negative definite function, $-0.1z^\top z$, on the right hand side of the CLF inequality (4.2) to "force" control effort; otherwise, the minimum-norm controller will be identically zero. For its plot, although the trajectory appears to converge towards the origin, there is no real guarantee that it will do so. This is not to mention that the controller is not differentiable at some point along the trajectory, which can be an undesirable property. In contrast, the control signals of our proposed formula are plotted in Figure 4.2. As guaranteed by Theorem 4.4.1, the signals are smooth and go to zero

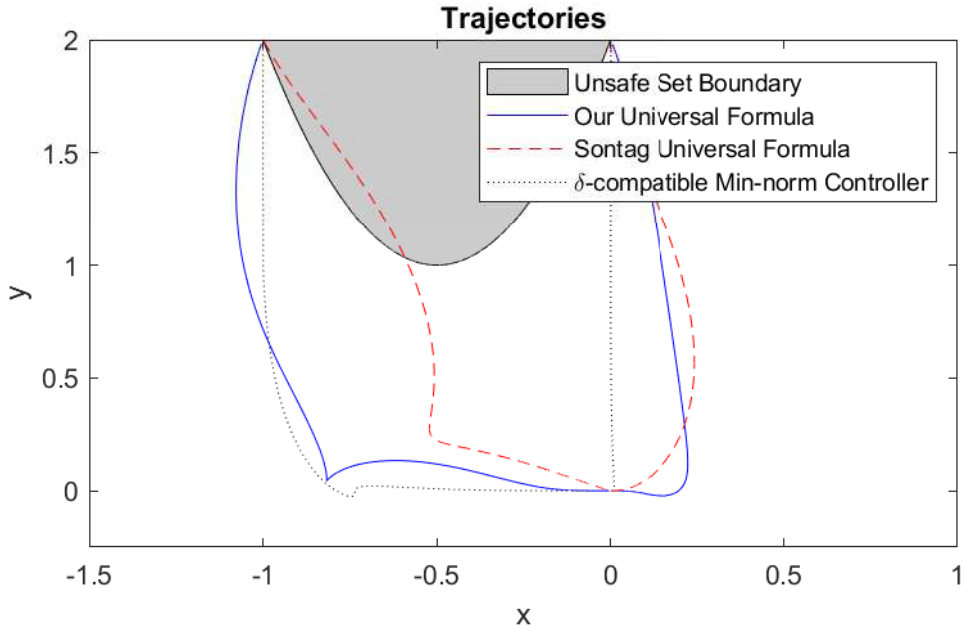


Figure 4.1: Trajectories for different types of controllers with two different initial conditions. Using Sontag's universal feedback formula results in a trajectory that violates the state constraint because it does not take safety into account. Both the minimum-norm and our proposed controller produce safe trajectories that progress towards the origin (however, there is no guarantee that the minimum-norm controller will reach it).

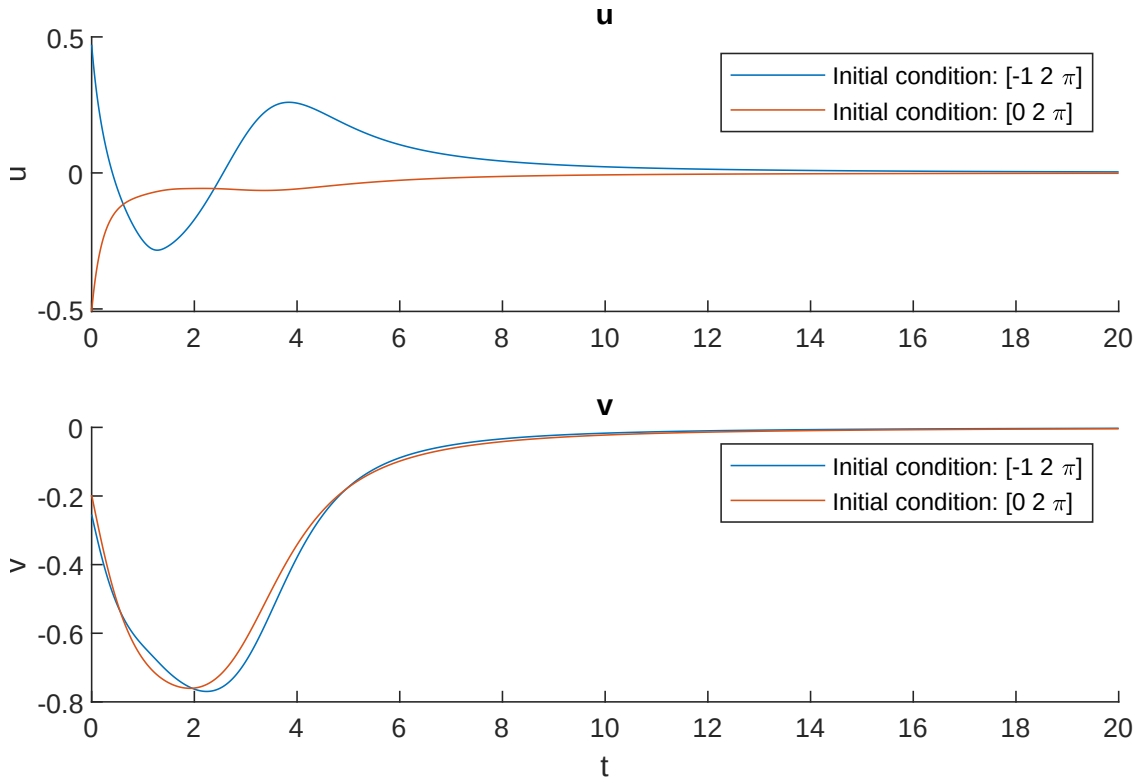


Figure 4.2: The control inputs along trajectory using our control feedback formula (4.6) with the two different initial conditions. Both control inputs appear smooth in simulation as predicted. In addition, because the small control property holds, the control both converges to zero as the state converges towards the origin.

Acknowledgements

This chapter, in part, is a reprint of the material [OC19] as it appears in 'Universal formula for smooth safe stabilization' by P. Ong and J. Cortés, in IEEE Conference on Decision and Control, 2019. The dissertation author was the primary investigator and author of this paper. The work in this chapter was partially supported by NSF Award CNS-144689.

Part II

Resource-Aware Implementation of Control Laws

Chapter 5

Improved Event-Triggered Control

This chapter proposes a novel framework for resource-aware control design termed performance-barrier-based triggering. Given a feedback policy, along with a Lyapunov function certificate that guarantees its correctness, we examine the problem of designing its digital implementation through event-triggered control while ensuring a prescribed performance is met and triggers occur as sparingly as possible. Our methodology takes into account the *performance residual*, i.e., how well the system is doing in regards to the prescribed performance. Inspired by the notion of control barrier function, the trigger design allows the certificate to deviate from monotonically decreasing, with leeway specified as an increasing function of the performance residual, resulting in greater flexibility in prescribing update times. We study different types of performance specifications, with particular attention to quantifying the benefits of the proposed approach in the exponential case. We build on this to design intrinsically Zeno-free distributed triggers for network systems. A comparison of event-triggered approaches in a vehicle platooning problem shows how the proposed design meets the prescribed performance with a significantly lower number of controller updates.

5.1 Problem Formulation

Consider a nonlinear control system of the form

$$\dot{x} = F(x, u),\ x \in \mathbb{R}^n,\ u \in \mathbb{R}^m,$$

with $F : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^n$. The digital implementation of a desired feedback policy $\kappa : \mathbb{R}^n \to \mathbb{R}^m$ as $u = \kappa(x)$ can be accomplished through a sample-and-hold strategy. This consists of updating the control signal at a specific time t_k, for $k \in \{0\} \cup \mathbb{N}$, and keeping it constant up until t_{k+1}, when the evaluation of the feedback policy provides the next adjustment. As a result, the closed-loop system is

$$\dot{x} = F(x, \kappa(x + e)) = f(x, e), \tag{5.1}$$

where the error $e = x_k - x$ is the state deviation from the last update at iteration k (here, we use the shorthand notation $x_k = x(t_k)$). The challenge is then how to prescribe the sequence of update times $\{t_k\}$ in order to ensure that the digital implementation retains the convergence and performance properties of the original continuous-time system.

Event-triggered control looks past time-periodic implementations to identify a state-dependent trigger criterion to determine the update times. To come up with such a criterion for a general nonlinear system, a common starting point is to assume that there exists an Input-to-State Stability (ISS) Lyapunov function for (5.1), see e.g., [Tab07, APDN16, PTNA15]. Formally, we assume there exists a smooth function $V : \mathbb{R}^n \to \mathbb{R}$ and class-$\mathcal{K}_\infty$ functions $\underline{\alpha}$, $\overline{\alpha}$, α, and γ satisfying

$$\underline{\alpha}(\|x\|) \leq V(x) \leq \overline{\alpha}(\|x\|), \tag{5.2a}$$

$$\mathcal{L}_f V(x, e) \leq -\alpha(\|x\|) + \gamma(\|e\|). \tag{5.2b}$$

The seminal work [Tab07] provides the trigger design

$$t_{k+1} = \left\{ t \geq t_k \mid -\sigma\alpha(\|x(t)\|) + \gamma(\|e(t)\|) = 0 \right\}, \tag{5.3}$$

with design parameter $\sigma \in (0, 1)$. Under (5.3), the rate of change of the Lyapunov function along (5.1) satisfies

$$\frac{d}{dt} V(x(t)) \leq (\sigma - 1)\alpha(\|x(t)\|).$$

Therefore, by design, the certificate V decreases along the trajectories of the sample-and-hold implementation. Stability cannot be established from this fact alone, however, due to the possibility of Zeno behavior: the state-dependency of the trigger criterion makes it possible for the inter-event time between consecutive updates to become increasingly small. This, in turn, leaves open the possibility of an infinite number of updates within a finite period of time. A common strategy to rule out Zeno behavior is to establish the existence of a minimum inter-event time (MIET). For the trigger design (5.3), the existence of a MIET can be established under mild assumptions, cf. [Tab07].

Triggering according to state-triggered criteria like (5.3) might lead to fewer controller updates than a time-triggered implementation at the cost of impacting performance (as measured, for

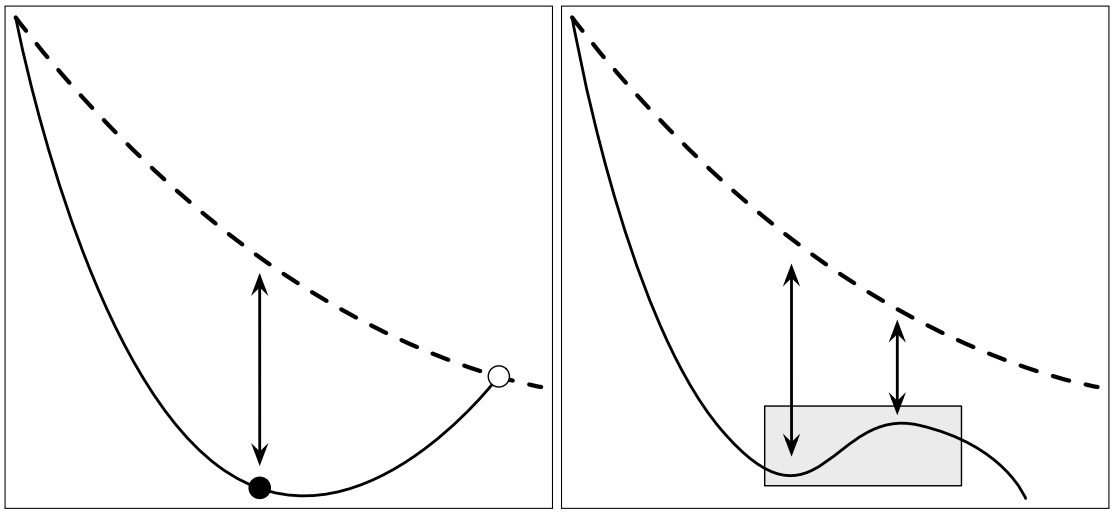


Figure 5.1: Prescribed performance (dashed line) and evolution of the certificate (solid line) under state-dependent triggering. (a) the controller update (black circle) prescribed by (5.3) does not take into account the performance residual, which would otherwise be positive until the curve of the certificate meets the prescribed performance (empty circle). (b) a possible evolution of the certificate that momentarily violates (gray area) the derivative condition on the certificate specified by (5.3), does not require a controller update while always meeting the performance specification.

instance, by the rate of decrease of the certificate V). Ideally, one would like the system to trigger as sparingly as possible while still guaranteeing a prescribed performance regarding convergence. In that regard, (5.3) tends to overprescribe updates, as the criterion looks exclusively at the derivative of the certificate without taking into account how much the certificate has decreased since the last update, cf. Figure 5.1(a). We refer to the difference between the prescribed performance and the value of the certificate as the *performance residual*. Presumably, allowing the certificate to momentarily violate the derivative condition, with leeway specified as an increasing function of the performance residual, could result in executions with even fewer controller updates that still meet the performance requirements, cf. Figure 5.1(b). In the context of network systems, the overprescription of controller updates is also related to the fact that the design of distributed event-triggered schemes based on (5.3) might result, in general, in sample-and-hold implementations that do not

have a MIET, see [HJT12, DH12, TC14, BH14].

The formalization of the ideas described above leads us to propose the *performance-barrier-based* design methodology for trigger design. In Section 5.2, we limit our discussion to linear systems to motivate and introduce the basic idea. We develop it further for general nonlinear systems in Section 5.3. As we show in our exposition, the new approach naturally leads to longer inter-event times while meeting the specified performance. This provides the necessary groundwork for tackling the design of Zeno-free distributed event-triggered schemes for network systems in Section 5.4.

5.2 Performance-Barrier-Based Event-Triggered Control Designs for Linear Systems

Here we introduce the performance-barrier-based ETC framework. In this section, we limit our discussion to linear systems for simplicity of exposition. Consider the sample-and-hold linear control system

$$\dot{x} = Ax + BKx_k = (A + BK)x + BKe, \tag{5.4}$$

with matrices $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times m}$ and $K \in \mathbb{R}^{m\times n}$ so that $A + BK$ is Hurwitz. In this case, it is easy to guarantee the existence of an ISS Lyapunov function satisfying (5.2). In fact, using the fact that $A + BK$ is Hurwitz, there exists positive definite matrices P and Q such that

$$V(x) = x^\top Px \tag{5.5a}$$

is an ISS Lyapunov function with

$$\begin{aligned}\mathcal{L}_f V(x,e) &= -x^\top Q x + 2x^\top PBKe \\ &\leq \left(\frac{\|PBK\|}{\theta} - \lambda_{\min}(Q)\right)\|x\|^2 + \theta\|PBK\|\|e\|^2 \\ &:= -c_\alpha\|x\|^2 + c_\gamma\|e\|^2, \end{aligned} \tag{5.5b}$$

where $\lambda_{\min}(Q)$ is the minimum eigenvalue of Q and Young's inequality [HLP52] is applied with $\theta > 0$ selected appropriately so that c_α, c_γ are positive. In particular, for the original continuous-time system ($e \equiv 0$ in (5.4)), one obtains the performance guarantee

$$V(x(t)) \leq V(x_0)\exp\left(c_\alpha\|P\|^{-1}t\right), \tag{5.6}$$

where x_0 denotes the initial condition. We next turn to the trigger design.

5.2.1 Derivative- and Function-Based Trigger Designs

For the sample-and-hold linear system (5.4), the derivative-based trigger design (5.3) takes the form

$$t_{k+1} = \min\left\{t \geq t_k \mid -\sigma c_\alpha\|x\|^2 + c_\gamma\|e\|^2 = 0\right\},$$

with the certificate along any trajectory satisfying $\frac{d}{dt}V(x(t)) \leq (\sigma - 1)c_\alpha \|x(t)\|^2$. Using this inequality, the evolution of the certificate satisfies

$$V(x(t)) \leq V(x_0) \exp\left((\sigma - 1)c_\alpha \|P\|^{-1} t\right). \tag{5.7}$$

A higher value of $\sigma \in (0, 1)$ results in a longer inter-event time and a slower exponential rate on the evolution of the certificate. This presents a trade-off for design. In order to compare different designs fairly, it would seem reasonable to establish a common performance criterion. Given the exponential convergence characteristic of linear systems, prescribing a desired rate of convergence $r > 0$ is a natural candidate. Formally, we specify

$$V(x(t)) \leq V(x_0) \exp(-rt), \tag{5.8}$$

at all time and for any initial condition. Given the performance (5.6) of the continuous state-feedback system, we require $r < c_\alpha \|P\|^{-1}$. Since the derivative-based trigger is guaranteed to perform according to (5.7), one can see that $\sigma = 1 - \frac{r\|P\|}{c_\alpha}$ is the value that yields the longest inter-event time (for the derivative-based design) while still satisfying the performance specification. The following result summarizes the asymptotic convergence properties under the derivative-based trigger design.

Lemma 5.2.1. ***(Derivative-Based Design – Linear Case).*** *Consider the sample-and-hold linear system* (5.4) *with an ISS Lyapunov function* (5.5)*. Given a desired rate of convergence* $r < c_\alpha \|P\|^{-1}$

and $\sigma \in (0, 1 - \frac{r\|P\|}{c_\alpha})$*, let* $g : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$ *be any function such that*

$$\mathcal{L}_f V(x, e) \leq g(x, e) \leq (\sigma - 1)c_\alpha \|x\|^2 + c_\gamma \|e\|^2.$$

Define the derivative-based trigger time as

$$t_{k+1}^{\mathrm{d}} = \min \left\{ t \geq t_k \mid g(x(t), e(t)) + rV(x(t)) \geq 0 \right\}. \tag{5.9}$$

There exists a MIET $\tau_\sigma^{\mathrm{d}} > 0$ *such that if* $V(x(t_k)) \leq V(x_0)\exp(-rt_k)$*, then* $t_{k+1}^{\mathrm{d}} - t_k \geq \tau_\sigma^{\mathrm{d}}$*. As a consequence, if the trigger sequence* $\{t_k\}_{k=0}^{\infty}$ *is defined iteratively via the derivative-based trigger, then* $V(x(t)) < V(x_0)\exp(-rt)$ *for all* $t > 0$*, and the origin is globally exponentially stable.* ■

Lemma 5.2.1 is essentially presented in [Tab07]. We omit its proof as it is a special case of Proposition 5.2.3 below. The basic idea behind the design (5.9) is to keep the time derivative of the Lyapunov function below an amount that, by application of the Comparison Lemma [Kha02, Lemma 3.4], would make the system satisfy the desired performance, i.e., $\frac{d}{dt}V(x(t)) < -rV(x(t))$. As a result, the gap $V(x_0)\exp(-rt) - V(x(t))$ between the desired performance and the Lyapunov function, which we call *performance residual*, is always increasing until the next update, see Figure 5.1. While meeting the desired specifications means keeping the performance residual nonnegative, doing so by having it always increase is overly conservative. To produce a less conservative design, one can instead look at the value of the Lyapunov function itself (rather than its time derivative), as specified in the following result.

Lemma 5.2.2. ***(Function-Based Design – Linear Case).*** *Consider the sample-and-hold linear system* (5.4) *with an ISS Lyapunov function* (5.5)*. Given a desired rate of convergence* $r < c_\alpha / \|P\|$*,*

define the function-based trigger time as

$$t_{k+1}^{\mathrm{f}} = \min\left\{t > t_k \mid 0 \geq V(x_0)\exp(-rt) - V(x(t))\right\}. \tag{5.10}$$

There exists a MIET $\tau_r^{\mathrm{f}} > 0$ *such that if* $V(x(t_k)) \leq V(x_0)\exp(-rt_k)$*, then* $t_{k+1}^{\mathrm{f}} - t_k \geq \tau_r^{\mathrm{f}}$*. As a consequence, if the trigger sequence* $\{t_k\}_{k=0}^{\infty}$ *is defined iteratively via the function-based trigger, then* $V(x(t)) \leq V(x_0)\exp(-rt_k)$*, and the origin is globally exponentially stable.* ■

The function-based design relies on the idea of directly enforcing $V(x(t)) \leq V(x_0)\exp(-rt)$. A problem with this design, however, is that it waits until the last moment, i.e., when the performance residual becomes zero (empty circle in Fig. 5.1(a)), to prescribe a controller update. Consequently, the implementation is not robust to errors (e.g., delays in evaluation or actual implementation). The performance-barrier-based trigger design, proposed next, is motivated by the idea of overcoming the conservatism of the derivative-based design and the lack of robustness of the function-based one.

5.2.2 Performance-Barrier-Based Trigger Design

Our ensuing design builds on the observation that to ensure that the evolution of V satisfies the specified performance, V needs to decrease faster than (or at the same rate as) the specification only when their values are equal. Formally, this can be established using Nagumo theorem [BM07]: $V(x(t)) \leq V(x_0)\exp(-rt)$ if and only if

$$\frac{d}{dt}V(x(t)) \leq -rV(x(t)) \text{ when } V(x(t)) = V(x_0)\exp(-rt). \tag{5.11}$$

Note that this condition does not restrict how fast V changes when $V(x(t)) < V(x_0)\exp(-rt)$, no matter how small the performance residual is. One can readily see that the condition (5.11) suffers from the same lack of robustness as the function-based design. To address this, and inspired by how control barrier functions [AXGT17, ACE+19] restrict the speed of their own evolution as the state approaches the boundary of the safe set, we instead prescribe

$$\frac{d}{dt}V(x(t)) + rV(x(t)) \leq c_\beta\big(V(x_0)\exp(-rt) - V(x(t))\big),$$

with a nonnegative constant $c_\beta \geq 0$. The key idea is restricting how fast V can increase proportionally to the performance residual. The following result summarizes the asymptotic convergence properties under this type of prescription.

Proposition 5.2.3. ***(Performance-Barrier-Based Design – Linear Case).*** *Consider the sample-and-hold linear system* (5.4) *with an ISS Lyapunov function* (5.5)*. Given a desired rate of convergence $r < c_\alpha/\|P\|$ and $\sigma \in (0, 1 - \frac{r\|P\|}{c_\alpha})$, let g be as in Lemma 5.2.1. Define the performance-barrier-based trigger time as*

$$t^{\mathrm{p}}_{k+1} = \min\left\{t \geq t_k \mid g(x(t), e(t)) + rV(x(t)) \geq c_\beta\big(V(x_0)\exp(-rt) - V(x(t))\big)\right\}. \tag{5.12}$$

Let $G(\tau) = \exp(A\tau) + \int_0^\tau \exp(A(\tau - s))ds\,BK$ and

$$M(\tau) = c_\beta P\exp(-r\tau) - c_\gamma\|\mathbf{I} - G(\tau)\|^2 - G(\tau)^\top((c_\beta + r)P + (\sigma - 1)c_\alpha\mathbf{I})G(\tau). \tag{5.13}$$

The constant

$$\tau_\sigma^{\mathrm{p}} := \min\{\tau > 0 \mid det(M(\tau)) = 0\}, \tag{5.14}$$

is a MIET such that if $V(x(t_k)) \leq V(x_0)\exp(-rt_k)$*, then* $t_{k+1}^{\mathrm{p}} - t_k \geq \tau_\sigma^{\mathrm{p}}$*. As a consequence, if the trigger sequence* $\{t_k\}_{k=0}^{\infty}$ *is defined iteratively via the performance-barrier-based trigger, then* $V(x(t)) \leq V(x_0)\exp(-rt)$ *for all time, and the origin is globally exponentially stable.*

Proof. First, we note that we can derive from the trigger design, $V(x(t)) \leq V(x_0)\exp(-rt)$ for every interval $[t_k, t_{k+1}^{\mathrm{p}})$, but we have omitted the proof here because it will appear in the proof of Proposition 5.3.3 later for the more general case. Nevertheless, we will prove here the result on the MIET, which will rule out the the sequence $\{t_k\}_{k=0}^{\infty}$ converging to a finite value (Zeno behavior). We start by deducing for each update

$$V(x(t_k)) \leq V(x_0)\exp(-rt_k)$$

$$V(x(t_k))\exp(-r\Delta t_k) \leq V(x_0)\exp(-rt)$$

for the time $t \in [t_k, t_{k+1}]$ where $\Delta t_k = t - t_k$. Using this bound to lower bound the right-hand side of the trigger condition in (5.12), as well as using the definition of g to upper bound the left-hand side, we derive the condition

$$x^\top(rP + (\sigma - 1)c_\alpha \mathbf{I})x + c_\gamma \|e\|^2 = c_\beta(V(x_k)\exp(-r\Delta t_k) - x^\top P x) \tag{5.15}$$

which must be met earlier. Note we have replaced inequality with equality due to continuity of all the terms along the trajectory. Under system (5.4), we can find the expression for the state during each iteration as $x(t) = G(\Delta t_k)x_k$. Substituting the state and moving everything of the left-hand

side to the right, (5.15) becomes

$$0 = x_k^\top M(\Delta t_k) x_k.$$

We know that $M(0) \succ 0$ because the right-hand side of (5.15) is zero, and the left-hand side is negative at time t_k due to the definition of r. The MIET is given by when $M(\tau)$ transits from positive definite to semi-positive definite which is when there exists an x_k such that the condition is satisfied. Therefore, the MIET is given by (5.14). As a result, $V(x(t)) < V(x_0)\exp(-rt)$ for all time. Lastly, the origin can be deemed exponentially stable as we can derive

$$\|x\| \leq \|x_0\| \frac{\|P\|^{1/2}}{\lambda_{\min}(P)^{1/2}} \exp(-rt/2),$$

concluding the proof. □

Proposition 5.2.3 generalizes both Lemmas 5.2.1 and 5.2.2. Note that the trigger design (5.9) is recovered by selecting $c_\beta = 0$ in (5.12), and the trigger design (5.10) corresponds to the limit of (5.12) as $c_\beta \to \infty$. Directly from the construction of the trigger designs, one can deduce $t_{k+1}^{\mathrm{d}} \leq t_{k+1}^{\mathrm{p}} \leq t_{k+1}^{\mathrm{f}}$ (inequalities are strict if g is continuous). Therefore, we can adjust the parameter c_β to control the inter-event times, which is also evident in the expression for the MIET. Note that the performance-barrier-based design enjoys longer inter-event times than the derivative-based one while still being able to achieve the prescribed performance. Although the performance-barrier-based strategy does not have a MIET as large as the function-based one, it does not suffer from the same lack of robustness to errors The design also includes the flexibility of using the surrogate function g if it is more convenient or easier to evaluate. Finally, Proposition 5.2.3 also provides a method to calculating the MIET using the design (5.12) for linear control systems. The expression

only depends on time (not on the state), which means that it can be calculated offline.

5.3 Performance-Barrier-Based Event-Triggered Control Designs for Nonlinear Systems

In this section we expand our presentation of the performance-barrier-based event-triggered control design to general nonlinear systems (5.1). Our starting point is the availability of an ISS Lyapunov function (5.2) in tandem with the feedback policy κ. Unlike the case of linear systems, the evolution of the Lyapunov function along the trajectories of the closed-loop system might not be exponentially decaying, and this raises the question of how to suitably define a performance specification. We do this by considering a continuously differentiable, time-dependent function $S(\cdot; x_0) : \mathbb{R}_+ \to \mathbb{R}_+$, parametrized by the initial condition x_0, encoding the desired behavior as

$$V(x(t)) \leq S(t; x_0). \tag{5.16}$$

We use Nagumo theorem [BM07] to write an equivalent condition (assuming that $V(x_0) \leq S(0; x_0)$) to the requirement (5.16) as

$$\frac{d}{dt}V(x(t)) \leq \frac{d}{dt}S(t; x_0) \text{ when } V(x(t)) = S(t; x_0). \tag{5.17}$$

With this in mind, we seek to identify different types of performance specification functions S that allow us to establish the existence of a MIET. In the following, we discuss several classes of specification functions.

5.3.1 Class-$\mathcal{K}$ Derivative Performance Specification

This class of specification function is an extension of the exponential decrease of the linear case. In particular, note that the desired convergence rate r is limited in the linear case by the performance (5.6) of the original continuous-time system. Similarly, in the nonlinear case, we look at the performance under the continuous-time controller implementation ($e \equiv 0$ in (5.1)). Hence, let $h : \mathbb{R}_+ \to \mathbb{R}_+$ be such that

$$\mathcal{L}_f V(x, 0) \leq -\alpha(\|x\|) < -h(V(x)),$$

for all x. In other words, h expects a slower convergence than the natural convergence of the system with a continuous controller.

Definition 5.3.1. ***(Class-$\mathcal{K}$ Derivative Specification).*** For $\sigma^* \in (0, 1)$, let $h : \mathbb{R}_+ \to \mathbb{R}_+$ be locally Lipschitz and class-$\mathcal{K}$ with $h(V(x)) \leq (1-\sigma^*)\alpha(\|x\|)$ for all x. A function $S(\cdot; x_0) : \mathbb{R}_+ \to \mathbb{R}_+$ is a class-$\mathcal{K}$ derivative performance specification if it is the unique solution to the differential equation

$$\dot{S} = -h(S), \quad S(0; x_0) \geq V(x_0),$$

for any initial condition x_0. •

According to this definition, S is strictly decreasing in time and $\lim_{t\to\infty} S(t; x_0) = 0$ for all x_0, and is increasing in $\|x_0\|$, cf. [Kha02, Lemma 4.4] (with a slight abuse of notation, writing the specification in the form $S(\|x_0\|, t)$ makes it a class $\mathcal{KL}$ function). Note that the exponential rate specification is a particular case of Definition 5.3.1 (by setting $h(s) = -r\,s$). The following result expands the treatment in [Tab07] regarding derivative-based triggers to account for this notion of

performance specification and follows a similar line of reasoning.

Proposition 5.3.2. ***(Derivative-Based Design – Class-$\mathcal{K}$ Derivative).*** *Consider the sample-and-hold nonlinear system* (5.1) *with an ISS Lyapunov function* (5.2)*. Given a class-$\mathcal{K}$ derivative performance specification S and $\sigma \in (0, \sigma^*)$, let $g : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$ be any function such that*

$$\mathcal{L}_f V(x, e) \leq g(x, e) \leq (\sigma - 1)\alpha(\|x\|) + \gamma(\|e\|).$$

Define the derivative-based trigger time as

$$t_{k+1}^{\mathrm{d}} = \min\left\{t \geq t_k \mid g(x(t), e(t)) + h(V(x(t))) \geq 0\right\}. \tag{5.18}$$

Under the assumption that F, κ, γ, α^{-1} are locally Lipschitz, there exists a MIET $\tau_\sigma^{\mathrm{d}} > 0$ such that if $V(x(t_k)) \leq S(t_k; x_0)$, then $t_{k+1}^{\mathrm{d}} - t_k \geq \tau_\sigma^{\mathrm{d}}$. As a consequence, if the sequence $\{t_k\}_{k=0}^{\infty}$ is defined iteratively via the derivative-based trigger, then $V(x(t)) \leq S(t; x_0)$ for all time, and the origin is globally asymptotically stable.

Proof. The trigger design directly enforces $g(x(t), e(t)) < -h(V(x(t))$ for $t \in [t_k, t_{k+1}^{\mathrm{d}})$. Therefore,

$$\frac{d}{dt}V(x(t)) = \mathcal{L}_f V(x(t), e(t)) \leq -h(V(x(t))).$$

Consequently, if $V(x(t_k)) \leq S(t; x_0)$, one can guarantee $V(x(t)) \leq S(t; x_0)$ for all $t \in [t_k, t_{k+1}^{\mathrm{d}}]$ via the Comparison Lemma [Kha02, Lemma 3.4]. Next, we prove the existence of a MIET. Because the sublevel set $\{x \in \mathbb{R}^n \mid V(x) \leq S(0; x_0)\}$) is forward invariant and compact, $\|e\| = \|x - x_k\|$ must be bounded by some constant $E > 0$ and hence the error remains in the compact set $\{e \mid \|e\| \leq E\}$.

On these compact sets, let L_γ and $L_{\alpha^{-1}}$ denote the Lipschitz constants for the functions γ and α^{-1}, respectively. Then,

$$\begin{aligned}
t_{k+1}^{\mathrm{d}} &= \min\left\{t \geq t_k \mid g(x,e) + h(V(x)) \geq 0\right\} \\
&\geq \min\left\{t \geq t_k \mid g(x,e) + (1-\sigma^*)\alpha(\|x\|) \geq 0\right\} \\
&\geq \min\left\{t \geq t_k \mid (\sigma-\sigma^*)\alpha(\|x\|) + \gamma(\|e\|) = 0\right\} \\
&\geq \min\left\{t \geq t_k \mid \frac{L_\gamma}{\sigma^*-\sigma}\|e\| = \alpha(\|x\|)\right\} \\
&\geq \min\left\{t \geq t_k \mid \frac{L_{\alpha^{-1}}L_\gamma}{\sigma^*-\sigma}\|e\| = \|x\|\right\} \\
&= \min\left\{t \geq t_k \mid \frac{\|e\|}{\|x\|} = D^{-1}\right\},
\end{aligned}$$

where $D = \frac{L_{\alpha^{-1}}L_\gamma}{\sigma^*-\sigma}$. Using Lemma 5.6.1, the time at which the condition in the last equation is met is lower bounded by $t_k + \frac{1}{L_f D + L_f}$, where L_f is the Lipschitz constant for f with respect to (x, e) (which exists because F and κ are locally Lipschitz). This establishes the existence of a positive MIET bound, ruling out the possibility of Zeno behavior in the sequence $\{t_k\}_{k=0}^{\infty}$. Finally, asymptotic stability follows from the fact that S is strictly decreasing and $\lim_{t\to\infty} S(t; x_0) = 0$, concluding the proof. □

Next, we build on the ideas presented in Sections 5.1 and 5.2 to introduce the performance-barrier-based trigger design (5.19) for the nonlinear case. The proposed design is based on enforcing the condition (5.17) to ensure the performance specification is met. In doing so, we take advantage of the *performance residual* $S(t; x_0) - V(x(t))$ to avoid overconstraining the evolution of the Lyapunov certificate V when $V(x(t)) < S(t; x_0)$.

Proposition 5.3.3. ***(Performance-Barrier-Based Design – Class-$\mathcal{K}$ Derivative).*** *Consider the*

sample-and-hold nonlinear system (5.1) *with an ISS Lyapunov function* (5.2)*. Given a class-*$\mathcal{K}$ *derivative performance specification* S *and* $\sigma \in (0, \sigma^*)$*, let* g *be as in Proposition 5.3.2 and let* β *be any* $\mathcal{K}_\infty$ *function on* $[0, \infty)$*. Define the performance-barrier-based trigger time as*

$$t_{k+1}^{\mathrm{p}} = \min\left\{t \geq t_k \mid g(x(t), e(t)) + h(V(x(t))) \geq \beta\big(S(t; x_0) - V(x(t))\big)\right\}. \tag{5.19}$$

Under the assumption that F*,* κ*,* γ*,* α^{-1} *are locally Lipschitz, there exists a MIET* $\tau_\sigma^{\mathrm{p}} > 0$ *such that if* $V(x(t_k)) \leq S(t; x_0)$*, then* $t_{k+1}^{\mathrm{p}} - t_k \geq \tau_\sigma^{\mathrm{p}}$*. As a consequence, if the sequence* $\{t_k\}_{k=0}^{\infty}$ *is defined iteratively via the performance-barrier-based trigger, then* $V(x(t)) \leq S(t_k; x_0)$ *for all time, and the origin is globally asymptotically stable.*

Proof. The trigger design directly enforces

$$g(x(t), e(t)) + h(V(x(t))) < \beta\big(S(t; x_0) - V(x(t))\big), \tag{5.20}$$

for $t \in [t_k, t_{k+1}^{\mathrm{p}})$. Thus, when $S(t; x_0) = V(x(t))$, we find $g(x(t), e(t)) + h(S(t; x_0)) < 0$, and hence $\mathcal{L}_f V(x(t), e(t)) < -h(S(t; x_0))$ from the properties of g. Since S is a derivative performance specification, it follows that

$$\frac{d}{dt}V(x(t)) < \frac{d}{dt}S(t; x_0),$$

implying (5.17). Consequently, $V(x(t)) \leq S(t; x_0)$ for all $t \in [t_k, t_{k+1}^{\mathrm{p}})$. We can also use this fact to

deduce

$$t_{k+1}^{\mathrm{p}} \geq \min\left\{t \geq t_k \mid g(x(t), e(t)) + h(V(x(t))) = 0\right\},$$

implying that the performance-barrier-based trigger time must occur after the derivative-based trigger one (5.18). Thus, τ_σ^{d} from Proposition 5.3.2 is a valid MIET for the performance-barrier-based trigger as well, ruling out the possibility of Zeno behavior in $\{t_k\}_{k=0}^{\infty}$. Finally, asymptotic stability follows from the properties of S. □

Note that the function β in Proposition 5.3.3 restricts the speed of evolution of the Lyapunov certificate V when $V(x(t)) < S(t; x_0)$ as a function of the performance residual.

Remark 5.3.4. ***(Comparison with Derivative-Based Approach: Longer Inter-Event Times).*** As pointed out by Propositions 5.3.2 and 5.3.3, both the derivative- and performance-barrier-based approaches meet the performance specification defined by S. However, since the performance residual on the right-hand side of (5.19) always remains greater than zero by design, the performance-barrier-based approach, for a given system state, has a longer inter-event time than the derivative-based one, and is therefore less conservative. In general, it is challenging to provide an explicit bound between the respective MIETs due to the generality of the system dynamics and the performance requirement. We show later in Section 5.3.2 that in the case of exponential performance specification this difference in MIETs can be quantified analytically. •

Remark 5.3.5. ***(Comparison with Function-Based Approach: Robustness to Input Disturbances).*** A purely function-based design would correspond to (5.19) with the left-hand side substituted by zero. Note that the error term does not show up explicitly in such design, in contrast to

the performance-barrier-based approach. Much like how one can use the ISS notion to deal with disturbances, the performance-barrier-based design allows for the analysis and mitigation of input disturbances. This is the intention of the presence of the parameter σ in the definition of g, that reserves a part of the negativity of the Lyapunov function decay. •

Remark 5.3.6. ***(Connection with Dynamic Trigger Design).*** We note that dynamic triggering can be interpreted as a particular case of the performance-barrier-based trigger design, where the performance function is specified in an online fashion. We elaborate on this point here. Formally, and with the same notation employed in Proposition 5.3.3, the dynamic trigger [Gir15] would take the form

$$t_{k+1}^{\text{dyn}} = \min\left\{t \geq t_k \mid \theta g(x(t), e(t)) \geq \eta(t)\right\}, \tag{5.21a}$$

for $\theta > 0$, where the variable η follows the dynamics

$$\dot{\eta} = -\iota(\eta) - g(x, e) \tag{5.21b}$$

with a locally Lipschitz class-$\mathcal{K}_\infty$ function ι. The basic idea is to store the decrease of V in the variable η through (5.21b) and use it to increase the inter-event times in (5.21a). The term $\iota(\eta)$ represents a decay in the stored amount, ensuring that the system as a whole loses total "energy" over time.

Interestingly, the dynamic design (5.21) can be interpreted from the perspective of performance-barrier-based ETC. Selecting the performance specification function $S(t; x_0) = \eta(t) +$

$V(x(t;x_0))$, one can see that the design (5.21) ensures

$$\frac{d}{dt}V(x(t;x_0)) - \frac{d}{dt}S(t;x_0) < \beta(S(t;x_0) - V(x(t;x_0))$$

with $\beta(\eta) = \iota(\eta) + \eta/\theta$ (note the parallelism with the performance-barrier-based design (5.19)), implying (5.17) is satisfied. Note that this performance specification S is not known a priori and is instead determined in an online fashion, tailored to the concrete initial condition of the system trajectory. In particular, this means that the explicit performance guarantee of the design is difficult to obtain unless additional assumptions are made on the dynamics. A final observation is that errors in the evaluation of the decrease of V might jeopardize the convergence properties of dynamic triggering, whereas the evaluation of the performance residual in a feedback fashion characteristic of the performance-barrier-based ETC approach makes it naturally robust to errors. •

5.3.2 Exponential Performance Specification

Here we discuss the exponential performance specification. This is a subfamily of the class-$\mathcal{K}$ derivative performance specifications in Section 5.3.1 for which an explicit analysis of the performance residual leads us to an improved MIET with respect to the derivative-based approach.

In this case, in lieu of the conditions (5.2) for the ISS Lyapunov function $V : \mathbb{R}^n \to \mathbb{R}$, assume the following stronger set of conditions hold: there exist positive constants c_1, c_2, c_3, and

c_4 such that

$$c_1\|x\|^2 \le V(x) \le c_2\|x\|^2, \tag{5.22a}$$

$$\frac{dV}{dx} f(x,0) \le -c_3\|x\|^2, \tag{5.22b}$$

$$\left\|\frac{dV}{dx}\right\| \le c_4\|x\|, \tag{5.22c}$$

for all $x \in \mathbb{R}^n$. Under the additional assumption that F and κ are globally Lipschitz, and using Young's inequality [HLP52], the following inequality holds for all (x, e),

$$\begin{aligned}\mathcal{L}_f V(x,e) = \frac{dV}{dx} f(x,e) &\le -c_3\|x\|^2 + c_4 L_f \|x\|\|e\| \\ &\le -c_\alpha\|x\|^2 + c_\gamma\|e\|^2,\end{aligned} \tag{5.23}$$

for some positive constants L_f, c_α, and c_γ. Notice that the functions $\underline{\alpha}$, $\overline{\alpha}$, α, and γ for this ISS Lyapunov function are defined as quadratic functions with constants c_1, c_2, c_α, c_γ, respectively. Note that in the absence of error, the value of V converges exponentially. Hence, we consider the exponential performance specification $S(t; x_0) = V(x_0)\exp(-rt)$ with $r < c_\alpha/c_2$, which is of class-$\mathcal{K}$ since it is the unique solution to $\dot{S} = -rS$, cf. Definition 5.3.1.

The next result provides an expression for the MIET for the performance-barrier-based trigger design (5.19) and shows it is strictly larger than the MIET τ_σ^{d} of the derivative-based trigger design.

Proposition 5.3.7. ***(Performance-Barrier-Based Design – Exponential Performance).*** *Consider the sample-and-hold nonlinear system* (5.1) *with a Lyapunov function* (5.22)*. Given an exponential performance specification S and $\sigma \in (0, 1 - \frac{rc_2}{c_\alpha})$, let g be as in Proposition 5.3.2, and $\beta(z) = c_\beta z$*

with a positive c_β. Define

$$\tau_\sigma^{\exp} := \min\left\{ \tau \geq 0 \mid (\xi(\tau) + r)\exp\left(\int_0^\tau \xi(s)ds\right) = c_\beta\left(\exp(-r\tau) - \exp\left(\int_0^\tau \xi(s)ds\right)\right)\right\} \tag{5.24}$$

where

$$\begin{aligned}
\tau_\sigma^{\mathrm{d}} = \xi^{-1}(-r) &:= \frac{\sqrt{((1-\sigma)c_\alpha - r)/c_\gamma}}{L_f + L_f\sqrt{((1-\sigma)c_\alpha - r)/c_\gamma}}, \\
\tau^* = \xi^{-1}(0) &:= \frac{\sqrt{(1-\sigma)c_\alpha/c_\gamma}}{L_f + L_f\sqrt{(1-\sigma)c_\alpha/c_\gamma}}, \\
\xi(\tau) &= \begin{cases} ((\sigma-1)c_\alpha + c_\gamma\phi(\tau)^2)/c_2 & 0 \leq \tau < \tau^* \\ ((\sigma-1)c_\alpha + c_\gamma\phi(\tau)^2)/c_1 & \tau^* \leq \tau \end{cases}, \\
\phi(\tau) &= \frac{L_f\tau}{1 - L_f\tau}.
\end{aligned}$$

Under the assumption that F and κ are globally Lipschitz, $\tau_\sigma^{\exp}$ is a MIET such that if $V(x(t_k)) \leq V(x_0)\exp(-rt_k)$, then $t_{k+1}^{\mathrm{p}} - t_k \geq \tau_\sigma^{\exp} > \tau_\sigma^{\mathrm{d}}$. As a consequence, if the trigger sequence $\{t_k\}_{k=0}^\infty$ is defined iteratively with the exponential performance-barrier-based trigger (5.19)*, then $V(x(t)) \leq V(x_0)\exp(-rt)$ for all time, and the origin is globally exponentially stable.*

Proof. The statements on performance and stability follow with the same arguments used in the proof of Proposition 5.3.3. Here, we only establish the MIET expression given for $\tau_\sigma^{\exp}$. First, we

use (5.22a) and Lemma 5.6.1 to find

$$\begin{aligned} g(x(t), e(t)) &\leq ((\sigma - 1)c_\alpha + c_\gamma \phi(t - t_k)^2)\|x\|^2 \\ &\leq \xi(t - t_k)V(x), \end{aligned}$$

where we have used that $\phi(\tau^*)^2 = (1 - \sigma)c_\alpha / c_\gamma$. This gives the bound

$$t_{k+1}^{\mathrm{p}} \geq \min \left\{ t \geq t_k \mid (\xi(t - t_k) + r)V(x(t)) \geq c_\beta (V(x_0)\exp(-rt) - V(x(t))) \right\}.$$

In addition, we can bound the Lyapunov function along the trajectory using the differential form of Gronwall's inequality [Kha02, Lemma A.1] as

$$V(x(t)) \leq V(x_k) \exp\left(\int_{t_k}^{t} \xi(s - t_k) ds \right). \tag{5.25}$$

This helps us isolate the state component, which in turn allows us to bound the trigger time with only the time variable as follows

$$\begin{aligned} t_{k+1}^{\mathrm{p}} \geq \min \Bigg\{ t \geq t_k \mid (\xi(t - t_k) + r) \exp\left(\int_{t_k}^{t} \xi(s) ds \right) \\ \geq c_\beta \left(\exp(-r(t - t_k)) - \exp\left(\int_{t_k}^{t} \xi(s - t_k) ds \right) \right) \Bigg\}. \end{aligned}$$

With the change of variables $\tau = t - t_k$, and using continuity, the condition defining the set is as in (5.24). Next, because ξ is strictly increasing and $\xi(\tau_\sigma^{\mathrm{d}}) = -r$, the left-hand side of the condition is nonpositive for $\tau \leq \tau_\sigma^{\mathrm{d}}$. At the same time, the right-hand side of the condition must always be

positive. Hence, the condition must be met at $\tau_\sigma^{\exp} > \tau_\sigma^{\mathrm{d}}$, concluding the proof. □

Note that the expression (5.24) in Proposition 5.3.7 for the MIET of the performance-barrier-based design with exponential specification does not depend on the state, and can therefore be calculated a priori, before the actual implementation of the controller. We take advantage of the ability to quantify the benefits of the performance-barrier-based approach for exponential specifications when discussing its application to network systems in our forthcoming discussion.

5.4 Performance-Barrier-Based Triggering for Network Systems

In this section we discuss the application of the performance-barrier-based triggering approach to the design of distributed triggers for network systems. Specifically, we consider exponential performance specifications and take advantage of the additional flexibility provided by the performance residual to ensure the existence of a MIET.

Consider a network of N agents whose interconnection is represented by a connected undirected graph $\mathcal{G} = ([N], \mathcal{E})$. By this, we mean that each agent can only communicate with its neighbors, and hence has access to limited information about the system. We make the assumption that the Lyapunov function satisfying (5.22) can be expressed as an aggregate

$$V(x) = \sum_{i=1}^{N} V_i(x_{\mathcal{N}_i}),$$

with each function V_i depending on the local information available to agent i. We assume each V_i to be continuously differentiable with Lipschitz gradient. Our goal is to design distributed triggers

that can be evaluated by individual agents with the information available to them.

5.4.1 Challenges for ETC in Network Systems

Here we describe the challenges in transcribing the derivative-based trigger approach to network systems. The direct transcription of (5.18) to the network setting would result in a centralized trigger that requires global information to be evaluated. Making use of the aggregate decomposition of V, one can instead define

$$t_{k+1} = \min \left\{ t \geq t_k \mid \exists i \in [N] \ni (\sigma - 1)c_\alpha \|x_i(t)\|^2 + c_\gamma \|e_i(t)\|^2 + rV_i(x_{\mathcal{N}_i}(t)) \geq 0 \right\}. \tag{5.26}$$

Note the slight abuse of notation here, where x_i and e_i now refer to the states associated with agent i, rather than the i-th component of vectors x and e, resp. This trigger corresponds to partitioning (5.18) across the network into multiple triggers, one per agent, that can be individually evaluated with local information. Note that the design means that when an agent triggers, a controller update request is sent network-wide. This relies on the observation that such messages, which do not require any state information, can be easily propagated through the network. The design is more conservative than the centralized one and, as a consequence, results in shorter inter-event times for an arbitrary network state. In fact, this type of distributed trigger schemes can suffer from Zeno behavior, see e.g., [HJT12, DH12, TC14, BH14]. A common practice to address this is to explicitly incorporate a MIET at the design stage, a process known as time regularization, see e.g., [BDH16, MT11, TC14]. For instance, with a slight modification to suit our context, [MT11]

proposes the following

$$t_{k+1} = \min\left\{t \geq t_k + \tau_\sigma^{\mathrm{d}} \mid \exists i \in [N] \ \ni (\sigma - 1)c_\alpha \|x_i(t)\|^2 + c_\gamma \|e_i(t)\|^2 + rV_i(x_{\mathcal{N}_i}(t)) \geq b_i(t_k)\right\},$$

where $b \in \mathbb{R}^n$ is a budget variable satisfying $\mathbb{1}^\top b = 0$, which we discuss below. Time regularization discards the possibility of Zeno behavior by forcing the inter-event time to be above the MIET known from the centralized design. The design builds on the fact that, from the analysis in Section 5.3, we know controller updates are not necessary for τ_σ^{d} seconds after the last update in order to meet the performance specification. Consequently, agents can ignore the trigger conditions for this amount of time and only start enforcing them thereafter.

However, note that time regularization does not change the fact that the error $\|e_i\|$ might have already surpassed the level at which the trigger would occur as soon as the trigger condition starts getting monitored, see e.g., [BH14]. The variable b seeks to address this by re-balancing the budget that each agent has in its trigger condition, allowing for the possibility of allocating at the triggering times some budget from a node where the condition has not been violated to another node where it has (in order to have the latter not trigger immediately next time once τ_σ^{d} seconds have elapsed). Among the potential disadvantages of the design (5.27) from a network perspective, we point out the following:

(i) the computation of the MIET τ_σ^{d} can be challenging and requires the execution of a dedicated distributed algorithm prior to the controller implementation. Moreover, the value obtained may turn out to be too conservative, making the trigger occur more frequently than necessary;

(ii) the proposed scheme requires a central entity, albeit only at each triggering time, to calculate

and assign budgets to all the agents;

(iii) without further assumptions on the nonlinear system, the evolution of the trigger condition cannot be predicted, and consequently there is no guarantee that the selected budgets b will successfully extend the inter-event time.

Our proposed method addresses these problems by designing a trigger that intrinsically exhibits a MIET and relying on distributed computation and communication among the agents to calculate their budgets.

5.4.2 Intrisically Zeno-Free Distributed ETC Design

We use two different elements to propose a distributed trigger scheme: dynamic average consensus algorithm and the performance-barrier-based trigger design. We approach the Zeno problem by attacking directly its root cause in distributed settings: partial information of the system states is insufficient to inform agents of system's overall performance. For this reason, our distributed trigger design makes use of dynamic average consensus algorithm to estimate, with some tracking errors, the global terms in the centralized version of the trigger. Doing so transforms the problem into ascertaining how well the trigger design can tolerate errors. This is where we leverage the additional flexibility provided by the performance-barrier-based approach over the derivative-based one regarding handling of the tracking errors. Particularly, as we will show later in the analysis of our design, the performance residual term offered by performance-barrier-based ETC plays a key role in ruling out Zeno behavior.

We begin by defining some notation functions for compactness of presentation. Let

$$\mathcal{W}^x(x) = (\sigma - 1)c_\alpha \|x\|^2 + (r + c_\beta)V(x),$$
$$\mathcal{W}^{xe}(x, e) = \mathcal{W}^x(x) + c_\gamma \|e\|^2.$$

These functions can be decomposed as sums of the following functions, respectively,

$$\mathcal{W}_i^x(x_{\mathcal{N}_i}) = (\sigma - 1)c_\alpha \|x_i\|^2 + (r + c_\beta)V_i(x_{\mathcal{N}_i}),$$
$$\mathcal{W}_i^{xe}(x_{\mathcal{N}_i}, e_i) = \mathcal{W}_i^x(x_{\mathcal{N}_i}) + c_\gamma \|e_i\|^2.$$

For convenience, we let W^x and W^{xe} be vector-valued functions with components $W_i^x = \mathcal{W}_i^x$ and $W_i^{xe} = \mathcal{W}_i^{xe}$, respectively. We omit the dependency on $x_{\mathcal{N}_i}$ and e_i when it is clear from the context. Notice that $\mathbb{1}^\top W^x = \mathcal{W}^x$ and $\mathbb{1}^\top W^{xe} = \mathcal{W}^{xe}$. The centralized performance-barrier-based trigger design (5.19) can be rewritten compactly as

$$t_{k+1}^{\exp} = \min\left\{t \geq t_k \mid \mathcal{W}^{xe}(t) = c_\beta V(x_0)\exp(-rt)\right\}. \tag{5.27}$$

This trigger has a MIET, cf. Proposition 5.3.7, but the direct computation of $\mathcal{W}^{xe}$ requires global information. However, given the aggregate decomposition $\mathbb{1}^\top W^{xe} = \mathcal{W}^{xe}$ and the fact that agent i knows $\mathcal{W}_i^{xe}$, a dynamic average consensus algorithm enables the agents to estimate the average $\mathcal{W}^{xe}/N$. This leads to the following trigger design,

$$t_{k+1} = \min\left\{t \geq t_k \mid \exists i \in [N] \ \ni a_i(t) = c_\beta V(x_0)\exp(-rt)/N\right\}, \tag{5.28a}$$

$$\dot{a} = \dot{W}^{xe} - \rho_a \mathbf{L} a, \tag{5.28b}$$

where $\rho_a > 0$ and $\mathbf{L} \in \mathbb{R}^{N \times N}$ is the graph's Laplacian. With this formulation, we denote the tracking error by $\epsilon_a := a - \mathbb{1}\mathbb{1}^\top W^{xe}/N$. In order for the dynamic average consensus to track the right variable, it is crucial to initialize a so that $\mathbb{1}^\top \epsilon_a = 0$. As such, we assume that $a(0)$ is so that $\epsilon_a(0) = 0$ at the initial time $t = 0$. Since the tracking error's mean $\mathbb{1}^\top \epsilon_a$ is conserved along the dynamics (5.28b), this ensures $\mathbb{1}^\top \epsilon_a = 0$ until the next triggering time. However, the value of $\mathcal{W}^{xe}$ jumps to $\mathcal{W}^x$ at each trigger time t_k due to e being reset to zero, and therefore the average estimate a must be reinitialized at each trigger time t_k to keep the tracking error's mean zero. To do this, we use another dynamic average consensus to keep track of $\mathcal{W}^x$ as

$$\dot{z} = \dot{W}^x - \rho_z \mathbf{L} z \tag{5.28c}$$

where $\rho_z > 0$, with the initial condition $z(0) = \mathbb{1}\mathcal{W}^x(x_0)/N$. Similarly, we denote the tracking error by $\epsilon_z := z - \mathbb{1}\mathbb{1}^\top W^x/N$. Note that the variable z does not depend on e, so it does not need to reinitialize at each t_k. With the new tracking variable, we reinitialize a to z at each trigger time with a jump map,

$$a^+ = z, \; t \in \{t_k\}_{k=0}^{\infty}. \tag{5.28d}$$

Remark 5.4.1. ***(Distributed Implementation).*** The design (5.28) does not require a central entity to estimate the evolution of the trigger condition, relying instead on dynamic average consensus. To implement (5.28), the i-th agent, with local exchange information on a_i and z_i, can evaluate the dynamic average consensus dynamics (5.28b) and (5.28c) if the time derivative of the reference signals $\dot{W}_i^{xe}$ and $\dot{W}_i^x$ are available to it. Each agent i has the information of the states x_i and e_i

and the dynamics $\dot{x}_i$ and $\dot{e}_i$. However, due to dependency on $x_{\mathcal{N}_i}$, the calculation of $\dot{W}_i^{xe}$ and $\dot{W}_i^{x}$ requires knowledge of $x_{\mathcal{N}_i}$ and $\dot{x}_{\mathcal{N}_i}$. The computation of the latter requires two-hop communication in the graph (alternatively, only one-hop communication is required if the decomposition of the Lyapunov function takes the form $V(x) = \sum_{i=1}^{N} V_i(x_i)$). •

Remark 5.4.2. ***(Extensions to Discrete-Time Consensus and Directed Graphs).*** Instead of the continuous-time algorithms in (5.28b) and (5.28c), the design (5.28) could employ discrete-time implementations of the dynamic average consensus algorithm, see e.g., [KSC+19]. Since the effective timescales of (5.28b) and (5.28c) scale linearly with ρ_a and ρ_z, respectively, cf. Lemma 2.2.1, the stepsizes of such discrete-time implementations would scale linearly with $1/\rho_a$ and $1/\rho_z$, respectively. A technical analysis analogous to the one presented in Section 5.4.3 below could be developed, albeit we do not pursue it here for simplicity of exposition. A similar observation can be made about the interconnection structure of the network, which could easily be extended from undirected to weight-balanced, strongly connected directed graphs, cf. [BCM09]. •

5.4.3 Convergence Analysis

In this section we show that the proposed distributed trigger design (5.28), with suitable choices of the parameters c_β, ρ_a, and ρ_z, makes the origin asymptotically stable. Our analysis includes establishing performance satisfaction and a MIET. Regarding the former, from the definition of the trigger, we have that

$$\mathcal{W}^{xe}(t)/N + \epsilon_{a,i}(t) = a_i(t) < c_\beta V(x_0)\exp(-rt)/N \tag{5.29}$$

along the trajectory for all $i \in [N]$. Using the fact that $\mathbb{1}^\top \epsilon_a = 0$ at all time and summing (5.29), we deduce that $\mathcal{W}^{xe}(t) < c_\beta V(x_0)\exp(-rt)$, i.e., the same condition enforced by the centralized trigger (5.27). This shows the satisfaction of performance. Establishing MIET is more complicated. The inequality (5.29) suggests that $\epsilon_{a,i}$ being nonzero can make the distributed trigger (5.28) occur prematurely in comparison to the centralized trigger (5.27). However, our analysis below shows that, by tuning different parameters appropriately, we can ensure that at least for the time interval $[t_k, t_k + \tau_\sigma^{\mathrm{d}})$, the presence of $\epsilon_{a,i}$ does not have this effect, and (5.28) is not triggered. Before establishing this fact, we show next that the reference signals W^{xe} and W^x have an exponentially bounded time derivative. Its proof is given in the appendix.

Lemma 5.4.3. ***(Exponential Bounds for Reference Signals).*** *Consider the distributed trigger design* (5.28) *for the sample-and-hold nonlinear system* (5.1) *with Lipschitz F and κ. Assume that each V_i is continuously differentiable with Lipschitz gradient. Given a desired rate of convergence $r < c_\alpha/c_2$ and $\sigma \in (0, 1 - \frac{rc_2}{c_\alpha})$, there exists $\Omega^{xe} > 0$ such that, for all $k \in \{0\} \cup \mathbb{N}$,*

$$\|\dot{W}^{xe}(t)\| \leq \Omega^{xe} V(x_k)\exp(-r\Delta t_k)$$

for $t \in [t_k, t_k + \tau_\sigma^{\mathrm{d}})$. Furthermore if $c_\beta > (1-\sigma)(c_\alpha/c_1) - r$, there exists $\Omega^x > 0$ such that

$$\|\dot{W}^{x}(t)\| \leq \Omega^{x} V(x_0)\exp(-rt)$$

for all time along the trajectory. □

Lemma 5.4.3 ensures that the requirements to apply Lemma 2.2.1 hold, allowing us to bound ϵ_a and ϵ_z. We are now ready to state the main result of this section.

Theorem 5.4.4. ***(Distributed ETC with Exponential Performance).*** *Consider the sample-and-hold nonlinear system* (5.1) *with a Lyapunov function* (5.22)*. Given a desired rate of convergence* $r < c_\alpha/c_2$ *and* $\sigma \in (0, 1 - \frac{rc_2}{c_\alpha})$*, let* t_{k+1} *be determined iteratively according to the performance-barrier-based distributed trigger* (5.28) *with* $c_\beta > (1-\sigma)(c_\alpha/c_1) - r$*. Under the assumption that* F *and* κ *are Lipschitz and that each* V_i *is continuously differentiable with Lipschitz gradient, let the constant* τ_σ^{d} *be defined as in Proposition 5.3.7. Then, there exist* ρ_a *and* ρ_z *large enough such that*

$$t_{k+1} - t_k \geq \tau_\sigma^{\mathrm{d}},$$

for all $k \in \{0\} \cup \mathbb{N}$*. Consequently, the performance requirement* $V(x(t)) \leq V(x_0)\exp(-rt)$ *is enforced for all time and the origin is rendered globally exponentially stable.*

Proof. Our proof strategy is to show that, for each $k \in \{0\}\cup\mathbb{N}$, $\max_{i\in[N]} a_i - c_\beta V(x_0)\exp(-rt)/N < 0$ during the time period $[t_k, t_k + \tau_\sigma^{\mathrm{d}})$, which implies that no trigger occurs in said period. Note the bound

$$\max_{i\in[N]} a_i = \max_{i\in[N]} \epsilon_{a,i} + \mathcal{W}^{xe}/N \leq \|\epsilon_a\| + \mathcal{W}^{xe}/N.$$

Therefore, it is enough to prove instead that

$$\|\epsilon_a\| + \frac{1}{N}\left(\mathcal{W}^{xe} - c_\beta V(x_0)\exp(-rt)\right) < 0. \tag{5.30}$$

We start bounding the second summand. Using the bounds $\|e\| \leq \phi(t - t_k)\|x\|$ from Lemma 5.6.1

and $\|x\|^2 \geq V(x)/c_2$ from (5.22a),

$$\begin{aligned}\mathcal{W}^{xe} &\leq \left((\sigma-1)c_\alpha + c_\gamma \phi(\Delta t_k)^2\right)\|x\|^2 + (r+c_\beta)V(x) \\ &\leq \left(\frac{(\sigma-1)c_\alpha + c_\gamma\phi(\Delta t_k)^2}{c_2} + r + c_\beta\right)V(x) \\ &= (\xi(\Delta t_k) + r + c_\beta)V(x)\end{aligned}$$

for $t \in [t_k, t_k + \tau_\sigma^{\mathrm{d}}]$. Notice from the second inequality that with $c_\beta > (1-\sigma)(c_\alpha/c_1) - r$, the coefficient of $V(x)$ is positive, so we can use the upper bound of V from (5.25) to get

$$\begin{aligned}\mathcal{W}^{xe} - c_\beta V(x_0)\exp(-rt) &= \mathcal{W}^{xe} - c_\beta V(x_k)\exp(-r\Delta t_k) - c_\beta(V(x_0)\exp(-rt) - V(x_k)\exp(-r\Delta t_k)) \\ &\leq (\xi(\Delta t_k) + r)V(x_k)\exp\left(\int_0^{\Delta t_k}\xi(s)ds\right) \\ &\quad - c_\beta V(x_k)\left(\exp(-r\Delta t_k) - \exp\left(\int_0^{\Delta t_k}\xi(s)ds\right)\right) \\ &\quad - c_\beta(V(x_0)\exp(-rt) - V(x_k)\exp(-r\Delta t_k)).\end{aligned}$$

Consider the first two terms in this expression. Both terms are strictly negative in the time interval $[t_k, t_k + \tau_\sigma^{\mathrm{d}})$, so the maximum value of their sum must be negative. Therefore, there exists $\Omega^* > 0$ (which can be found explicitly by examining its derivative and endpoints on the time interval $\Delta t_k \in [0, \tau_\sigma^{\mathrm{d}}]$) independent of x_k such that

$$\begin{aligned}\mathcal{W}^{xe} - c_\beta V(x_0)\exp(-rt) &\leq -\Omega^* V(x_k) - c_\beta\left(V(x_0)\exp(-rt) - V(x_k)\exp(-r\Delta t_k)\right) \\ &= -\Omega^* V(x_k) - c_\beta\left(V(x_0)\exp(-rt_k) - V(x_k)\right)\exp(-r\Delta t_k) \\ &\leq -\Omega^* V(x_k) - c_\beta\left(V(x_0)\exp(-rt_k) - V(x_k)\right)\exp(-r\tau_\sigma^{\mathrm{d}}). \qquad (5.31)\end{aligned}$$

Note here that both terms in the bound are non-positive.

Regarding the first summand $\|\epsilon_a\|$ in (5.30), we resort to Lemma 2.2.1 to bound it. We write (2.2), with a change of variable to shift time by t_k, for ϵ_a,

$$\|\epsilon_a(t)\| \leq \frac{\Omega^{xe}V(x_k)}{\rho_a\lambda_2 - r}\exp(-r\Delta t_k) + \left(\|\epsilon_a(t_k)\| - \frac{\Omega^{xe}V(x_k)}{\rho\lambda_2 - r}\right)\exp(-\rho\lambda_2\Delta t_k).$$

Over the time interval $\Delta t_k \geq 0$, the bound either achieves the maximum value at $\Delta t_k = 0$ or where its time derivative is zero on the positive interval $\Delta t_k > 0$. In other words, $\|\epsilon_a(t)\| \leq \max\{\|\epsilon_a(t_k)\|, \frac{\Omega^{xe}V(x_k)}{\rho_a\lambda_2 - r}\}$. We consider these two scenarios separately.

First, consider the case where the $\|\epsilon_a(t_k)\| \leq \frac{\Omega^{xe}V(x_k)}{\rho_a\lambda_2 - r}$. By selecting $\rho_a > (1/\lambda_2)(N\Omega^{xe}/\Omega^* + r)$, we can ensure that $\frac{\Omega^{xe}V(x_k)}{\rho_a\lambda_2 - r} < \Omega^* V(x_k)/N$. This shows that the first term in the upper bound (5.31) is enough to dominate $\|\epsilon_a(t)\|$, guaranteeing that (5.30) holds.

Next, consider the case where $\|\epsilon_a(t_k)\| > \frac{\Omega^{xe}V(x_k)}{\rho_a\lambda_2 - r}$. Because $W^{xe}(t_k) = W^x(t_k)$ holds at the update time t_k, we deduce from the jump map (5.28d) that $\epsilon_a(t_k) = \epsilon_z(t_k)$. Thus, the size of $\|\epsilon_a(t_k)\|$ directly depends on how well the dynamic average consensus (5.28c) performs, so we tune ρ_z appropriately so that (5.30) holds. Particularly, we look at the possibility that

$$\|\epsilon_z(t_k)\| \geq c_\beta(V(x_0)\exp(-rt_k) - V(x_k))\exp(-r\tau_\sigma^{\mathrm{d}})/N$$

(otherwise, the second term of the upper bound (5.31) already dominates $\|\epsilon_a(t)\|$). From (2.2), and given the initialization of z with $\epsilon_z(0) = 0$, we have

$$\|\epsilon_z(t_k)\| \leq \frac{\Omega^x V(x_0)}{\rho_z\lambda_2 - r}\big(\exp(-rt_k) - \exp(-\rho_z\lambda_2 t_k)\big).$$

Since $\exp(-\rho_z \lambda_2 t_k) \geq 0$, we obtain the relationship

$$\frac{\Omega^x V(x_0)}{\rho_z \lambda_2 - r} \exp(-r t_k) \geq \|\epsilon_z(t_k)\| \geq c_\beta (V(x_0)\exp(-r t_k) - V(x_k))\exp(-r\tau_\sigma^{\mathrm{d}})/N. \tag{5.32}$$

After some algebraic manipulations, this implies

$$V(x_0)\exp(-r t_k) \leq \frac{c_\beta \exp(-r\tau_\sigma^{\mathrm{d}})}{c_\beta \exp(-r\tau_\sigma^{\mathrm{d}}) - \frac{N\Omega^x}{\rho_z\lambda_2 - r}} V(x_k),$$

if the denominator of the right-hand side is positive. For this to be the case, we have to make sure that our choice of ρ_z satisfies $\rho_z > (1/\lambda_2)(\frac{N\Omega^x}{c_\beta \exp(-r\tau_\sigma^{\mathrm{d}})} + r)$. Substituting the bound above into the upper bound in (5.32), we get

$$\|\epsilon_z(t_k)\| \leq \frac{\Omega^x c_\beta \exp(-r\tau_\sigma^{\mathrm{d}})}{c_\beta \exp(-r\tau_\sigma^{\mathrm{d}})(\rho_z\lambda_2 - r) - N\Omega^x} V(x_k).$$

Now, any selection of ρ_z such that

$$\rho_z > \frac{1}{\lambda_2}\left(\frac{N\Omega^x}{\Omega^*} + \frac{N\Omega^x}{c_\beta \exp(-r\tau_\sigma^{\mathrm{d}})} + r\right),$$

ensures that

$$\frac{\Omega^x c_\beta \exp(-r\tau_\sigma^{\mathrm{d}})}{c_\beta \exp(-r\tau_\sigma^{\mathrm{d}})(\rho_z\lambda_2 - r) - N\Omega^x} < \frac{\Omega^*}{N},$$

and therefore $\|\epsilon_z(t_k)\| < \Omega^* V(x_k)/N$, implying that the first term of the upper bound in (5.31) dominates $\|\epsilon_a(t)\|$. Therefore, (5.30) holds for $t \in [t_k, t_k + \tau_\sigma^{\mathrm{d}})$, and τ_σ^{d} is a MIET for the distributed

trigger design (5.28). With the existence of the MIET, performance satisfaction and global exponential stability follow. □

Theorem 5.4.4 shows that, with the appropriate tuning of the design parameters, (5.28) is an intrinsically Zeno-free event-triggered design for network systems with exponential performance (without the need to prescribe the MIET in the design as in (5.27)). This property relies critically on the performance-barrier-based design approach, particularly on the robustness to errors provided by the performance residual.

Remark 5.4.5. ***(Conservativeness in Design Parameters).*** The required bounds for the design parameters ρ_a and ρ_z developed in the proof of Theorem 5.4.4 are conservative and, in fact, we have observed in practice that values that violate these bounds also result in successful executions. Such bounds must be computed offline, a requirement that is also shared by the time-regularization method regarding the computation of the MIET. However, the key difference, beyond the fact that the method proposed here overcomes the challenges (i)-(iii) described in Section 5.4.1, is that conservativeness in the MIET computation leads to higher actuation resource usage, whereas conservativeness in the bounds of Theorem 5.4.4 imposes requirements on the communication and computational resources of the agents, without affecting the timing of the triggers. •

5.5 Simulations on Vehicle Platooning

To illustrate the effectiveness of the performance-barrier-based trigger design approach, we consider a vehicle platooning problem with $N = 5$ vehicles driving in a line formation along a rectilinear curve. Following [DPH17], we seek to take advantage of the inter-agent communication resources to minimize the usage in actuation resources. The goal is to synchronize the speed v_i of

each vehicle $i \in \{2, \dots, 5\}$ to the leader's desired speed v_{des}, and the vehicle's following distance d_i to a safe distance $d_{\text{des},i} = d_0 + T_v v_i$. Here, d_0 is the standstill following distance and T_v represents the factor for the additional distance to keep with respect to the vehicle's speed. Vehicle 1 is the leader and measures distance with respect to a virtual reference vehicle. We define $\delta_i := d_i - d_{\text{des},i}$ and $\nu := v_i - v_{\text{des}}$ to be the mismatch between the actual and the desired variables. Each vehicle uses a dynamic feedback controller to compute its control input u_i, which directly affects the vehicle's acceleration q_i. The closed-loop dynamics of the leading car, with state $x_i = \begin{bmatrix} \delta_i & \nu_i & q_i & u_i \end{bmatrix}^\top$, can be written as

$$\begin{aligned} \dot{x}_1 &= \begin{bmatrix} 0 & -1 & -T_v & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & -\frac{1}{T_d} & \frac{1}{T_d} \\ \frac{k_p}{T_v} & -\frac{k_d}{T_v} & -k_d & -\frac{1}{T_v} \end{bmatrix} x_1 + \begin{bmatrix} 0 \\ 0 \\ \frac{1}{T_d} \\ 0 \end{bmatrix} e_1 \\ &= \bar{A}_{\text{diag}} x_1 + \bar{E} e_1, \end{aligned}$$

where $e_1 = x_{1,4}(t_k) - x_{1,4}$ encodes the fact that the actual control input u is sampled at time t_k and held constant until t_{k+1}. We use the system parameters $k_p = 0.2$, $k_d = 0.7$, $T_v = 0.6$, and $T_d = 0.1$. Vehicles $\{2, \dots, 5\}$ have dynamics that depend on the cars in front of them, as follows

$$\dot{x}_i = \bar{A}_{\text{diag}} x_i + \bar{A}_{\text{off}} x_{i-1} + \bar{E} e_i,$$

where $e_i = x_{i,4}(t_k) - x_{i,4}$ is the sample-and-hold error and

$$\bar{A}_{\text{off}} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & \frac{k_d}{T_v} & 0 & \frac{1}{T_v} \end{bmatrix}.$$

We next explain how we obtain an ISS Lyapunov function. First, we find $\mathcal{P} > 0$ such that $\bar{A}_{\text{diag}}^\top \mathcal{P} + \mathcal{P}\bar{A}_{\text{diag}} = -\mathbb{I}$ (this corresponds to ignoring the interconnection of each following vehicle with the one in front). Next, we define

$$V(x) = \sum_{i=1}^{N} \pi^{N-i} x_i^\top \mathcal{P} x_i,$$

where π is a weight factor to be chosen. Note that this definition naturally places more weight to the vehicles towards the front of the platoon. The Lie derivative of V is given by

$$L_f V(x,e) = -\pi^{N-1}\|x_1\|^2 + \sum_{i=2}^{N} \pi^{N-i}(-\|x_i\|^2 + 2x_i^\top \mathcal{P}\bar{A}_{\text{off}} x_{i-1}) + \sum_{i=1}^{N} \pi^{N-i} 2x_i^\top \mathcal{P}\bar{E}e_i.$$

Using Young's inequality [HLP52], we can bound the cross terms as $2x_i^\top P\bar{A}_{\text{off}} x_{i-1} \leq 5\|\mathcal{P}\bar{A}_{\text{off}}\|^2\|x_{i-1}\|^2 + (1/5)\|x_i\|^2$. Selecting then $\pi = 31.25\|\mathcal{P}\bar{A}_{\text{off}}\|^2$, we find, after some algebraic manipulations, that

$$L_f V(x,e) \leq -0.145V(x) + \sum_{i=1}^{N} \pi^{N-i} 2x_i^\top \mathcal{P}\bar{E}e_i.$$

This implies a rate of convergence of $r^* = 0.145$ in the absence of sample-and-hold error e. In our simulations, we specify the desired exponential convergence rate $r = 0.08 < 0.75r^*$ for the triggered implementations.

With all the elements in place, we are ready to provide a comparison of different event-triggered control approaches. We implement the centralized performance-barrier-based trigger design, specifically the linear one in (5.12), and compare it to the derivative-based design (5.9). For this, we use

$$g(x,e) = -0.75\pi^{N-1}\|x_1\|^2 + 0.75\sum_{i=2}^{N}\pi^{N-i}(-\|x_i\|^2 + 2x_i^\top \mathcal{P}\bar{A}_{\text{off}}x_{i-1}) + \sum_{i=1}^{N}\pi^{N-i}2x_i^\top \mathcal{P}\bar{E}e_i, \quad (5.33)$$

and $c_\beta = 1$. Each simulation lasts 400 seconds. Figure 5.2 shows the evolution of the Lyapunov functions in logarithmic scale for different trigger designs and Table 5.1 shows the empirical MIET (which might be larger than the actual MIET) and average number of controller updates across 50 different trajectories with random initial conditions. As expected, both designs satisfy the required performance. However, it is evident from Figure 5.2 that the derivative-based design outperforms the requirement, meaning that the number of updates could be significantly reduced. This is precisely what the performance-barrier-based design accomplishes by tuning the timing of the updates to the degree of satisfaction of the prescribed performance, reducing their number by almost 20-fold on average.

We also compare the proposed designs with the dynamic trigger approach, cf. Remark 5.3.6. To do so, we choose a linear decay function $\iota(\eta) = c_\iota\eta$, and consider different values of c_ι. According to (5.21), the degree of decay of the Lyapunov function V grows with the value of c_ι, but it is not possible to determine in advance whether a given value of c_ι will guarantee that the evolution meets

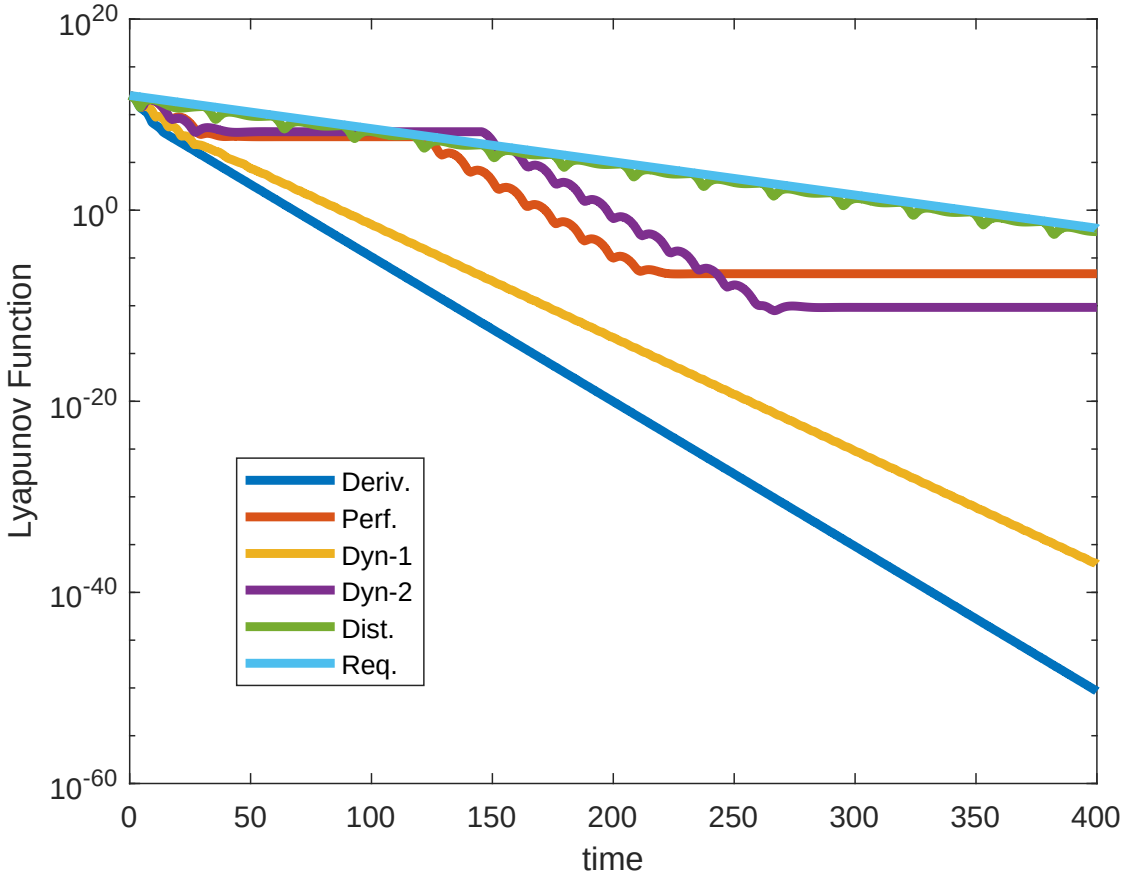


Figure 5.2: Evolution of the Lyapunov function for different trigger designs.

the desired performance specification. We first use $c_t = 1$, and observe, cf. Figure 5.2, that the evolution of V is, similarly to that of the derivative-based design, too conservative. Consequently, we employ $c_t = 0.05$, which leads to a significant decrease in the number of updates, cf. Table 5.1, at the cost of not meeting the performance specification any more, cf. Figure 5.2. One could go through the exercise of fine-tuning the value of c_t to make sure the trajectories meet the desired performance, but this would have to be verified a posteriori in an empirical way, rather than a priori by design, as the performance-barrier-based approach does.

Lastly, we also report the simulation results of the distributed trigger design (5.28) with $\rho_a = 10$ and $\rho_z = 20$. In fact, notice that both the Lyapunov function V and g in (5.33) can be

Table 5.1: Empirical MIET and average number of updates from 50 different random initial conditions

Design	MIET (s)	Avg. no. of updates
Derivative-Based (5.9)	0.009	198.28
Performance-Barrier-Based (5.12)	0.009	9.94
Dynamic (5.21)	0.013	108.48
Dynamic (5.21) – small decay	0.013	7.18
Distributed (5.28)	0.003	96.38

expressed as the sum of functions, one per agent, whose value can be computed by each agent with local information,

$$V_i(x_i) = \pi^{N-i} x_i^\top \mathcal{P} x_i, \qquad W_1^x(x_1) = -0.75\pi^{N-1}\|x_1\|^2,$$

$$W_i^x(x_{\mathcal{N}_i}) = 0.75\pi^{N-i}(-\|x_i\|^2 + 2x_i^\top \mathcal{P}\bar{A}_{\text{off}} x_{i-1}),\ \forall i \geq 2$$

$$W_i^{xe}(x_{\mathcal{N}_i}, e_i) = W_i^x(x_{\mathcal{N}_i}) + \pi^{N-i} 2x_i^\top \mathcal{P}\bar{E} e_i,\ \forall i \geq 1.$$

The distributed implementation meets the prescribed performance, cf. Figure 5.2 and is free of Zeno behavior, as guaranteed by Theorem 5.4.4. This implementation triggers less often than the centralized derivative-based approach and, as expected, more often than the centralized performance-barrier-based design, cf. Table 5.1.

5.6 Chapter Appendix

Proof of Lemma 5.4.3. Note that, since F and κ are Lipschitz, then f is Lipschitz too. Consider the column vector composed of $\{V_i\}_{i=1}^N$ and let $J_V(x)$ be its Jacobian. Then, because each V_i have Lipschitz gradients, there exist constants L_{dV} and L_f on the compact sublevel set $\left\{x \mid V(x) \leq \right.$

$V(x_0)\big\}$ such that

$$\begin{aligned}\|\dot{W}^{xe}\| &= \|\big((\sigma-1)c_\alpha x^\top - c_\gamma e^\top + (r+c_\beta)J_V(x)\big)f(x,e)\| \\ &\leq \big((1-\sigma)c_\alpha\|x\| + c_\gamma\|e\| + (r+c_\beta)L_{dV}\|x\|\big)(L_f(\|x\|+\|e\|)). \end{aligned} \tag{5.34}$$

We next bound the quadratic terms $\|x\|^2$, $\|e\|^2$ and $\|x\|\|e\|$ in terms of $V(x_k)\exp(-r\Delta t_k)$ for the duration of the interval $[t_k, t_k+\tau_\sigma^{\mathrm{d}})$. First, knowing that $V(x) \leq V(x_k)\exp(-r\Delta t_k)$ over the interval, we can immediately bound $\|x\|^2 \leq V(x_k)\exp(-r\Delta t_k)/c_1$. Next, for $\|e\|^2$, recall that τ_σ^{d} is the minimum inter-event time for the derivative-based design, and we can therefore bound

$$\begin{aligned}\|e\|^2 &\leq (1/c_\gamma)\big((1-\sigma)c_\alpha\|x\|^2 - rV(x)\big) \\ &\leq (1/c_\gamma)((1-\sigma)c_\alpha/c_1 - r)V(x) \\ &\leq (1/c_\gamma)((1-\sigma)c_\alpha/c_1 - r)V(x_k)\exp(-r\Delta t_k),\end{aligned}$$

for $t \in [t_k, t_k+\tau_\sigma^{\mathrm{d}})$. Finally, it follows that

$$\|x\|\|e\| \leq \sqrt{\frac{(1-\sigma)c_\alpha/c_1 - r}{c_1 c_\gamma}}V(x_k)\exp(-r\Delta t_k),$$

for $t \in [t_k, t_k+\tau_\sigma^{\mathrm{d}})$. Substituting the bounds back into (5.34) leads to the identification of $\Omega^{xe} > 0$, proving the claim for $\|\dot{W}^{xe}\|$.

For the bound of $\|\dot{W}^{x}\|$, we consider the entire time interval $t \in [t_k, t_{k+1})$. Using the

performance satisfaction, we bound

$$\|x\|^2 \leq V(x)/c_1 \leq V(x_0)\exp(-rt)/c_1.$$

From the trigger condition and $c_\beta > (1-\sigma)(c_\alpha/c_1) - r$,

$$\begin{aligned} c_\gamma\|e\|^2 &\leq c_\beta V(x_0)\exp(-rt) - (r + c_\beta + (\sigma-1)\frac{c_\alpha}{c_1})V(x) \\ &\leq c_\beta V(x_0)\exp(-rt) \end{aligned}$$

The result now follows using the same line of reasoning as in the proof of the bound for $\|\dot{W}^{xe}\|$ to conclude the existence of $\Omega^x > 0$ as stated. □

For the sake of completeness, we state the following result on the sample-and-hold error bound.

Lemma 5.6.1. ***(Sample-and-Hold Error Bound [Tab07, Thm III.1]).*** *Consider the sample-and-hold nonlinear system* (5.1)*. If the functions f is Lipschitz with a constant L_f, then for $t \in [t_k, t_k + 1/L_f)$, the state deviation is bounded as*

$$\|e\| \leq \phi(t - t_k)\|x\|$$

where $\phi(\tau) = \frac{L_f\tau}{1-L_f\tau}$. ■

Acknowledgements

This chapter, in part, has been submitted for publication of the material [OC21b] as it may appear as 'Performance-barrier-based event-triggered control with applications to network systems' by P. Ong and J. Cortés, in IEEE Transactions on Automatic Control, 2021. The dissertation author was the primary investigator and author of this paper. The work in this chapter was partially supported by NSF Award ECCS-1917177.

Chapter 6

Event-Triggered Safety

The efficient utilization of available resources while simultaneously achieving control objectives is a primary motivation in the event-triggered control paradigm. In many modern control applications, one such objective is enforcing the safety of a system. The goal of this paper is to carry out this vision by combining event-triggered and safety-critical control design. We discuss how a direct transcription, in the context of safety, of event-triggered methods for stabilization may result in designs that are not implementable on real hardware due to the lack of a minimum interevent time. We provide an example showing this phenomena and, building on the insight gained, propose an event-triggered control approach via Input-to-State Safe Barrier Functions that achieves safety while ensuring that interevent times are uniformly lower bounded.

6.1 Event-Triggered Stability

In this section we discuss event-triggered control and review the problem of event-triggered stabilization following [Tab07]. This review will motivate our approach for achieving event-

triggered safety presented in Sections 6.3 and 6.4.

Consider the nonlinear control system given by:

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}), \tag{6.1}$$

where $\mathbf{x} \in \mathbb{R}^n$, $\mathbf{u} \in \mathbb{R}^m$, and $\mathbf{f} : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^n$ is locally Lipschitz continuous in both arguments on $\mathbb{R}^n \times \mathbb{R}^m$. We further assume that $\mathbf{f}(\mathbf{0}, \mathbf{u}_0) = \mathbf{0}$ for some $\mathbf{u}_0 \in \mathbb{R}^m$. Under the choice of a Lipschitz continuous state-feedback controller $\mathbf{k} : \mathbb{R}^n \to \mathbb{R}^m$, with $\mathbf{k}(\mathbf{0}) = \mathbf{u}_0$, the closed-loop system dynamics are given by:

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{k}(\mathbf{x})). \tag{6.2}$$

The assumption on local Lipschitz continuity of $\mathbf{f}$ and $\mathbf{k}$ implies that for any initial condition $\mathbf{x}_0 := \mathbf{x}(0) \in \mathbb{R}^n$, there exists a maximum time interval $I(\mathbf{x}_0) = [0, t_{\max})$ such that $\mathbf{x}(t)$ is the unique solution to (6.2) on $I(\mathbf{x}_0)$. In the case $\mathbf{f}(\cdot, \mathbf{k}(\cdot))$ is forward complete, $t_{\max} = \infty$.

In an event-triggered context, the implementation of the feedback control law $\mathbf{k}$ is done by sampling the state at sequential time instances, $t_0, t_1, t_2, \ldots$, and evaluating the controller on the corresponding states $\mathbf{x}(t_0), \mathbf{x}(t_1), \mathbf{x}(t_2), \ldots$ Between measurements the control input is held constant:

$$\mathbf{u}(t) = \mathbf{k}(\mathbf{x}(t_i)) \quad \forall t \in [t_i, t_{i+1}). \tag{6.3}$$

The time instances at which the controller is updated are determined by a state-dependent execution rule or trigger law. We define the measurement error as:

$$\mathbf{e}(t) = \mathbf{x}(t_i) - \mathbf{x}(t) \qquad \forall t \in [t_i, t_{i+1}), \tag{6.4}$$

noting that this in conjunction with (6.3) implies:

$$\dot{\mathbf{x}}(t) = -\dot{\mathbf{e}}(t) = \mathbf{f}(\mathbf{x}(t), \mathbf{k}(\mathbf{x}(t_i))) \qquad \forall t \in [t_i, t_{i+1}), \tag{6.5}$$

The closed-loop dynamics (6.5) can alternatively be written as:

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{k}(\mathbf{x} + \mathbf{e})). \tag{6.6}$$

where $\mathbf{e} \in \mathbb{R}^n$ is the measurement error.

Event-triggered stabilization relies on the robustness to disturbances of the original dynamics, formalized through Input-to-State Stable Lyapunov Functions (ISS-LF) [SW95, Son08].

Notation: Throughout the paper we make use of the following basic definitions. A continuous function $\alpha : [0, a) \to \mathbb{R}_+$, with $a > 0$, is *class* $\mathcal{K}$ ($\alpha \in \mathcal{K}$) if $\alpha(0) = 0$ and α is strictly monotonically increasing. If $a = \infty$ and $\lim_{r\to\infty} \alpha(r) = \infty$, then α is *class* $\mathcal{K}_\infty$ ($\alpha \in \mathcal{K}_\infty$). A continuous function $\alpha : (-b, a) \to \mathbb{R}$, with $a, b > 0$, is *extended class* $\mathcal{K}$ ($\alpha \in \mathcal{K}_e$) if $\alpha(0) = 0$ and α is strictly monotonically increasing. If $a, b = \infty$, $\lim_{r\to\infty} \alpha(r) = \infty$, and $\lim_{r\to-\infty} \alpha(r) = -\infty$, then α is *extended class* $\mathcal{K}_\infty$ ($\alpha \in \mathcal{K}_{\infty,e}$)

Definition 6.1.1 (*ISS Lyapunov Function (ISS-LF)*)**.** *A continuously differentiable function* $V : \mathbb{R}^n \to \mathbb{R}_+$ *is an Input-to-State Stable Lyapunov Function (ISS-LF) for* (6.6)*, with respect to measurement errors* $\mathbf{e}$*, if there exists* $\alpha_1, \alpha_2, \alpha_3 \in \mathcal{K}_\infty$ *and* $\gamma \in \mathcal{K}_\infty$ *such that for all* $\mathbf{x}, \mathbf{e} \in \mathbb{R}^n$*:*

$$\alpha_1(\|\mathbf{x}\|_2) \leq V(\mathbf{x}) \leq \alpha_2(\|\mathbf{x}\|_2), \tag{6.7a}$$

$$\frac{\partial V}{\partial \mathbf{x}}(\mathbf{x})\mathbf{f}(\mathbf{x}, \mathbf{k}(\mathbf{x} + \mathbf{e})) \leq -\alpha_3(\|\mathbf{x}\|_2) + \gamma(\|\mathbf{e}\|_2). \tag{6.7b}$$

As in [Tab07], if we define the trigger law to enforce:

$$\gamma(\|\mathbf{e}(t)\|_2) \leq \sigma\alpha_3(\|\mathbf{x}(t)\|_2) \qquad 0 < \sigma < 1, \tag{6.8}$$

the ISS-LF condition (6.7b) leads to:

$$\frac{\partial V}{\partial \mathbf{x}}(\mathbf{x})\mathbf{f}(\mathbf{x}, \mathbf{k}(\mathbf{x}+\mathbf{e})) \leq (\sigma - 1)\alpha_3(\|\mathbf{x}\|_2), \tag{6.9}$$

implying asymptotic stability of (6.6) to $\mathbf{x}^\star = \mathbf{0}$. The inequality in (6.8) can be enforced by defining the trigger law as:

$$t_{i+1} = \min\left\{t \geq t_i \mid \gamma(\|\mathbf{e}(t)\|_2) = \sigma\alpha_3(\|\mathbf{x}(t)\|_2)\right\}. \tag{6.10}$$

As is typical in event-triggered control formulations, it is critical to show that such a trigger law does not lead to the control being updated at arbitrarily close time instances [PTNA15], or that the *interevent times* $\{t_{i+1} - t_i\}_{i\in\mathbb{N}}$ are lower bounded by a positive constant $\tau \in \mathbb{R}$, $\tau > 0$, referred to as the minimum interevent time (MIET). This differs slightly from preventing the stronger notion of Zeno behavior [BH14], in which the series of interevent times converges (implying the lack of a MIET). The results of [Tab07] ensure that a MIET exists under the trigger (6.9) under the assumption of Lipschitz continuity on compacts of $\mathbf{f}$, $\mathbf{k}$, α, and γ.

6.2 Input-to-State Safety

In this section we provide background information on Barrier Functions (BFs) and Input-to-State Safe Barrier Functions (ISSf-BFs) that will be used to construct an event-triggered control

paradigm that ensures safety.

Consider a set $\mathcal{C} \subseteq \mathbb{R}^n$ defined as the 0-superlevel set of a continuously differentiable function $h : \mathbb{R}^n \to \mathbb{R}$, yielding:

$$\mathcal{C} \triangleq \{\mathbf{x} \in \mathbb{R}^n : h(\mathbf{x}) \geq 0\}, \tag{6.11a}$$

$$\partial\mathcal{C} \triangleq \{\mathbf{x} \in \mathbb{R}^n : h(\mathbf{x}) = 0\}, \tag{6.11b}$$

$$\text{Int}(\mathcal{C}) \triangleq \{\mathbf{x} \in \mathbb{R}^n : h(\mathbf{x}) > 0\}. \tag{6.11c}$$

We assume that $\mathcal{C}$ is nonempty and has no isolated points, that is, $\text{Int}(\mathcal{C}) \neq \varnothing$ and $\overline{\text{Int}(\mathcal{C})} = \mathcal{C}$. We refer to $\mathcal{C}$ as the *safe set*. This construction motivates the following definitions:

Definition 6.2.1 (*Forward Invariant & Safety*)**.** *A set $\mathcal{C}$ is forward invariant if for every $\mathbf{x}_0 \in \mathcal{C}$, the solution $\mathbf{x}(t)$ to* (6.2) *satisfies $\mathbf{x}(t) \in \mathcal{C}$ for all $t \in I(\mathbf{x}_0)$. The system* (6.2) *is safe on $\mathcal{C}$ if the set is forward invariant.*

Verifying that the system (6.2) is safe on a set $\mathcal{C}$ can be done via a Barrier Function:

Definition 6.2.2 (*Barrier Function (BF)*)**.** *A continuously differentiable function $h : \mathbb{R}^n \to \mathbb{R}$ is a Barrier Function (BF) for* (6.2) *on a set $\mathcal{C} \subset \mathbb{R}^n$ defined as in* (6.11a)-(6.11c)*, if there exists $\alpha \in \mathcal{K}_{\infty,e}$ such that for all $\mathbf{x} \in \mathbb{R}^n$:*

$$\frac{\partial h}{\partial \mathbf{x}}(\mathbf{x})\mathbf{f}(\mathbf{x}, \mathbf{k}(\mathbf{x})) \geq -\alpha(h(\mathbf{x})), \tag{6.12}$$

As shown in [XTGA15], the existence of a barrier function for (6.2) on a set $\mathcal{C}$ is sufficient to prove the safety and asymptotic stability of $\mathcal{C}$. To consider the impact of measurement errors on safety, we consider the notion of input-to-state safety [KA18].

Definition 6.2.3 (*Input-to-State Safety (ISSf)*)**.** *Let the signal* $\mathbf{e} : \mathbb{R}_+ \to \mathbb{R}^n$ *be essentially bounded and define* $\|\mathbf{e}\|_\infty = ess\sup_{t\geq 0} \|\mathbf{e}(t)\|_2$*. The closed-loop system* (6.6) *is input-to-state safe (ISSf) on* $\mathcal{C}$ *with respect to measurement errors* $\mathbf{e}$ *if there exists* $\gamma \in \mathcal{K}_\infty$ *and a set* $\mathcal{C}_\mathbf{e} \supset \mathcal{C}$ *defined as:*

$$\mathcal{C}_\mathbf{e} \triangleq \left\{\mathbf{x} \in \mathbb{R}^n : h(\mathbf{x}) + \gamma(\|\mathbf{e}\|_\infty) \geq 0\right\}, \tag{6.13a}$$

$$\partial\mathcal{C}_\mathbf{e} \triangleq \{\mathbf{x} \in \mathbb{R}^n : h(\mathbf{x}) + \gamma(\|\mathbf{e}\|_\infty) = 0\}, \tag{6.13b}$$

$$Int(\mathcal{C}_\mathbf{e}) \triangleq \{\mathbf{x} \in \mathbb{R}^n : h(\mathbf{x}) + \gamma(\|\mathbf{e}\|_\infty) > 0\}, \tag{6.13c}$$

such that (6.6) *is safe on* $\mathcal{C}_\mathbf{e}$.

We refer to $\mathcal{C}$ as an *input-to-state safe* set (ISSf set) if such a set $\mathcal{C}_\mathbf{e}$ exists. This definition implies that though the set $\mathcal{C}$ may not be safe, a larger set $\mathcal{C}_\mathbf{e}$, depending on $\mathbf{e}$, is safe. If $\mathbf{e} \equiv \mathbf{0}$, we recover that the set $\mathcal{C}$ is safe. This motivates the following definition of Input-to-State Safe Barrier Functions:

Definition 6.2.4 (*Input-to-State Safe Barrier Function (ISSf-BF)*)**.** *A continuously differentiable function* $h : \mathbb{R}^n \to \mathbb{R}$ *is an Input-to-State Safe Barrier Function (ISSf-BF) for* (6.6) *on a set* $\mathcal{C} \subset \mathbb{R}^n$ *defined as in* (6.11a)-(6.11c)*, if there exists* $\alpha \in \mathcal{K}_{\infty,e}$ *and* $\iota \in \mathcal{K}_\infty$ *such that for all* $\mathbf{x}, \mathbf{e} \in \mathbb{R}^n$*:*

$$\frac{\partial h}{\partial \mathbf{x}}(\mathbf{x})\mathbf{f}(\mathbf{x}, \mathbf{k}(\mathbf{x}+\mathbf{e})) \geq -\alpha(h(\mathbf{x})) - \iota(\|\mathbf{e}\|_2), \tag{6.14}$$

As shown in [KA18], the existence of an ISSf-BF for (6.6) on $\mathcal{C}$ implies $\mathcal{C}$ is an ISSf set, implying safety and asymptotic stability to the set $\mathcal{C}_\mathbf{e}$.

6.3 Towards Resource-Aware Safety: from Lyapunov to Barriers

To more efficiently utilize actuation resources when implementing safe controllers, we seek to unify the preceding concepts of event-triggered control and input-to-state safety. In this section we discuss challenges that arise in directly transferring ideas from event-triggered stabilization to safety. Given the similarity of the ISS-LF constraint (6.7b) and the ISSf-BF constraint (6.14), it is natural to propose a trigger law that enforces:

$$\iota(\|\mathbf{e}(t)\|_2) \leq \sigma\alpha(h(\mathbf{x}(t))) \qquad 0 < \sigma, \tag{6.15}$$

implying:

$$\frac{\partial h}{\partial \mathbf{x}}(\mathbf{x})\mathbf{f}(\mathbf{x}, \mathbf{k}(\mathbf{x}+\mathbf{e})) \geq -(1+\sigma)\alpha(h(\mathbf{x})). \tag{6.16}$$

This can be interpreted as allowing the system to more quickly approach the boundary of the safe set at the expense of actuation resources. It is important to note that, inside $\mathcal{C}$, it is possible to satisfy (6.15) and thus enforce safety, but the inequality is impossible to satisfy outside $\mathcal{C}$ as $\alpha(h(\mathbf{x})) < 0$ if $\mathbf{x} \notin \mathcal{C}$. This type of behavior does not arise in the context of event-triggered stabilization, where convergence is to a point. One way to solve this issue is to instead define the trigger law:

$$\iota(\|\mathbf{e}(t)\|_2) \leq \sigma|\alpha(h(\mathbf{x}(t)))| \qquad 0 < \sigma < 1, \tag{6.17}$$

which enforces (6.16) if $\mathbf{x} \in C$ and enforces:

$$\frac{\partial h}{\partial \mathbf{x}}(\mathbf{x})\mathbf{f}(\mathbf{x}, \mathbf{k}(\mathbf{x}+\mathbf{e})) \geq -(1-\sigma)\alpha(h(\mathbf{x})), \tag{6.18}$$

if $\mathbf{x} \notin C$. In this formulation, the system is not only allowed to more quickly approach the boundary, but is also not required to converge to the set as quickly when outside of the set. This is a generalization of event-triggered stabilization to a set.

Even with this solution, it is not guaranteed that this trigger law will have a MIET. Although ruling out Zeno behavior is not required to guarantee safety, unlike stabilization, it is important to have a MIET in term of implementation of the controller (cf. [BH14]). The key difference between stability and safety leading to the failure of a MIET to exist for a safe event-triggered controller lies in how the system dynamics must behave close to an equilibrium point compared to how they can behave close to the boundary of the safe set. In stabilization, continuity of the dynamics requires the dynamics to vanish as the equilibrium is approached, leading to the error dynamics in (6.5) vanishing. In safety, the dynamics close to the boundary of the safe set need not vanish as the boundary is approached, such that the error dynamics in (6.5) need not vanish. We provide the following example to illustrate how this difference can lead to a MIET failing to exist for the trigger design (6.17).

6.3.1 Example

Consider the following system:

$$\frac{\mathrm{d}}{\mathrm{d}t}\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} x_2 \\ -x_1 \end{bmatrix} + \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} u. \tag{6.19}$$

for which we wish to ensure the safety of the set $\mathcal{C}$, given by the 0-superlevel set of the continuously differentiable function $h(\mathbf{x}) = 1 - x_1^2 - x_2^2 = 1 - \|\mathbf{x}\|_2^2$. The time derivative of this function along solutions to (6.19) is given by $\dot{h}(\mathbf{x}, u) = -2(x_1^2 + x_2^2)u$, for which the state-feedback controller $k(\mathbf{x}) = \frac{1}{2}(1 - x_1^2 - x_2^2)$ yields $\dot{h}(\mathbf{x}) = -(x_1^2 + x_2^2)h(\mathbf{x}) \geq -h(\mathbf{x})$, which implies h is a valid BF for the set $\mathcal{C}$.

In an event triggered context, the closed-loop dynamics of the system are then given by:

$$\dot{\mathbf{x}}(t) = \begin{bmatrix} k(\mathbf{x}(t_i)) & 1 \\ -1 & k(\mathbf{x}(t_i)) \end{bmatrix} \mathbf{x}(t), \tag{6.20}$$

for each time $t \in [t_i, t_{i+1})$. This leads to the time derivative of h along solutions to (6.20) being given by:

$$\dot{h}(\mathbf{x}(t), \mathbf{e}(t)) = -\|\mathbf{x}(t)\|_2^2 h(\mathbf{x}(t_i)) = -\|\mathbf{x}(t)\|_2^2 h(\mathbf{x}(t) + \mathbf{e}(t))$$

for each time $t \in [t_i, t_{i+1})$, where $\mathbf{e}(t) = \mathbf{x}(t_i) - \mathbf{x}(t)$. To see that the BF h is in fact an ISSf-BF, we

note its time derivative can be bounded as follows (we omit the time dependence):

$$\begin{aligned}
\dot{h}(\mathbf{x},\mathbf{e}) &= -\|\mathbf{x}\|_2^2 h(\mathbf{x}+\mathbf{e}) \\
&= -\|\mathbf{x}\|_2^2(1-\|\mathbf{x}\|_2^2 - 2\mathbf{x}^\top\mathbf{e} - \|\mathbf{e}\|_2^2) \\
&\geq -\|\mathbf{x}\|_2^2(1-\|\mathbf{x}\|_2^2) - 2\|\mathbf{x}\|_2^3\|\mathbf{e}\|_2 \\
&\geq -(1-\|\mathbf{x}\|_2^2) - 2\|\mathbf{x}\|_2^3\|\mathbf{e}\|_2 \\
&\geq -h(\mathbf{x}) - 2r^3\|\mathbf{e}\|_2,
\end{aligned}$$

for $\|\mathbf{x}\|_2 \leq r$. Given that h is an ISSf-BF on some domain containing the unit circle (choose $r > 1$), the trigger law enforcing (6.17) is given by:

$$t_{i+1} = \min\{t \geq t_i \mid 2r^3\|\mathbf{e}(t)\|_2 = \sigma|h(\mathbf{x}(t))|\}, \tag{6.21}$$

with $0 < \sigma < 1$. This will guarantee that C is safe as:

$$\dot{h}(\mathbf{x},\mathbf{e}) \geq \begin{cases} -(1+\sigma)h(\mathbf{x}), & \|\mathbf{x}\|_2 \leq 1, \\ -(1-\sigma)h(\mathbf{x}), & 1 < \|\mathbf{x}\|_2 \leq r. \end{cases}$$

Despite $\mathbf{f}$, $\mathbf{k}$, α, and ι being Lipschitz continuous on compacts as sufficient in the case of stabilization, the following results shows the trigger design lacks a MIET.

Lemma 6.3.1 (MIET does not exist)**.** *The system* (6.20) *with the trigger law defined as in* (6.21) *does not possess a MIET.*

Proof. To show that the interevent times $\{t_{i+1} - t_i\}_{i\in\mathbb{N}}$ are not lower bounded, we will proceed via

contradiction. In particular, let us assume that there exists $\tau > 0$ such that $t_{i+1} - t_i \geq \tau$ for all $i \in \mathbb{N}$. If the state $\mathbf{x}_i = \mathbf{x}(t_i)$ at event time t_i is used as an initial condition, the solution to (6.20) is:

$$\begin{aligned}\mathbf{x}(t) &= \exp\left(\frac{h(\mathbf{x}_i)\Delta t_i}{2}\right)\begin{bmatrix}\cos\Delta t_i & \sin\Delta t_i \\ -\sin\Delta t_i & \cos\Delta t_i\end{bmatrix}\mathbf{x}_i, \\ &= \mathbf{M}_i(\Delta t_i)\mathbf{x}_i,\end{aligned}$$

for $t \in [t_i, t_{i+1})$ with $\Delta t_i = t - t_i$. Denoting $\omega_i = h(\mathbf{x}_i)\Delta t_i$, we see that the norm of the error is lower bounded by a function monotonically increasing in time:

$$\begin{aligned}\|\mathbf{e}(t)\|_2 &= \|(\mathbf{I} - \mathbf{M}_i(\Delta t_i))\mathbf{x}_i\|_2, \\ &= \sqrt{\left(\exp(\omega_i) - 2\exp\left(\frac{\omega_i}{2}\right)\cos(\Delta t_i) + 1\right)}\|\mathbf{x}_i\|_2, \\ &\geq \left|\exp\left(\frac{\omega_i}{2}\right) - 1\right|\|\mathbf{x}_i\|_2.\end{aligned}$$

This lower bound on the error grows unbounded in time. This implies that no matter the state in $\text{Int}(C)$ that an event occurs, another event must occur at some time in the future (or the bound in (6.21) will be violated as h is upper bounded on C). Thus, for all $T > 0$, there exists an event time $t_i > T$.

Next, we show that $\lim_{t\to\infty} h(\mathbf{x}(t)) = 0$. Note that:

$$h(\mathbf{x}(t)) = 1 - \|\mathbf{x}(t)\|_2^2 = 1 - \exp(h(\mathbf{x}_i)\Delta t_i)\|\mathbf{x}_i\|_2^2,$$

with time derivative:

$$\dot{h}(\mathbf{x}(t)) = -h(\mathbf{x}_i)\exp\left(h(\mathbf{x}_i)\Delta t_i\right)\|\mathbf{x}_i\|_2^2,$$

for $t \in [t_i, t_i + 1)$. Within the safe set we have that $\dot{h}(\mathbf{x}(t)) \leq 0$, such that $h(\mathbf{x}(t))$ is monotonically decreasing in time. The safety of $\mathcal{C}$ implies $h(\mathbf{x}(t))$ is lower bounded by 0, and thus we can conclude that $\lim_{t\to\infty} h(\mathbf{x}(t))$ exists. Assume that this limit is some value $0 < c < 1$. Thus for any $\delta > 0$, there exists $T > 0$ such that for $t > T$, $h(\mathbf{x}(t)) < c + \delta$. Since there are an infinite number of events, we deduce there exists $t_i > T$ such that $h(\mathbf{x}(t_i)) < c + \delta$. As $h(\mathbf{x}(t))$ is monotonically decreasing, it also follows $h(\mathbf{x}(t)) \geq c$ for all t. This implies:

$$\dot{h}(\mathbf{x}(t)) \leq -c\exp\left(c\Delta t_i\right)(1 - (c + \delta)) \leq -c + c^2 + c\delta < 0.$$

for $t \geq t_i$ where δ is chosen small enough to enforce the strict inequality with 0. Thus between two events we have:

$$h(\mathbf{x}(t_{i+1})) \leq h(\mathbf{x}(t_i)) + \tau(-c + c^2 + c\delta),$$

where τ is the assumed MIET. Choosing $\delta < \tau(c - c^2)/(1 + \tau c)$ implies $h(\mathbf{x}(t_{i+1})) < c$, contradicting the assumption that $c \neq 0$ (and maintaining the assumption on the existence of τ).

To complete the proof, note $\mathbf{e}(t_i) = \mathbf{0}$ and take the second-order (one-sided) Taylor expansion of $\|\mathbf{e}(t)\|_2^2$ at $t = t_i$:

$$\begin{aligned}
\|\mathbf{e}(t)\|_2^2 &= \left(\dot{\mathbf{e}}(t_i)^\top \dot{\mathbf{e}}(t_i)\right)(t - t_i)^2 + \mathcal{O}((t - t_i)^3) \\
&= (1 + k(\mathbf{x}(t_i)))\|\mathbf{x}(t_i)\|_2^2(t - t_i)^2 + \mathcal{O}((t - t_i)^3) \\
&\geq \|\mathbf{x}(t_i)\|_2^2(t - t_i)^2 - c_3(t - t_i)^3,
\end{aligned}$$

with $c_3 > 0$. The first term in the inequality follows from $k(\mathbf{x}(t_i)) \geq 0$ for $\mathbf{x}(t_i) \in \mathcal{C}$. The second term follows if we view $\frac{\mathrm{d}^3}{\mathrm{d}t^3}\|\mathbf{e}(t)\|_2^2$ as a continuous function of the state, which remains within the compact set $\mathcal{C}$. Then $\frac{\mathrm{d}^3}{\mathrm{d}t^3}\|\mathbf{e}(t)\|_2^2$ will be bounded for all time as $\mathcal{C}$ is forward invariant. We can use this bound in conjunction with Lagrange's Remainder Formula [Abb01] to assert the existence of c_3.

At trigger time t_i, let $h(\mathbf{x}(t_i)) = \epsilon_i$ with $\epsilon_i > 0$ arbitrarily small due to the existence of infinite triggers and convergence of h. This implies $\|\mathbf{x}(t_i)\|_2^2 = 1 - \epsilon_i$. Let $n \in \mathbb{N}$ be such that $\frac{1}{c_3} < n\tau$ and define $t_i^\star = t_i + \frac{1}{n}\left(\frac{1-\epsilon_i}{c_3}\right)$, noting $t_i^\star < t_i + \tau$. It follows from the Taylor expansion that:

$$\|\mathbf{e}(t_i^\star)\|_2^2 \geq \frac{(1-\epsilon_i)^3}{c_3^2}\frac{n-1}{n^3}.$$

As ϵ_i can be chosen arbitrarily small, we choose it such that:

$$(1-\epsilon_i)^3 \geq \frac{\sigma^2 n^3}{4r^6(n-1)}\epsilon_i^2,$$

which indicates that:

$$2r^3\|\mathbf{e}(t_i^\star)\|_2 \geq \sigma|h(\mathbf{x}(t_i))| \geq \sigma|h(\mathbf{x}(t^\star))|,$$

as $h(\mathbf{x}(t))$ is monotonically decreasing. As $t_i^\star < t_i + \tau$, this contradicts that τ is the MIET. Figure 6.1 shows the number of events as a function of time and distance from the barrier. The blue curves in Figure 6.2 correspond to the interevent times. □

Throughout the proof of Lemma 6.3.1, a critical concept arises at multiple steps. The fact that the state dynamics (6.20) are not required to converge to $\mathbf{0}$ at the boundary of the safe set $\mathcal{C}$ leads

to the derivative of the measurement error, $\frac{\mathrm{d}}{\mathrm{d}t}\|\mathbf{e}(t_i)\|_2$, being uniformly lower bounded at event times t_i, which, together with the convergence of h to 0, cf. Figure 6.1, leads in turn to arbitrarily small interevent times. In particular, the dynamics are allowed to evolve tangentially to the boundary of the safe set, which leads to growing measurement error while moving arbitrarily close to the 0-level set of h.

As the original controller may have additional objectives beyond safety (such as stabilization), it is desirable that the event-triggered implementation not completely eliminate tangential motion near the boundary that may be necessary to achieve the other objectives. To accommodate this, we will introduce a trigger law that limits dynamic evolution tangential to the boundary of the safe set.

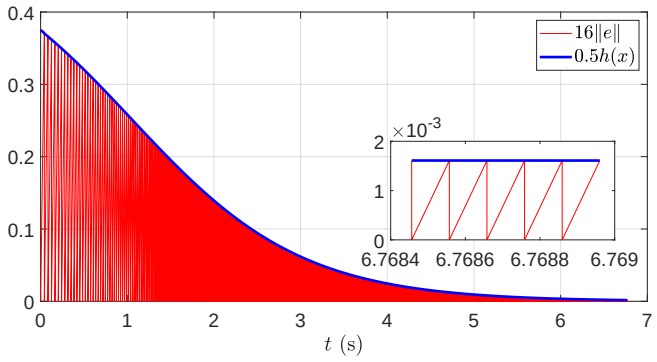


Figure 6.1: Simulation results for the system (6.20) using the trigger law (6.21). Even as the boundary of the safe set is approached $h(t) \to 0$, the growth rate of the error does not diminish, leading to arbitrarily small interevent times.

6.4 Event-Triggered Safety

In this section, we propose an alternative trigger to the one formulated in Section 6.3 that ensures a MIET exists. To resolve the issues highlighted in the preceding example, we introduce the following definition:

Definition 6.4.1 (*Strong ISSf Barrier Property*)**.** *An ISSf-BF h satisfies the strong ISSf barrier property if there exists $d \in \mathbb{R}$ with $d > 0$ such that for all $\mathbf{x}, \mathbf{e} \in \mathbb{R}^n$:*

$$\frac{\partial h}{\partial \mathbf{x}}(\mathbf{x})\mathbf{f}(\mathbf{x}, \mathbf{k}(\mathbf{x}+\mathbf{e})) \geq -\alpha(h(\mathbf{x})) + d - \iota(\|\mathbf{e}\|_2), \tag{6.22}$$

This property introduces a positive constant, d, into the ISSf-BF condition (6.14). In the presence of zero measurement error, this enforces that the state dynamics must lie in the interior of the tangent cone [BM07] when on the boundary of the safe set $\mathcal{C}$. It also enforces that $\frac{\mathrm{d}}{\mathrm{d}t}|\sigma h(\mathbf{x}(t_i))|$ will be greater than a positive constant as we approach the boundary, similarly to $\frac{\mathrm{d}}{\mathrm{d}t}\|\mathbf{e}(t_i)\|_2^2$. We now show this property is sufficient to design a trigger law that ensures safety with a MIET.

Theorem 6.4.2 (Trigger Law for Safety Critical Systems)**.** *Let h be an ISSf-BF for* (6.6) *on a set $\mathcal{C} \subset \mathbb{R}^n$ defined as in* (6.11a)-(6.11c)*, with corresponding functions $\alpha \in \mathcal{K}_{\infty,e}$ and $\iota \in \mathcal{K}_\infty$. Let $\beta \in \mathcal{K}_{\infty,e}$, $\sigma \in (0, 1]$. If the following assumptions hold:*

(i) h satisfies the strong ISSf barrier property for a constant $d \in \mathbb{R}$, $d > 0$,

(ii) ι is Lipschitz continuous with Lipschitz constant L_ι,

(iii) there exists $F \in \mathbb{R}$, $F > 0$, such that for all $\mathbf{x}, \mathbf{e} \in \mathbb{R}^n$:

$$\|\mathbf{f}(\mathbf{x}, \mathbf{k}(\mathbf{x}+\mathbf{e}))\|_2 \leq F,$$

(iv) $\beta(r) \geq \alpha(r)$ *for all* $r \in \mathbb{R}$,

then the trigger law:

$$t_{i+1} = \min\left\{t \geq t_i \mid \iota(\|\mathbf{e}(t)\|_2) = \beta(h(\mathbf{x}(t))) - \alpha(h(\mathbf{x}(t))) + \sigma d\right\}, \tag{6.23}$$

deployed recursively enforces:

$$\dot{h}(\mathbf{x}, \mathbf{e}) = \frac{\partial h}{\partial \mathbf{x}}(\mathbf{x})\mathbf{f}(\mathbf{x}, \mathbf{k}(\mathbf{x} + \mathbf{e})) \geq -\beta(h(\mathbf{x})), \tag{6.24}$$

thus rendering the set $\mathcal{C}$ *safe and asymptotically stable. Furthermore, there exists a MIET given by:*

$$t_{i+1} - t_i \geq \tau \triangleq \frac{\sigma d}{L_\iota F}, \quad i \in \mathbb{N}.$$

Before proving the result, we make a few observations regarding its assumptions. Assumption 3 on the boundedness of the dynamics need not hold over the entire state space for safety, but can hold for $(\mathbf{x}, \mathbf{e}) \in C \times \mathbb{R}^n$. Furthermore, if C is compact, the trigger law enforces the existence of a compact set $E \subset \mathbb{R}^n$ such that $\mathbf{e} \in E$. Thus, the continuity of $\mathbf{f}$ and $\mathbf{k}$ implies the assumption is satisfied on $C \times E$, which also would be sufficient for the result to hold. Assumption 4 ensures the right-hand side of the equality in the trigger will always be positive. β can be thought of as a tuning function, which can practically raise interevent times (but not the MIET) at the expense of less "braking" within the safe set and convergence outside of the safe set. One choice is $\beta = \alpha$, in which case interevent times are lowered for more braking and faster convergence. We additionally note that this trigger law can be used in conjunction with (6.10) to jointly achieve event-triggered stabilization and safety.

Proof. To see the set $\mathcal{C}$ is rendered safe, observe that:

$$\begin{aligned}\dot{h} = \frac{\partial h}{\partial \mathbf{x}}(\mathbf{x})\mathbf{f}(\mathbf{x}, \mathbf{k}(\mathbf{x}+\mathbf{e})) &\geq -\alpha(h(\mathbf{x})) + d - \iota(\|\mathbf{e}\|_2) \\ &\geq -\alpha(h(\mathbf{x})) + d - (\beta(h(\mathbf{x})) - \alpha(h(\mathbf{x})) + \sigma d), \\ &= -\beta(h(\mathbf{x})) + (1-\sigma)d \geq -\beta(h(\mathbf{x})),\end{aligned}$$

implying the barrier condition (6.12) is satisfied and $\mathcal{C}$ is safe.

To see the interevent time is lower bounded, observe that

$$\begin{aligned}\|\mathbf{e}(t)\|_2 &= \left\| \mathbf{e}(0) + \int_{t_i}^{t} (-\mathbf{f}(\mathbf{x}(\tau), \mathbf{e}(\tau)))\mathrm{d}\tau \right\|_2 \\ &= \left\| \int_{t_i}^{t} (-\mathbf{f}(\mathbf{x}(\tau), \mathbf{e}(\tau)))\mathrm{d}\tau \right\|_2 \\ &\leq \int_{t_i}^{t} F \mathrm{d}\tau.\end{aligned}$$

This inequality together with the trigger law (6.23) yields:

$$\begin{aligned}t_{i+1} &\geq \min\left\{ t \geq t_i \mid L_\iota \|\mathbf{e}(t)\|_2 = \sigma d \right\}, \\ &\geq \min\left\{ t \geq t_i \mid L_\iota F(t - t_i) = \sigma d \right\} \\ &= \frac{\sigma d}{L_\iota F} + t_i,\end{aligned}$$

ensuring the desired result. This in conjunction with the barrier function condition implies $\mathcal{C}$ is asymptotically stable. □

In the case that an ISSf-BF h does not satisfy the strong barrier property, an auxiliary ISSf-

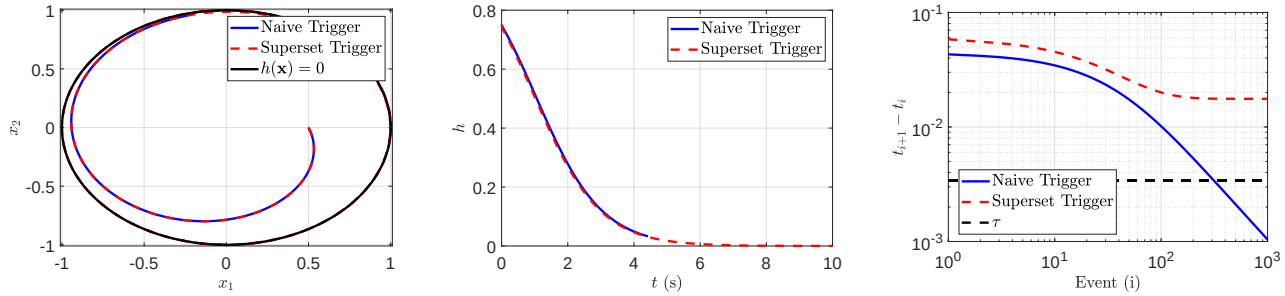


Figure 6.2: Simulation results demonstrating safety achieved with an event-triggered controller. **(Left)** State trajectories for both triggers remain within the safe set for the length of the simulation. **(Center)** The value of the ISSf-BF h remains above zero for the length of the simulation, corresponding to the system remaining safe. **(Right)** The interevent times of the two trigger laws. The interevent times of the trigger law (6.21) decreases towards 0 as predicted by Lemma 6.3.1 while the trigger law (6.23) satisfies the theoretical bound.

BF, h_b, satisfying the strong ISSf barrier property can be synthesized via h at the expense of guaranteeing only a larger set is kept safe:

Theorem 6.4.3 (Strong ISSf Barrier Property in Supersets)**.** *Let h be an ISSf-BF for* (6.6) *on a set $\mathcal{C} \subset \mathbb{R}^n$ defined as in* (6.11a)-(6.11c)*, with corresponding functions $\alpha \in \mathcal{K}_{\infty,e}$ and $\iota \in \mathcal{K}_\infty$. Then the function h_b defined as $h_b(\mathbf{x}) = h(\mathbf{x}) + b$, with $b \in \mathbb{R}$, $b > 0$, is an ISSf-BF satisfying the strong ISSf barrier property on the set $\mathcal{C}_b$ defined as:*

$$\mathcal{C}_b \triangleq \{\mathbf{x} \in \mathbb{R}^n \mid h_b(\mathbf{x}) \geq 0\} \tag{6.25}$$

Proof. Observe that:

$$\begin{aligned}\frac{\partial h_b}{\partial \mathbf{x}}(\mathbf{x})\mathbf{f}(\mathbf{x}, \mathbf{k}(\mathbf{x}+\mathbf{e})) &= \frac{\partial h}{\partial \mathbf{x}}(\mathbf{x})\mathbf{f}(\mathbf{x}, \mathbf{k}(\mathbf{x}+\mathbf{e})) \\ &\geq -\alpha(h(\mathbf{x})) - \iota(\|\mathbf{e}\|_2) \geq -\alpha(h_b(\mathbf{x}) - b) + \alpha(-b) \\ &\quad - \alpha(-b) - \iota(\|\mathbf{e}\|_2) = -\alpha_b(h_b(\mathbf{x})) + d_b - \iota(\|\mathbf{e}\|_2),\end{aligned}$$

where $\alpha_b \in \mathcal{K}_{\infty,e}$ is defined as $\alpha_b(r) = \alpha(r-b) - \alpha(-b)$ and $d_b = -\alpha(-b) > 0$. □

Thus an ISSf-BF can be used with the trigger law (6.23) by enlarging the safe set an arbitrarily small amount:

Corollary 6.4.4 (Superset Trigger Law)**.** *If h is an ISSf-BF for* (6.6) *on the set $\mathcal{C}$ satisfying Assumptions (2-4) of Theorem 6.4.2, then h_b is an ISSf-BF for* (6.6) *on the set $\mathcal{C}_b$ satisfying Assumptions (1-4) of Theorem 6.4.2 such that the corresponding trigger law renders $\mathcal{C}_b$ safe and asymptotically stable with a MIET.*

This is effectively an instance of Input-to-State Safety, in which case the original safe set $\mathcal{C}$ defined via h becomes an ISSf safe set. We note that the larger the set is made (via a larger choice of b), the larger the MIET will be. This effectively highlights a trade-off that arises in the context of safety but not in stabilization: allowing motion near the boundary of a safe set requires additional relaxations to achieve the additional desirable property of the MIET.

To verify the ability of this trigger to keep the system safe and have a MIET, we simulated the system (6.19) using both the trigger law (6.21) and the trigger law (6.23). The results of these simulations can be seen in Figure 6.2. We see that although both systems are kept safe, the trigger law not using the strong ISSf barrier property has interevent times that approach 0.

Acknowledgements

This chapter, in full, is a reprint of the material [TOCA21] as it appears in 'Safety-critical event triggered control via input-to-state safe barrier functions' by A. J. Taylor, P. Ong, J. Cortés, and A.D. Ames, in IEEE Control Systems Letters, 2021. The dissertation author was one of the two

primary investigators and authors of this paper. The work in this chapter was partially supported by NSF Award ECCS-1917177.

Chapter 7

Opportunistic Human for Objective Prioritization

This chapter proposes opportunistic state-triggered strategies for solving convex multiobjective optimization problems that involve human-robot interaction. The robot is aware of the multiple objective functions defining the problem, but requires human input to find the most desirable Pareto solution. In order to avoid overloading the human with queries, we view her as a limited resource to the robot, and design event-triggered controllers that opportunistically prescribe the information exchanges among them. We consider various models of human performance, starting with an ideal one where queries are responded instantaneously, and later considering constraints on the response time and the interaction frequency. For each model, we formally establish the asymptotic convergence to the desired optimizer and rule out the existence of Zeno behavior.

7.1 Interactive Multiobjective Optimization

Consider a human-robot system that seeks to find the minimizer to a vector-valued, continuously differentiable function, function $f : \mathbb{R}^n \to \mathbb{R}^m$, i.e. $\min_{x \in \mathbb{R}^n} f(x)$. Each of the components of f represents a goal the robot is trying to achieve, and x represents the decision variable. A point $x_{\text{po}} \in \mathbb{R}^n$ is a solution to the minimization problem if there does not exist $x \in \mathbb{R}^n$ with $f_i(x) \leq f_i(x_{\text{po}})$ for all $i \in [m]$ with at least one inequality being strict. These solutions, called Pareto points, capture the fact that improving the minimization of one component of f cannot be done without increasing the value of another. In principle, there exist multiple Pareto points corresponding to the different trade-offs in optimizing the components of f.

Many multiobjective optimization problems find applications in practical scenarios involving control formulations, cf. [PD18]. Usually, the decision variable corresponds to the control input. For example, [PSOB+17] considers an MPC formulation in autonomous driving where the control u can affect both the arrival time and energy consumption. When doing motion planning, the robot must consider the different conflicting factors involved, see [SG17].

7.1.1 Interactive Approach

In general, finding the whole set of Pareto points is computationally expensive. Furthermore, additional considerations might make some Pareto points more desirable than others. One way to address this is via the interactive approach, where a human is involved in determining the desirable outcome. This has the added benefits of reduction in computational resource usage, improved desirability of the obtained solution, and adaptability to different scenarios.

Consider the following human-robot model. The robot has first-order fully actuated dynam-

ics and is assumed to have knowledge of the component objective functions in f. A human operator assists the robot in selecting the most appropriate Pareto point. As is commonly done in trade-off approaches to multiobjective optimization problems, we assume the human has a scalar-valued, continuously differentiable function $v : \mathbb{R}^m \to \mathbb{R}$ that ranks the different outcomes, i.e., $v(f(x))$ represents the 'value' that the human gives to the outcome $f(x)$ achieved at $x \in \mathbb{R}^n$. This function can then be used to establish a preference among all Pareto points. However, the function v is implicit, meaning that the human does not know it in closed form, but can respond to queries about it. Specifically, we model the human as being able to express preferences about an outcome being better than another one, and we abstract this with gradient information of v: if the robot queries the human about its current value $f(x)$, the human can provide the value $\nabla v(f(x))$, indicating the direction of change in which outcomes are more highly valued.

The optimization problem consists of maximizing $v \circ f$. For convenience, we instead formulate it as a minimization problem by considering the cost function $c = -v$. The problem to solve is then $\min_{x \in \mathbb{R}^n} (c \circ f)(x)$. We assume the composition function $c \circ f$ is strictly convex, and that the problem has a unique minimizer x^*, which we can find via the gradient descent algorithm

$$\dot{x} = -\nabla(c \circ f)(x)^\top = -(\nabla c(f(x)) \boldsymbol{J}_f(x))^\top,$$

which globally asymptotically converges to x^*. This can be shown by considering the value of $c \circ f$, which strictly decreases over time along the trajectory. Note, however, that the implementation of the gradient dynamics by the robot is problematic. The robot knows the objective function f and can therefore compute its Jacobian, $\boldsymbol{J}_f$. However, $\nabla c \circ f$ can only be provided by the human because only she knows the cost function. Therefore, executing the dynamics would require the human to

continuously relay preference information to the robot, which is not feasible. The discretization of the dynamics with a constant stepsize would make its implementation plausible, albeit it still requires constant, periodic human involvement. Given that the stepsize needs to be sufficiently small to guarantee convergence for arbitrary initial conditions, this may still impose an unnecessary burden on the human. The basic premise of the paper to tackle this is to endow the robot with criteria that allow it to determine, in an opportunistic fashion, when to query the human to avoid her unnecessary involvement.

7.1.2 Problem Statement

Motivated by Section 7.1.1, we consider the following gradient dynamics, which discretizes the human component but maintains the continuous evolution of the robot component,

$$\dot{x} = -(\nabla c(f(x_k))\boldsymbol{J}_f(x))^\top, \quad t_k \leq t < t_{k+1}, \tag{7.1}$$

where x_k is shorthand notation to represent $x(t_k)$. Under this dynamics, the human operator only needs to assess the robot performance at the time instants $\{t_k\}_{k=0}^{\infty}$. Our goal in this paper is to design triggers that the robot can evaluate on its own to determine this sequence of times efficiently, while still guaranteeing the asymptotic convergence to the desired solution and the feasibility of the resulting implementation (i.e., interevent times are uniformly lower bounded and hence the implementation is free of Zeno behavior). What makes the trigger design and analysis different from other event-triggered control formulations is that the resource to be aware of here is the human. In particular, the fact that the preference function c is unknown to the robot (even the human does not explicitly know it, as discussed above) and the various human behaviors (e.g., unable to comply

with multiple rapidly succeeding requests for information) detailed later in Sections 7.2 and 7.3 rule out the use of standard results in event-triggered control.

In this paper, we consider different models for human behavior, starting with an ideal model where the human can respond instantaneously. We then move to consider models with timing constraints, such as when the human needs time to rest between queries, may take time to respond to a query, or a combination thereof. For each model, we propose a trigger design that satisfies the above criteria.

Remark 7.1.1. ***(Strict Convexity of the Composition Function).*** Note that, if all the component functions of f are strictly convex and the cost function c is both strictly increasing in each component (which is reasonable, given that the human seeks to minimize each individual component) and strictly convex, then $c \circ f$ is also strictly convex. •

7.2 Event-Triggered Design: Ideal Human

Here we synthesize a triggering condition for the robot that allows it to efficiently query the human about her preferences on the optimization of the vector-valued objective function. We assume that the human performance is ideal, meaning that the human can respond to queries immediately, i.e., there is no delay in obtaining the value of $\nabla c \circ f$.

Our trigger design is based on analyzing the evolution of the cost function evaluated on the objectives towards its optimal value. We consider

$$V(x) = c(f(x)) - c(f(x^*)). \tag{7.2}$$

Note that V is radially unbounded and has compact sublevel sets due to our assumptions on strict convexity and existence of a unique minimizer. For convenience, we use the shorthand notation $w = \nabla c \circ f : \mathbb{R}^n \to \mathbb{R}$, and, for $k \in \{0\} \cup \mathbb{N}$ and $t \geq t_k$, we let $\Delta x_k = x(t) - x_k$ denote the error between the state at time t and the state when the gradient was last updated at time t_k. The next result identifies a gradient update triggering condition that ensures V decreases on a neighborhood of the optimizer.

Proposition 7.2.1. ***(Trigger for Ideal Human).*** *Consider the event-triggered human-robot system* (7.1) *and let $x_k \neq x^*$ be the state when the gradient information was last updated. Let $\mathcal{V} \subseteq \mathbb{R}^n$ be a neighborhood of the optimizer such that $x_k \in \mathcal{V}$ and let L_c be the Lipschitz constant of $\nabla c \circ f$ over $\mathcal{V}$. For $\sigma \in (0, 1]$, let t_{k+1} be determined by*

$$t_{k+1} = \min \left\{ t \geq t_k \mid \|\Delta x_k\| = \frac{\sigma \|\nabla c(f(x_k)) J_f(x)\|}{L_c \|J_f(x)\|} \right\}. \tag{7.3}$$

If $S_k = \{x \in \mathbb{R}^n \mid V(x) \leq V(x_k)\} \subseteq \mathcal{V}$, then for all $t \in [t_k, t_{k+1})$, we have

$$\frac{d}{dt} V(x(t)) < -\frac{1 - \sigma}{(1 + \sigma)^2} \|\nabla c(f(x(t))) J_f(x(t))\|^2. \tag{7.4}$$

Proof. First we note that V is positive definite because x^* is unique. Let $\Delta w_k = w(x) - w(x_k)$.

Then the time derivative of V at $t \in [t_k, t_{k+1})$ is

$$\begin{aligned}\frac{d}{dt}V(x(t)) &= (w(x)J_f(x))\dot{x} = -(w(x)J_f(x))(w(x_k)J_f(x))^\top \\ &= -((w(x_k) + \Delta w_k)J_f(x))(w(x_k)J_f(x))^\top \\ &\leq -\|w(x_k)J_f(x)\|^2 + \|\Delta w_k\|\|J_f(x)\|\|w(x_k)J_f(x)\| \\ &\leq -\|w(x_k)J_f(x)\|^2 + L_c\|\Delta x_k\|\|J_f(x)\|\|w(x_k)J_f(x)\|.\end{aligned}$$

The last inequality relies on the assumption that w is Lipschitz on S_k with constant L_c so that $\|\Delta w_k\| \leq L_c\|\Delta x_k\|$. Since $\|\Delta x_k\| = 0$ at time t_k, and given the definition (7.3) of t_{k+1}, we have $\|\Delta x_k\| < \sigma\left(\frac{\|\nabla c(f(x_k))J_f(x)\|}{L_c\|J_f(x)\|}\right)$ on the interval $[t_k, t_{k+1})$. We can then deduce that

$$\frac{d}{dt}V(x(t)) < -(1-\sigma)\|w(x_k)J_f(x(t))\|^2. \tag{7.5}$$

This shows that $\frac{d}{dt}V(x(t))$ is negative, and hence the set S_k is invariant under (7.1). Next, we find a relationship between $\|w(x)J_f(x)\|$ and $\|w(x_k)J_f(x)\|$ as follows,

$$\begin{aligned}\|w(x)J_f(x)\| &\leq \|w(x_k)J_f(x)\| + \|\Delta w_k J_f(x)\| \\ &\leq \|w(x_k)J_f(x)\| + L_c\|\Delta x_k\|\|J_f(x)\| \\ &< (1+\sigma)\|w(x_k)J_f(x)\|,\end{aligned} \tag{7.6}$$

where we have again used the bound on $\|\Delta x_k\|$ from the design to bound the last inequality. The result now follows by substituting (7.6) into (7.5). □

Note that the hypothesis that $S_k \subseteq \mathcal{V}$ is easily satisfied given that V is radially unbounded.

Proposition 7.2.1 provides a trigger design (7.3) under which the function V is strictly monotonically decreasing. One important feature of the trigger design is that the design only relies on $\nabla c(f(x))$ at $x = x_k$, which is available to the robot. This ensures that the monitoring of this condition can be evaluated during each iteration independently by the robot, i.e., the human is only queried at discrete instants of time. Nevertheless, we cannot yet conclude that the optimizer is asymptotically stable. The reason for this is that we first need to discard Zeno behavior, i.e., the possibility of (7.3) inducing an infinite number of trigger updates in a finite amount of time. To do so, it is useful to characterize how the system state discretization error, $\|\Delta x_k\|$, evolves during interexecution periods, i.e., between consecutive updates, independently of how triggering times are determined.

Lemma 7.2.2. ***(State Deviation Bound).*** *Consider the event-triggered human-robot system* (7.1) *and let x_k be the state when the gradient information was last updated. For any triggering time t_{k+1} such that $x(t) \in S_k$ for all $t \in [t_k, t_{k+1})$, the system state discretization error satisfies*

$$\|\Delta x_k\| \leq \phi_k(t - t_k)\|\dot{x}\| \tag{7.7}$$

during the interexecution period $[t_k, t_{k+1})$, where $\phi_k(t) = \frac{1}{M_k}(e^{M_k t} - 1)$ with $M_k = \max_{x \in S_k} \|\nabla^2(w(x_k)f(x))\|$.

Proof. We first note that the case where $x_k = x^*$ is trivial. Then for $x_k \neq x^*$, we must have that

$\dot{x} \neq 0$ at time t_k and we can examine the dynamics of $\|\Delta x_k\|/\|\dot{x}\|$.

$$\begin{aligned}
\frac{d}{dt}\frac{\|\Delta x_k\|}{\|\dot{x}\|} &= \frac{d}{dt}\frac{(\Delta x_k^\top \Delta x)^{1/2}}{(\dot{x}^\top \dot{x})^{1/2}} \\
&= \frac{(\Delta x_k^\top \Delta x_k)^{-1/2}\Delta x_k^\top \Delta \dot{x}_k (\dot{x}^\top \dot{x})^{1/2}}{\dot{x}^\top \dot{x}} - \frac{(\dot{x}^\top \dot{x})^{-1/2}\dot{x}^\top \ddot{x}(\Delta x_k^\top \Delta x_k)^{1/2}}{\dot{x}^\top \dot{x}} \\
&= \frac{\Delta x_k^\top \Delta \dot{x}_k}{\|\Delta x_k\|\|\dot{x}\|} - \frac{\dot{x}^\top \ddot{x}\|\Delta x_k\|}{\|\dot{x}\|^3} \\
&\leq \frac{\|\Delta x_k\|\|\Delta \dot{x}_k\|}{\|\Delta x_k\|\|\dot{x}\|} + \frac{\|\dot{x}\|\|\ddot{x}\|\|\Delta x_k\|}{\|\dot{x}\|^3} \\
&= 1 + \frac{\|\Delta x_k\|}{\|\dot{x}\|}\frac{\|\ddot{x}\|}{\|\dot{x}\|},
\end{aligned} \tag{7.8}$$

where in the last step we have used the fact that $\Delta \dot{x}_k = \dot{x}$. Now, we define the function $V_k(x) = w(x_k)f(x)$, and write

$$\ddot{x} = \frac{d}{dt}(-w(x_k)J_f(x))^\top = -\frac{d}{dt}(\nabla V_k(x))^\top = -\nabla^2 V_k(x)\dot{x}.$$

We then find that $\|\ddot{x}\| \leq \|\nabla^2 V_k(x)\|\|\dot{x}\| \leq M_k\|\dot{x}\|$. We use this bound in (7.8) to obtain

$$\frac{d}{dt}\frac{\|\Delta x_k\|}{\|\dot{x}\|} \leq 1 + M_k\frac{\|\Delta x_k\|}{\|\dot{x}\|}. \tag{7.9}$$

Now, because $\phi_k(t - t_k)$ satisfies the differential equation $\dot{\phi}_k = 1 + M_k\phi_k$ with the initial condition $\phi_k(0) = 0$, we have that $\phi_k(t - t_k) \geq \frac{\|\Delta x_k\|}{\|\dot{x}\|}$ by the Comparison Lemma, cf. [Kha02, Lemma 3.4]. Finally, we show that the bound (7.7) is valid for all time $[t_k, t_{k+1})$ by ruling out the possibility that $\|\dot{x}\| = 0$ along the trajectory. This can be proven by contradiction. Let $t_{\text{stop}} > t_k$ denote the first instance when $\|\dot{x}(t_{\text{stop}})\| = 0$. The resulting bound (7.7) is then valid for the duration $[t_k, t_{\text{stop}})$. In this duration, we note $\phi_k(t - t_k)$ is upper bounded by a positive value $\phi_k(t_{\text{stop}} - t_k)$ because it is

strictly increasing. Therefore, as $\|\dot{x}(t)\| \to \|\dot{x}(t_{\text{stop}})\| = 0$, we have $\|\Delta x_k\| \to 0$. This implies $\|\Delta x_k\| = 0$ at $t = t_{\text{stop}}$, i.e., $x(t_{\text{stop}}) = x_k$. This contradicts the fact $\dot{x} \neq 0$ at $x = x_k$, concluding the proof. □

With the bound on how the state discretization error evolves given in Lemma 7.2.2, we next establish a lower bound on the interexecution time.

Proposition 7.2.3. ***(Lower Bound on Interexecution Time).*** *For the event-triggered human-robot system* (7.1) *with updates determined according to* (7.3) *and initial condition* x_0*, if* $S_0 \subseteq \mathcal{V}$*, then the interexecution time is lower bounded as*

$$t_{k+1} - t_k \geq \tau_\sigma^{\text{i}} := \frac{1}{M_0} \ln\left(1 + \frac{M_0 \sigma}{L_c J_{\max}}\right) \tag{7.10}$$

for all $k \in \{0\} \cup \mathbb{N}$ *with* $J_{\max} = \max_{x \in S_0} \|J_f(x)\|$.

Proof. We aim to show that there is a finite lower bound to the time it takes before the condition defining the next update time in (7.3) is met. For convenience, notice that this condition can be equivalently rewritten as

$$\frac{\|\Delta x_k\|}{\|\dot{x}\|} = \frac{\sigma}{L_c \|J_f(x)\|}. \tag{7.11}$$

Then, by continuity, it takes longer to evolve from $\frac{\|\Delta x_k\|}{\|\dot{x}\|} = 0$ to $\frac{\|\Delta x_k\|}{\|\dot{x}\|} = \frac{\sigma}{L_c J_{\max}}$ than it takes to reach condition (7.11).

Now using the result (7.4), because $S_0 \subseteq \mathcal{V}$ we can deduce through induction that that $S_{k+1} \subset S_k \in \mathcal{V}$ for all $k \in \{0\} \cup \mathbb{N}$. From this, we note here as well that $M_k \leq M_0$ for all $k \in \{0\} \cup \mathbb{N}$. By the Comparison Lemma, we can show that $\phi_k(t - k) \leq \phi_0(t - t_k)$. Together

with (7.7), we have $\frac{\|\Delta x_k\|}{\|\dot{x}\|} \leq \phi_0(t - t_k)$, so it takes an even shorter time for $\phi_0(t - t_k)$ to reach $\frac{\sigma}{L_c J_{\max}}$, which is precisely τ_σ^{i}. □

The lower bound on the interexecution time in Proposition 7.2.3 rules out the possibility of Zeno behavior. Combining this result with Proposition 7.2.1, we deduce asymptotic convergence towards the desired optimizer.

Corollary 7.2.4. ***(Asymptotic Stability – Ideal Human Design).*** *For the event-triggered human-robot system* (7.1) *with updates determined according to* (7.3)*, the optimizer x^* is asymptotically stable, with $\mathcal{X}_0 = \{x_0 \in \mathbb{R}^n \mid \mathcal{S}_0 \subseteq \mathcal{V}\}$ contained in its region of attraction. Moreover, if $c \circ f$ is strongly convex with constant $\mu > 0$ on $\mathcal{V}$, then given an initial condition $x_0 \in \mathcal{X}_0$,*

$$V(x(t)) \leq \begin{cases} V(x_k) e^{-2\mu \int_{t_k}^{t} \frac{1-\xi_0(s-t_k)}{(1+\xi_0(s-t_k))^2} ds} & , \ t \in [t_k, t_k + \tau_\sigma^{\mathrm{i}}] \\ V(x(t_k + \tau_\sigma^{\mathrm{i}})) e^{-\frac{2\mu(1-\sigma)}{(1+\sigma)^2}(t - t_k - \tau_\sigma^{\mathrm{i}})} & , \ t \in [t_k + \tau_\sigma^{\mathrm{i}}, t_{k+1}) \end{cases} \tag{7.12}$$

for all $k \in \{0\} \cup \mathbb{N}$, where $\xi_0(t) = L_c J_{\max} \phi_0(t)$. As a consequence, the certificate satisfies for all t,

$$V(x(t)) \leq V(x_0) e^{-\frac{2\mu(1-\sigma)}{(1+\sigma)^2} t}, \tag{7.13}$$

and the optimizer is exponentially stable for $\sigma \in (0, 1)$.

Proof. Asymptotic stability follows directly from Propositions 7.2.1 and 7.2.3. Next, similar to the derivation of (7.4) in Proposition 7.2.1, we can use Lemma 7.2.2 to bound the time derivative of

the Lyapunov function as

$$\frac{d}{dt}V(x(t)) \leq -\frac{1-\xi_0(t-t_k)}{\left(1+\xi_0(t-t_k)\right)^2}\|\nabla c(f(x(t)))\boldsymbol{J}_f(x(t))\|^2$$

for the interval $[t_k, t_{k+1})$. Now, if $c \circ f$ is strongly convex, using (2.3) to bound the inequality above and also (7.4), we find that for $t \in [t_k, t_{k+1})$,

$$\frac{d}{dt}V(x(t)) \leq -2\mu\frac{1-\min\{\sigma, \xi_0(t-t_k)\}}{\left(1+\min\{\sigma, \xi_0(t-t_k)\}\right)^2}V(x(t)).$$

To find what the min function evaluates to, we use the fact that for $t \in [t_k, t_k + \tau_\sigma^{\mathrm{i}})$, if $t - t_k \leq \tau_\sigma^{\mathrm{i}} = \frac{1}{M_0}\ln\left(1+\frac{M_0\sigma}{L_c J_{\max}}\right)$, then $\xi_0(t-t_k) \leq \sigma$. As a result, we separate the intervals into two accordingly to use the better bound. Using the Comparison Lemma, we get the bound (7.12). Finally, (7.13) follows by using (7.4) as the bound on the Lyapunov function's time derivative along the trajectory. Next, we note that from the strong convexity of function V, there exists $M \geq \mu > 0$ such that for all $x \in S_0$, $\frac{\mu}{2}\|x-x^*\|^2 \leq V(x) \leq \frac{M}{2}\|x-x^*\|^2$. As a result, we deduce from (7.13),

$$\|x(t)-x^*\| \leq \sqrt{\frac{M}{m}}\|x_0-x^*\|e^{-\frac{\mu(1-\sigma)}{(1+\sigma)^2}t}, \tag{7.14}$$

and exponential stability is proven. □

Corollary 7.2.4 shows that one can discretize the human component of the continuous-time gradient descent of the robot motion in an opportunistic fashion while guaranteeing convergence to the desired outcome. Our results show that, under the trigger design (7.3), the robot can determine when to query an ideal human operator for gradient information: Proposition 7.2.1 states that the

design choice of $\sigma \in (0, 1)$ affects the magnitude of the time derivative of V, cf. (7.4), and therefore, the speed of convergence to the optimizer. At the same time, Proposition 7.2.3 suggests that σ affects the amount of trigger updates, cf. (7.10), and therefore the amount of human workload. The choice of σ can therefore be adjusted depending on the model of human performance, an issue that we address in the following section.

Remark 7.2.5. ***(Generalizations of Corollary 7.2.4).*** Corollary 7.2.4 can be generalized in a number of ways. One can, for instance, state a global version of it provided that $c \circ f$ is globally Lipschitz by taking $\mathcal{V} = \mathbb{R}^n$. This would come at the cost of having a larger constant L_c, which in turn affects the interexecution time, making it shorter, and hence increasing the human workload. Also, if the composite function $c \circ f$ is not convex, the convergence arguments employed to establish Corollary 7.2.4 are still valid on a sufficiently small neighborhood of a local minimizer. •

7.3 Event-Triggered Design: Constraints on Human Performance

In this section, we extend our trigger design and analysis to deal with practical constraints on human performance. Specifically, we consider the following models on the amount of workload that the human can take:

(i) "Need to rest" model: the human needs some time after providing gradient information before she can respond to the next query;

(ii) "Need to think" model: the human cannot respond to queries instantaneously and instead requires some time to provide gradient information;

(iii) "Need to think then rest" model: this is a human with both "need to rest" and "need to think" constraints.

Our treatment takes advantage of the possibility of tuning the design parameter σ to handle these constraints.

7.3.1 "Need To Rest" Human

We consider the scenario where the human cannot respond in quick succession to multiple queries, i.e., after providing an answer to the robot, some time must elapse before the human can respond to another query. We assume that an upper bound $T_{\text{rest}} \geq 0$ on the time the human needs for resting is known. Our first approach to this problem tunes the design parameter σ so that the interexecution time is longer than the resting time T_{rest}.

Note that, besides σ, the parameters L_c, M_0, and $J_{\max}$ also affect the bound (7.10) on the interexecution time. As defined, the parameters M_0 and $J_{\max}$ depend on the initial condition x_0. When dealing with the constraints on human performance, it becomes relevant to explicitly calculate the bound on the interexecution time for our design, and hence we would like them to hold independently of the initial condition $x_0 \in \mathcal{X}_0$. We assume that the set of initial conditions $\mathcal{X}_0$ satisfies $\bar{S} = \{x \in \mathbb{R}^n \mid V(x) \leq \max_{x_0 \in \mathcal{X}_0} V(x_0)\} \subseteq \mathcal{V}$ (in words, the largest possible initial sublevel set of V is contained in $\mathcal{V}$). With the assumption, we can instead consider the parameters

$$\hat{M} = m \cdot \max_{x \in \mathcal{V}} \|w(x)\| \cdot \max_{x \in \mathcal{V}, i \in [m]} \|\nabla^2 f_i(x)\|,$$

$$\hat{J} = \max_{x \in \mathcal{V}} \|J_f(x)\|.$$

Note that $\hat{M} \geq M_0$ and $\hat{J} \geq J_{\max}$ for all initial conditions $x_0 \in \mathcal{X}_0$. With this in place, we define the interexecution time lower bound $\tau_\sigma^{\mathcal{V}} := \frac{1}{\hat{M}} \ln(1 + \frac{\sigma \hat{M}}{L_c \hat{J}})$, which applies to all trajectories starting in the region of attraction $\mathcal{X}_0$.

The following result shows that our trigger design of Section 7.2 can accommodate sufficiently small resting times.

Proposition 7.3.1. ***(Trigger for "Need to Rest" Human).*** *Consider the event-triggered human-robot system* (7.1) *with updates determined according to* (7.3) *and initial condition* $x_0 \in \mathcal{X}_0$. *If* $T_{\text{rest}} < \tau_1^{\mathcal{V}} := \frac{1}{\hat{M}} \ln(1 + \frac{\hat{M}}{L_c \hat{J}})$, *let* $\sigma \in (0, 1]$ *be such that*

$$\sigma \geq \frac{L_c \hat{J}}{\hat{M}} (e^{\hat{M} T_{\text{rest}}} - 1). \tag{7.15}$$

Then, $t_{k+1} \geq t_k + T_{\text{rest}}$ *for all* $k \in \{0\} \cup \mathbb{N}$. ■

The proof of this result follows from Proposition 7.2.3 since the choice of σ satisfying (7.15) makes $T_{\text{rest}} \leq \tau_\sigma^{\mathcal{V}}$. If the resting time T_{rest} does not satisfy the bound identified in Proposition 7.3.1, then we cannot guarantee that the Lyapunov function is monotonically decreasing while the human is resting and cannot answer robot queries.

To accommodate longer resting times, we explore next the possibility of allowing the Lyapunov function to increase at times during the evolution, as long as it decreases when evaluated at consecutive human's queries (note that this corresponds to a standard discrete Lyapunov function). By doing so, we develop a new trigger design that combines both event- and time-triggered ideas. Before getting into the technical exposition, we outline here the basic rationale behind this approach, cf. Figure 7.1. First, we examine the dynamics to determine a time after which the Lyapunov function V can potentially start increasing. We refer to this time as *critical*. We make the

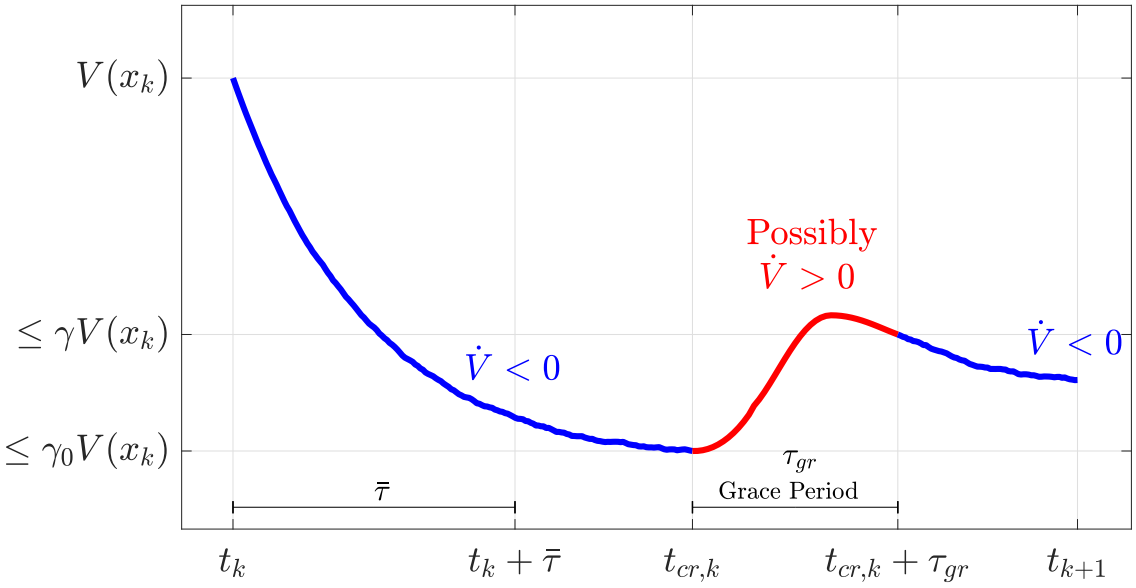


Figure 7.1: Evolution of Lyapunov function with grace period. The diagram shows an example of the evolution of the Lyapunov function using our strategy to extend the resting time.

robot wait after the critical time for a pre-specified amount of time, referred to as *grace period* (later formally introduced in Proposition 7.3.5). This period is determined in a way that ensures that the Lyapunov function remains below its value at t_k by the end of it. After the grace period, allow the system to continue without querying the gradient information only if the Lyapunov function is decreasing.

Our first result characterizes for how long without updating the human's input and by how much we can guarantee the monotonic decrease of the Lyapunov function.

Lemma 7.3.2. ***(Trigger for Critical Time).*** *Consider the event-triggered human-robot system* (7.1) *and let $c \circ f$ be strongly convex with parameter μ. Define the* critical time*,*

$$t_{\mathrm{cr},k} = \min \left\{ t \geq t_k \mid \|\Delta x_k\| = \frac{\|\nabla c(f(x_k)) J_f(x)\|}{L_c \|J_f(x)\|} \right\}. \tag{7.16a}$$

If the robot does not receive any update on the gradient information from the human during $(t_k, t_{\mathrm{cr},k}]$*, then* $t_{\mathrm{cr},k} \geq t_k + \tau_1^{\mathcal{V}}$*, and*

$$V(x(t_{\mathrm{cr},k})) \leq \gamma_0 V(x_k) \tag{7.16b}$$

with the constant $\gamma_0 = e^{-2\mu \int_0^{\tau_1^{\mathcal{V}}} \frac{1-\xi(s)}{(1+\xi(s))^2} ds} < 1$ *and* $\xi(t) = \frac{L_c \hat{J}}{\hat{M}}(e^{\hat{M}t} - 1)$*.*

Proof. Notice that the definition (7.16a) corresponds to (7.3) with $\sigma = 1$. Therefore, from (7.10), we deduce that $t_{\mathrm{cr},k} - t_k \geq \tau_1^{\mathrm{i}} \geq \tau_1^{\mathcal{V}}$. If the robot does not receive any update on the gradient information from the human during $(t_k, t_{\mathrm{cr},k}]$, then, using (7.12), we deduce that

$$V(x(t)) \leq \begin{cases} V(x_k) e^{-2\mu \int_{t_k}^{t} \frac{1-\xi(s-t_k)}{(1+\xi(s-t_k))^2} ds} & \text{if } t \in [t_k, t_k + \tau_1^{\mathcal{V}}], \\ V(x(t_k + \tau_1^{\mathcal{V}})) & \text{if } t \in [t_k + \tau_1^{\mathcal{V}}, t_{\mathrm{cr},k}), \end{cases}$$

which implies (7.16b). □

The expression (7.16b) estimates how much the Lyapunov function has decreased before we can no longer guarantee that it will not increase. Next, we turn our attention to bound how much the Lyapunov may increase after the critical time if the robot does not get updated gradient information from the human. To find such a bound, we make the following additional assumption.

Assumption 7.3.3. ***(Strong Convexity of the Composition Function).*** The composition function $c \circ f$ is strongly convex with parameter μ as a consequence of

(i) each objective function $f_i \in \{f_i\}_{i \in [m]}$ being strongly convex; and

(ii) the cost function c being strictly convex and increasing with respect to each component. •

Under Assumption 7.3.3, the function $V_k(x) = w(x_k)f(x)$ is strongly convex because every component of w is always positive. Therefore, there exist a strongly convex parameter $\mu_k > 0$ such that $\mu_k I \preceq \nabla^2 V_k(x)$. Define also

$$\hat{\mu} = \min_{x \in \mathcal{V}} \|w(x)\| \cdot \min_{x \in \mathcal{V}, i \in [m]} \|\nabla^2 f_i(x)\|,$$

and note that, by definition, $\hat{\mu} \leq \mu_k$ for all k. Our next result characterizes how fast the Lyapunov function increases after the critical time, and how long it will take for the function to exceed the amount it previously decreased.

Lemma 7.3.4. ***(Lyapunov Function Bound After Critical Time).*** *Consider the event-triggered human-robot system* (7.1) *with Assumption 7.3.3. If t_{k+1} is such that $x(t) \in S_k$, for all $t \in [t_k, t_{k+1})$, and $t_{k+1} \geq t_{\mathrm{cr},k}$, then*

$$V(x(t)) \leq V(x(t_{\mathrm{cr},k})) + V(x(t_k))\beta(t) \tag{7.17a}$$

for $t \in [t_{\mathrm{cr},k}, t_{k+1}]$, where β is the strictly increasing function

$$\beta(t) = \frac{2\hat{M}^2}{\hat{\mu}} \int_{t_{\mathrm{cr},k}}^{t} (\xi(s - t_{\mathrm{cr},k} + \tau_1^{\mathcal{V}}) - 1) e^{-2\hat{\mu}(s - t_k)} ds. \tag{7.17b}$$

with $\xi(t) = \frac{L_c \hat{J}}{\hat{M}}(e^{\hat{M}t} - 1)$.

Proof. We begin by finding a bound on the state deviation after the critical time. The assumption on t_{k+1} is the same as that of Lemma 7.2.2, so we can deduce (7.9). Because $\hat{M} \geq M_k$, we can find

$$\frac{d}{dt} \frac{\|\Delta x_k\|}{\|\dot{x}\|} \leq 1 + \hat{M} \frac{\|\Delta x_k\|}{\|\dot{x}\|}$$

for $t \in [t_k, t_{k+1})$. From (7.16a), we note that $\frac{\|\Delta x_k\|}{\|\dot{x}\|} \leq \frac{1}{L_c \hat{J}}$ at time $t_{\mathrm{cr},k}$. Using the Comparison Lemma with dynamics $\dot{\phi} = 1 + \hat{M}\phi$ and initial condition $\phi(t_{\mathrm{cr},k}) = \frac{1}{L_c \hat{J}}$, we have

$$\begin{aligned}\frac{\|\Delta x_k\|}{\|\dot{x}\|} &\leq \frac{1}{\hat{M}} \left(\left(1 + \frac{\hat{M}}{L_c \hat{J}}\right) e^{\hat{M}(t - t_{\mathrm{cr},k})} - 1 \right) \\ &\leq \frac{1}{\hat{M}} (e^{\hat{M}(t - t_{\mathrm{cr},k} + \tau_1^{\mathcal{V}})} - 1)\end{aligned}$$

for $t \in [t_{\mathrm{cr},k}, t_{k+1})$. We use this bound in the time derivative of the Lyapunov function along the trajectory as follows,

$$\begin{aligned}\frac{d}{dt} V(x(t)) &\leq -\|w(x_k) J_f(x)\|^2 + \|\Delta w_k J_f(x)\| \|w(x_k) J_f(x)\| \\ &\leq -\|w(x_k) J_f(x)\|^2 + L_c \|\Delta x_k\| \|J_f(x)\| \|w(x_k) J_f(x)\| \\ &\leq -\|w(x_k) J_f(x)\|^2 + L_c \hat{J} \frac{\|\Delta x_k\|}{\|\dot{x}\|} \|w(x_k) J_f(x)\|^2 \\ &\leq (-1 + \xi(t - t_{\mathrm{cr},k} + \tau_1^{\mathcal{V}})) \|w(x_k) J_f(x)\|^2. \end{aligned} \tag{7.18}$$

for $t \in [t_{\mathrm{cr},k}, t_{k+1})$. By definition, $\xi(t)$ is strictly increasing and $\xi(\bar{\tau}) = 1$, therefore $\xi(t - t_{\mathrm{cr},k} + \tau_1^{\mathcal{V}}) - 1 > 0$ for $t > t_{\mathrm{cr},k}$. Therefore, we proceed by finding the upper bound to $\|w(x_k) J_f(x(t))\|^2$ using (2.3) as follows

$$\begin{aligned}\|w(x_k) J_f(x(t))\|^2 &\leq 2\hat{M} V_k(x(t)) \overset{(a)}{\leq} 2\hat{M} e^{-2\hat{\mu}(t - t_k)} V_k(x_k) \\ &\leq \frac{\hat{M}}{\hat{\mu}} e^{-2\hat{\mu}(t - t_k)} \|w(x_k) J_f(x_k)\|^2 \\ &\leq \frac{2\hat{M}^2}{\hat{\mu}} e^{-2\hat{\mu}(t - t_k)} V(x_k),\end{aligned}$$

where (a) follows from $\frac{d}{dt}V_k(x(t)) = -\|w(x_k)J_f(x(t))\|^2 \leq -2\hat{\mu}V_k(x(t))$. Substituting in (7.18), we obtain

$$\frac{d}{dt}V(x(t)) \leq (\xi(t - t_{\mathrm{cr},k} + \tau_1^{\mathcal{V}}) - 1)\Big(\frac{2\hat{M}^2}{\hat{\mu}}e^{-2\hat{\mu}(t-t_k)}V(x_k)\Big),$$

and the result follows via the Comparison Lemma. □

The combination of Lemmas 7.3.2 and 7.3.4 bounds the evolution of the Lyapunov function before and after the critical time. With these results, we can use guarantee an overall decrease between two interexecution times despite some increase in the Lyapunov function after the critical time. Using this idea, we propose a novel event-triggered design.

Proposition 7.3.5. ***(Trigger for "Need to Rest" Human using Grace Period).*** *Consider the human-robot system* (7.1) *with Assumption 7.3.3. For $\gamma \in [\gamma_0, 1)$, let the* grace period τ_{gr} *be the solution to*

$$\gamma - \gamma_0 = \frac{2\hat{M}^2}{\hat{\mu}}\int_{\tau_1^{\mathcal{V}}}^{\tau_{\mathrm{gr}}+\tau_1^{\mathcal{V}}} (\xi(s) - 1)e^{-2\hat{\mu}s}ds.$$

For $T_{\mathrm{rest}} \in [\tau_1^{\mathcal{V}}, \tau_{\mathrm{gr}} + \tau_1^{\mathcal{V}}]$, let the updates $\{t_{k+1}\}_{k\in\{0\}\cup\mathbb{N}}$ be determined according to

$$t_{k+1} = \min\left\{t \geq t_{\mathrm{cr},k} + \tau_{\mathrm{gr}} \mid \|\Delta x_k\| \geq \frac{\|\nabla c(f(x_k))J_f(x)\|}{L_c\|J_f(x)\|}\right\}. \tag{7.19}$$

Then, for each $k \in \{0\}\cup\mathbb{N}$, $V(x(t_{k+1})) \leq \gamma V(x(t_k))$, and $t_{k+1} - t_k \geq T_{\mathrm{rest}}$. As a result, the optimizer

is asymptotically stable, with $\mathcal{X}_0$ contained in its region of attraction, and

$$V(x(t_k)) \leq \gamma^k V(x_0). \tag{7.20}$$

Proof. Note that due to the trigger design (7.19), $t_{k+1} \geq t_{\mathrm{cr},k}$, and therefore, both Lemma 7.3.2 and Lemma 7.3.4 hold. We use the following bounds

$$\begin{aligned}
\beta(t_{\mathrm{cr},k} + \tau_{\mathrm{gr}}) &= \frac{2\hat{M}^2}{\hat{\mu}} \int_{t_{\mathrm{cr},k}}^{t_{\mathrm{cr},k}+\tau_{\mathrm{gr}}} (\xi(s - t_{\mathrm{cr},k} + \tau_1^{\mathcal{V}}) - 1) e^{-2\hat{\mu}(s-t_k)} ds \\
&= \frac{2\hat{M}^2}{\hat{\mu}} \int_{\tau_1^{\mathcal{V}}}^{\tau_{\mathrm{gr}}+\tau_1^{\mathcal{V}}} (\xi(s) - 1) e^{-2\hat{\mu}(s-t_k+t_{\mathrm{cr},k}-\tau_1^{\mathcal{V}})} ds \\
&\leq \frac{2\hat{M}^2}{\hat{\mu}} \int_{\tau_1^{\mathcal{V}}}^{\tau_{\mathrm{gr}}+\tau_1^{\mathcal{V}}} (\xi(s) - 1) e^{-2\hat{\mu}s} ds \\
&= \gamma - \gamma_0
\end{aligned}$$

where the inequality holds because $\xi(s) - 1 \geq 0$ for $s \geq \tau_1^{\mathcal{V}}$, and $t_k - t_{\mathrm{cr},k} + \bar{\tau} \leq 0$ from Lemma 7.3.2. Substituting the inequality above and (7.16b) to evaluate (7.17a), we obtain

$$V(x(t_{\mathrm{cr},k} + \tau_{\mathrm{gr}} - \tau_1^{\mathcal{V}})) \leq \gamma_0 V(x_k) + (\gamma - \gamma_0) V(x_k) = \gamma V(x_k).$$

Now, for $t \in [t_{\mathrm{cr},k} + \tau_{\mathrm{gr}} - \tau_1^{\mathcal{V}}, t_{k+1})$, we can find that $\|\Delta x_k\| < \sigma\left(\frac{\|\nabla c(f(x_k))J_f(x)\|}{L_c \|J_f(x)\|}\right)$. As such, (7.5) holds and Lyapunov function decreases during the duration. In other words, $V(x(t_{k+1})) \leq \gamma V(x_k)$, and (7.20) follows. Finally, because $t_{\mathrm{cr},k} - \tau_1^{\mathcal{V}} \geq t_k$, we note that $t_{k+1} \geq t_k + \tau_{\mathrm{gr}} + \tau_1^{\mathcal{V}}$ by design, so $t_{k+1} - t_k \geq T_{\mathrm{rest}}$ as claimed. Since the design is Zeno-free, the optimizer is asymptotically stable. □

Note that the grace period τ_{gr} can be determined offline given the various problem parameters and the design parameter γ. Proposition 7.3.5 offers a strategy for accommodating longer resting times than the ones obtained in Proposition 7.3.1. Let us recapture here the ideas behind the trigger design (7.19) that allows to accomplish this. After each human update, we use the trigger (7.16a) to determine $t_{\text{cr},k}$, where we know the decrease in V given by (7.16b). We then let the system proceed without any human update for τ_{gr}. In this period, the definition of τ_{gr} in Proposition 7.3.5 guarantees that V can increase but cannot exceed $\gamma V(x_k)$, which is a direct result from Lemma 7.3.4. Finally, we let the system continue with trigger (7.19), which will prescribe an update once V stops decreasing (this might be immediate). As a consequence, our design ensures that the Lyapunov function decreases between two consecutive execution times.

The design parameter γ directly corresponds to the convergence rate guarantee, cf. (7.20). Note that the convergence rate is given with respect to the number of iterations rather than time, which are not equivalent when the interexecution time is not fixed. In any case, much like how the accommodation of longer resting times increases the value σ in Proposition 7.3.1, here it requires a larger value of γ, which slows down the convergence rate.

7.3.2 "Need to Think" Human

In this section, we deal with the case when the human does not respond instantaneously to queries from the robot and instead, once asked, takes some time "to think" and provide information. Formally, for each $k \in \{0\} \cup \mathbb{N}$, when the robot asks the human at time t_{k+1} for the evaluation of the gradient ∇c at $f(x_{k+1})$, the human takes some time $D_{k+1} \geq 0$ to relay the information $\nabla c \circ f(x_{k+1})$. This means that, up until $t_{k+1} + D_{k+1}$, the robot still uses the "old" information $\nabla c \circ f(x_k)$ provided

in the previous communication with the human. The dynamics is then given by

$$\dot{x} = -(\nabla c(f(x_k))J_f(x))^\top, \tag{7.21}$$

for $t \in [t_k + D_k, t_{k+1} + D_{k+1}]$. Human thinking time spans are not necessarily equal across different time instants, but we assume them to be uniformly upper bounded by a known constant $T_{\text{thk}} > 0$, representing the maximum time it takes the human to relay her gradient information. Regarding the initialization of the dynamics, we assume that the optimization starts when the human gives his initial gradient information, and therefore, $D_0 = 0$.

Given the model above, there are two new complications that arise in designing the event-triggered law. First, it is clear that the robot should not wait until it absolutely needs the new gradient information available to request it from the human, as it did in the ideal human case considered in Section 7.2. In other words, if we were to define the time at which the trajectory satisfies the condition for (7.3) as

$$t_{\text{nec},k} = \min\left\{ t \geq t_k \mid \|\Delta x_k\| = \frac{\sigma\|\nabla c(f(x_k))J_f(x(t))\|}{L_c\|J_f(x(t))\|} \right\},$$

then we would like $t_{k+1} + D_{k+1}$ to occur before $t_{\text{nec},k}$ to ensure condition (7.4), like we did in the ideal human case. However, this is not simple as subtracting T_{thk} from $t_{\text{nec},k}$ because we do not know exactly what $t_{\text{nec},k}$ is since it is determined by an event. The robot should anticipate the human delay in responding and ask in advance, ideally D_{k+1} before the need for updated information arises. Another complication in designing a trigger is that the trigger may occur too often. We assume that when queried, the human operator is busy during the time interval $[t_{k+1}, t_{k+1} + D_{k+1}]$, and therefore

cannot accept another query during this time.

In summary, we want our new design to prescribe t_{k+1} satisfying the following:

- t_{k+1} occurs before $t_{\text{nec},k} - D_{k+1}$;
- t_{k+2} happens after $t_{k+1} + D_{k+1}$ when prescribed iteratively.

To achieve these objectives, we use a similar trigger design as in the ideal human case with a new design parameter σ'. Our strategy is based on tuning this parameter so that t_{k+1} neither occurs too late nor too early. This is done by estimating how long after t_{k+1} it takes for $t_{\text{nec},k}$ to occur, and how long it takes for t_{k+2} to occur after t_{k+1}. The following result makes this statement precise.

Proposition 7.3.6. ***(Trigger for "Need to Think" Human).*** *Consider the event-triggered human-robot system* (7.21)*. With $\sigma \in (0,1)$, let D^*_{thk} be the unique solution to*

$$\left(\frac{1+\sigma}{1-\sigma}\right)^2 = \frac{e^{\hat{M}(\tau_\sigma^{\mathcal{V}} - D^*_{\text{thk}})} - 1}{e^{\hat{M} D^*_{\text{thk}}} - 1},$$

and assume $D^*_{\text{thk}} > T_{\text{thk}}$. Let $\sigma' \in (0,1)$ be such that

$$\frac{L_c \hat{J}}{\hat{M}} \left(\frac{1+\sigma}{1-\sigma}\right)^2 (e^{\hat{M} T_{\text{thk}}} - 1) < \sigma' \leq \frac{L_c \hat{J}}{\hat{M}} (e^{\hat{M}(\tau_\sigma^{\mathcal{V}} - T_{\text{thk}})} - 1). \tag{7.22}$$

For $k \in \{0\} \cup \mathbb{N}$, let t_{k+1} be determined by

$$t_{k+1} = \min \left\{ t \geq t_k \mid \|\Delta x_k\| = \frac{\sigma' \|\nabla c(f(x_k)) J_f(x)\|}{L_c \hat{J}} \right\}. \tag{7.23}$$

Then, for each $k \in \{0\} \cup \mathbb{N}$, we have $\|\Delta x_k\| < \frac{\sigma \|\nabla c(f(x_k)) J_f(x)\|}{L_c \|J_f(x)\|}$ for $t \in [t_k, t_{k+1} + D_{k+1})$, $t_{k+2} > t_{k+1} + D_{k+1}$, and as a consequence, the performance guarantee (7.4) *on the Lyapunov function*

holds for all $t \in [t_k + D_k, t_{k+1} + D_{k+1})$.

Proof. We start by guaranteeing the existence of σ'. For this, we simply show that the upper bound in (7.22) is greater than or equal to the lower bound, or equivalently,

$$\left(\frac{1+\sigma}{1-\sigma}\right)^2 < \frac{e^{\hat{M}(\tau_\sigma^{\mathrm{i}} - T_{\mathrm{thk}})} - 1}{e^{\hat{M}T_{\mathrm{thk}}} - 1}$$

The right hand side is strictly decreasing with respect to T_{thk}. Given the definition of D^*_{thk}, we deduce that all $T_{\mathrm{thk}} < D^*_{\mathrm{thk}}$ satisfy the inequality. Note also that $\sigma' \leq \sigma$.

Next, with a slight abuse of notation, we use $\dot{x}^{[k]} = (\nabla c(f(x_k))J_f(x))^\top$. We resort to Table 7.1 to help specify desired values of $\|\Delta x\|$ in effect at different time intervals.

Table 7.1: Desired state deviation at different time intervals.

Interval	$(t_k + D_k, t_{k+1})$	t_{k+1}	$(t_{k+1}, t_{k+1} + D_{k+1})$
$\|\Delta x_k\|$	$< \frac{\sigma'\|\dot{x}^{[k]}\|}{L_c\hat{J}}$	$\frac{\sigma'\|\dot{x}^{[k]}\|}{L_c\hat{J}}$	$< \frac{\sigma\|\dot{x}^{[k]}\|}{L_c\|J_f(x)\|}$
$\|\Delta x_{k+1}\|$	Undefined	0	$< \frac{\sigma'\|\dot{x}^{[k+1]}\|}{L_c\hat{J}}$

The first part of the proof focuses on the evolution of $\|\Delta x_k\|$. As shown in the last column of Table 7.1, the trigger (7.23) requesting the gradient at time t_{k+1} should not violate $\|\Delta x_k\| < \sigma\left(\frac{\|\dot{x}^{[k]}\|}{L_c\|J_f(x)\|}\right)$ up until the gradient implementation at $t_{k+1} + D_{k+1}$. To do so, we would like D_{k+1} to be shorter than the time it takes for $\frac{\|\Delta x_k\|}{\|\dot{x}^{[k]}\|}$ to evolve, from $\frac{\sigma'}{L_c\hat{J}}$ to $\frac{\sigma}{L_c\hat{J}}$ (Notice that $\frac{\sigma}{L_c\hat{J}} < \frac{\sigma}{L_c\|J_f(x)\|}$, $\forall x \in \mathcal{X}_0$). This leads to applying the Comparison Lemma with the function ϕ satisfying $\dot{\phi} = 1 + \hat{M}\phi$ with $\phi(t_{k+1}) = \frac{\sigma'}{L_c\hat{J}}$, and asking for

$$D_{k+1} \leq \frac{1}{\hat{M}}\ln(1 + \hat{M}\frac{\sigma}{L_c\hat{J}}) - \frac{1}{\hat{M}}\ln(1 + \hat{M}\frac{\sigma'}{L_c\hat{J}}).$$

To ensure this condition, we can select σ' so that the right hand side is an upper bound on T_{thk}, which leads to the upper bound on σ' in (7.22).

The second part of the proof examines the possibility of the state error $\|\Delta x_{k+1}\|$ to begin with a larger value than the trigger value at the time $t_{k+1} + D_{k+1}$ of implementation of the new gradient. This is possible because Δx_{k+1} evolves with $\dot{x}^{[k]}$ until $t_{k+1} + D_{k+1}$. For this, let $T_{\text{allow}} = t_{k+2} - t_{k+1}$ be the time it takes $\|\Delta x_{k+1}\|$ to evolve from 0 to $\frac{\sigma'\|\dot{x}^{[k+1]}\|}{L_c \hat{J}}$ with the dynamics $\dot{x} = \dot{x}^{[k]}$. We will show that enforcing the lower bound on σ' in (7.22) ensures $D_{k+1} \leq T_{\text{allow}}$. We reason by contradiction. Assume $D_{k+1} > T_{\text{allow}}$ and let us examine the dynamics of $\frac{\|\Delta x_{k+1}\|}{\|\dot{x}^{[k]}\|}$. Following a similar derivation as in the proof of Lemma 7.2.2, we arrive at

$$\frac{d}{dt}\frac{\|\Delta x_{k+1}\|}{\|\dot{x}^{[k]}\|} \leq \frac{\|\dot{x}^{[k+1]}\|}{\|\dot{x}^{[k]}\|} + \hat{M}\frac{\|\Delta x_{k+1}\|}{\|\dot{x}^{[k]}\|}. \tag{7.24}$$

We next proceed to bound $\frac{\|\dot{x}^{[k+1]}\|}{\|\dot{x}^{[k]}\|}$. Note that, from (7.6), we have $\|w(x)J_f(x)\| \leq (1+\sigma)\|\dot{x}^{[k]}\|$ for $t \in [t_k, t_{k+1} + D_{k+1}]$. Additionally, from $w(x_k)J_f(x) = (w(x) - \Delta w_k)J_f(x)$,

$$\begin{aligned}
\|w(x_{k+1})J_f(x)\| &\leq \|w(x)J_f(x)\| + \|\Delta w_{k+1}\|\|J_f(x)\| \\
&\leq \|w(x)J_f(x)\| + L_c\|\Delta x_{k+1}\|\|J_f(x)\| \\
&\leq \|w(x)J_f(x)\| + \sigma'\|w(x_{k+1})J_f(x)\| \\
&\leq \|w(x)J_f(x)\| + \sigma\|w(x_{k+1})J_f(x)\|,
\end{aligned}$$

during $t \in [t_{k+1}, t_{k+1} + T_{\text{allow}}]$. Using this in conjunction with (7.6) and the fact that $T_{\text{allow}} < D_{k+1}$, we have

$$(1-\sigma)\|\dot{x}^{[k+1]}\| \leq \|w(x)J_f(x)\| \leq (1+\sigma)\|\dot{x}^{[k]}\|,$$

valid for $t \in [t_{k+1}, t_{k+1} + T_{\text{allow}}]$, and hence $\frac{\|\dot{x}^{[k+1]}\|}{\|\dot{x}^{[k]}\|} \leq \frac{1+\sigma}{1-\sigma}$. Substituting this ratio in (7.24), we get

$$\frac{d}{dt}\frac{\|\Delta x_{k+1}\|}{\|\dot{x}^{[k]}\|} \leq \frac{1+\sigma}{1-\sigma} + \hat{M}\frac{\|\Delta x_{k+1}\|}{\|\dot{x}^{[k]}\|}.$$

Now, we solve for $\dot{\psi} = \frac{1+\sigma}{1-\sigma} + \hat{M}\psi$ with initial condition $\psi(t_{k+1}) = 0$, and use the Comparison Lemma to ensure

$$\psi(t_{k+1} + T_{\text{allow}}) = \frac{(1+\sigma)}{\hat{M}(1-\sigma)}(e^{\hat{M}T_{\text{allow}}} - 1) \geq \frac{\|\Delta x_{k+1}\|}{\|\dot{x}^{[k]}\|}.$$

Recall this inequality is true for all $t \in [t_{k+1}, t_{k+1} + T_{\text{allow}}]$ where we can use once again the relationship $\|\dot{x}^{[k+1]}\| \geq \frac{1-\sigma}{1+\sigma}\|\dot{x}^{[k]}\|$. Therefore, at $t = t_{k+1} + T_{\text{allow}}$,

$$\frac{\sigma'}{L_c\hat{J}} = \frac{\|\Delta x_{k+1}\|}{\|\dot{x}^{[k+1]}\|} \leq \frac{1}{\hat{M}}\left(\frac{1+\sigma}{1-\sigma}\right)^2(e^{\hat{M}T_{\text{allow}}} - 1) \tag{7.25}$$

Using now the lower bound in (7.22), we deduce $T_{\text{thk}} < T_{\text{allow}}$, which is a contradiction because $T_{\text{thk}} > D_{k+1}$. Therefore, $D_{k+1} \leq T_{\text{allow}}$. Since $t_{k+2} = t_{k+1} + T_{\text{allow}}$, we have $t_{k+2} > t_{k+1} + D_{k+1}$. Finally, since at all times $t \in [t_k + D_k, t_{k+1} + D_{k+1})$, we have bounded $\|\Delta x_k\| \leq \sigma\left(\frac{\|\nabla(c(f(x_k)))J_f(x)\|}{L_c\|J_f(x)\|}\right)$, the Lyapunov function rate (7.4) is ensured. □

Proposition 7.3.6 requires the thinking time T_{thk} to be smaller than D^*_{thk}. Consistent with the treatment of delays in the event-triggered control literature [DBH17, LWL12, HDI06, WRGL15], it does not come as a surprise that the thinking time must be sufficiently small for a trigger design to exist; otherwise, the system will receive no human updates for too long and start behaving unsatisfactorily. D^*_{thk} can then be interpreted as the maximum allowable thinking time for the human.

Although we interpret the delays as caused by the human's thinking time, other sources of delay could be equally accommodated by Proposition 7.3.6.

Similarly to the ideal human case of Section 7.2, to ensure convergence, we need to show that the trigger (7.23) will not exhibit Zeno behavior. The following result provides a uniform lower bound on the interexecution time.

Proposition 7.3.7. ***(Interexecution Time with Update Delay).*** *For the event-triggered human-robot system* (7.21) *with updates determined according to* (7.23)*, and under the same hypotheses as Proposition 7.3.6, the interexecution time is lower bounded as* $t_{k+1} - t_k \geq \tau^{\text{thk}}_{\sigma'}$ *where*

$$\tau^{\text{thk}}_{\sigma'} := \frac{1}{\hat{M}} \ln \left(\frac{1 + \hat{M} \frac{\sigma'}{L_c J_{\max}}}{1 + \left(\frac{1+\sigma}{1-\sigma} \right)^2 (e^{\hat{M} T_{\text{thk}}} - 1)} \right) + T_{\text{thk}}. \tag{7.26}$$

Proof. By construction, during $[t_k + D_k, t_{k+1}]$, the dynamics is given by $\dot{x}^{[k]}$. Similarly to how we obtained inequality (7.25) in the proof of Proposition 7.3.6, we have

$$\frac{\|\Delta x_k\|}{\|\dot{x}^{[k]}\|} \leq \frac{1}{\hat{M}} \left(\frac{1+\sigma}{1-\sigma} \right)^2 (e^{\hat{M} D_k} - 1),$$

at time $t_k + D_k$. Setting $\phi(t_k + D_k)$ equal to the right hand side of the above inequality as the initial condition, we solve the dynamics $\dot{\phi} = 1 + \hat{M}\phi$,

$$\frac{1}{\hat{M}} \ln \left(\frac{1 + \hat{M}\phi(t)}{1 + \hat{M}\phi(t_k + D_k)} \right) = t - (t_k + D_k),$$

for $t \geq t_k + D_k$. Using the Comparison Lemma, we know $\frac{\|\Delta x_k\|}{\|\dot{x}^{[k]}\|} \leq \phi(t)$, so it takes longer time for $\frac{\|\Delta x_k\|}{\|\dot{x}^{[k]}\|}$ to evolve to $\frac{\sigma'}{L_c \hat{J}}$ (precisely $t_{k+1} - t_k - D_k$) than it takes $\phi(t_k + D_k)$ to increase to $\phi(t) = \frac{\sigma'}{L_c \hat{J}}$.

As such, we find that

$$t_{k+1} - t_k \geq \frac{1}{\hat{M}} \ln\left(\frac{1 + \hat{M} \frac{\sigma'}{L_c J_{\max}}}{1 + \left(\frac{1+\sigma}{1-\sigma}\right)^2 (e^{\hat{M} D_k} - 1)} \right) + D_k.$$

The result now follows by observing that the right hand side is decreasing in D_k. □

The combination of Propositions 7.3.6 and 7.3.7 ensures that the event-triggered human-robot system (7.21) with updates determined according to (7.23) enjoys the same convergence guarantee as stated in Corollary 7.2.4.

Corollary 7.3.8. ***(Asymptotic Stability – "Need to Think" Human Design).*** *For the event-triggered human-robot system* (7.21) *with updates determined according to* (7.23)*, the optimizer x^* is asymptotically stable, with $\mathcal{X}_0 = \{x_0 \in \mathbb{R}^n \mid \mathcal{S}_0 \subseteq \mathcal{V}\}$ contained in its region of attraction. Moreover, if $c \circ f$ is strongly convex with constant $\mu > 0$ on $\mathcal{V}$, then given an initial condition $x_0 \in \mathcal{X}_0$,* (7.12) *holds for all $k \in \{0\} \cup \mathbb{N}$. As a consequence, the optimizer is exponentially stable with the bound* (7.13) *for $\sigma \in (0, 1)$.* □

7.3.3 "Need to Think Then Rest" Human

Here, we combine the "need to rest" and "need to think" models into a single one: not only does the human take some time in responding to a robot's query, but she also has to rest before she can reply to the robot again. Formally, this means that the robot follows the dynamics (7.21) with the additional constraint that $t_{k+1} \geq t_k + D_k + T_{\text{rest}}$ for all $k \in \{0\} \cup \mathbb{N}$. Our next result addresses this problem.

Proposition 7.3.9. ***(Trigger for "Need to Think Then Rest" Human).*** *Consider the event-triggered*

human-robot system (7.21). *Given* $\sigma \in (0,1)$ *and* T_{rest}, *let* $D^*_{\text{tnk}-\text{rst}}$ *be the unique solution to*

$$\left(\frac{1+\sigma}{1-\sigma}\right)^2 = \frac{e^{\hat{M}(\tau_\sigma^{\mathcal{V}} - D^*_{\text{tnk}-\text{rst}} - T_{\text{rest}})} - 1}{e^{\hat{M} D^*_{\text{tnk}-\text{rst}}} - 1}, \tag{7.27}$$

and assume $D^*_{\text{tnk}-\text{rst}} > T_{\text{thk}}$. *For* $k \in \{0\} \cup \mathbb{N}$, *let* t_{k+1} *be determined according to* (7.23), *where* $\sigma' \in (0,1)$ *is such that*

$$\frac{L_c \hat{J}}{\hat{M}}\left(\left(\frac{1+\sigma}{1-\sigma}\right)^2 (e^{\hat{M}T_{\text{thk}}} - 1)e^{\hat{M}T_{\text{rest}}} + (e^{\hat{M}T_{\text{rest}}} - 1)\right) < \sigma' \leq \frac{L_c \hat{J}}{\hat{M}}(e^{\hat{M}(\tau_\sigma^{\mathcal{V}} - T_{\text{thk}})} - 1).$$

Then, for each $k \in \{0\} \cup \mathbb{N}$, *we have* $\|\Delta x_k\| < \frac{\sigma \|\nabla c(f(x_k)) J_f(x)\|}{L_c \|J_f(x)\|}$ *for* $t \in [t_k, t_{k+1} + D_{k+1})$, $t_{k+2} > t_{k+1} + D_{k+1} + T_{\text{rest}}$, *and as a consequence, the bound* (7.4) *on the evolution of the Lyapunov function holds for all time,* $t \in [t_k + D_k, t_{k+1} + D_{k+1})$.

Proof. First, note that the newly introduced rest time constraint has no effect on how the upper bound is derived in (7.22), so it remains the same here. On the other hand, the lower bound to σ' is affected by the rest time constraint. Specifically, we must now guarantee that the value of $\|\Delta x_{k+1}\|$ must not exceed the trigger condition $\frac{\sigma' \|\dot{x}^{[k+1]}\|}{L_c J_{\max}}$, but at the time $t_{k+1} + D_{k+1} + T_{\text{rest}}$ (instead of the earlier $t_{k+1} + D_{k+1}$). We break the time of interest into two intervals, $[t_{k+1}, t_{k+1} + D_{k+1}]$ and $[t_{k+1} + D_{k+1}, t_{k+1} + D_{k+1} + T_{\text{rest}}]$ because in these two intervals, the dynamics are different due to the human's update.

First, we focus on the latter of the two time intervals. From (7.8),

$$\frac{d}{dt}\frac{\|\Delta x_{k+1}\|}{\dot{x}^{[k+1]}} \leq 1 + \hat{M}\frac{\|\Delta x_{k+1}\|}{\dot{x}^{[k+1]}}.$$

As such, we know that if

$$\frac{\|\Delta x_{k+1}\|}{\dot{x}^{[k+1]}} \leq \frac{1}{\hat{M}}\left((1+\hat{M}\frac{\sigma'}{L_c J_{\max}})e^{-\hat{M}T_{\text{rest}}}-1\right),$$

at time $t_{k+1}+D_{k+1}$, then $\frac{\|\Delta x_{k+1}\|}{\dot{x}^{[k+1]}} \leq \frac{\sigma'}{L_c J_{\max}}$ at time $t_{k+1}+D_{k+1}+T_{\text{rest}}$ by using the Comparison Lemma. Next, we deal with the interval $[t_{k+1}, t_{k+1}+D_{k+1}]$ to show that the lower bound given in the statement ensures the above inequality at time $t_{k+1}+D_{k+1}$. This can be done by following the same contradiction proof procedure as presented for Proposition 7.3.6. □

Proposition 7.3.9 gives a method to deal with both human resting and thinking time. Once again, these constraints must be sufficiently small. We can interpret $D^*_{\text{thk}-\text{rst}}$ as the maximum allowable thinking time for the human, given the amount of time he needs to rest. As Zeno behavior is absent due to the interexecution time being lower bounded by the resting time, an analogous statement to Corollary 7.3.8 follows. The given model is the richest in term of dealing with constraints on human performance, and we can recover earlier models by setting resting or thinking time to zero.

Remark 7.3.10. ***(Units of Time).*** The two types of constraints on human performance considered here both have parameters dealing with time. The resting time T_{rest} and the thinking time D most likely will be quantified with units of time that are meaningful in the real world (e.g., seconds and hours). On the other hand, the gradient descent has its own unit of time that is encoded in the dynamics. To use our results, it is important to reconcile the difference in units. One way to do this is by noting that the gradient descent is calculated by the robot. In practice, the robot will probably implement the continuous gradient dynamics through a discretization with a constant

stepsize. Given how long the robot takes to compute each step in the gradient descent, we have a convenient unit conversion between the two time units. We note here that the conversion depends on the robot's computing power and the selected stepsize. •

7.4 Simulations

We consider a human-robot interaction scenario where a human aids the robot in determining a safe trajectory through an environment populated with threats of different levels. The robot is tasked to travel from the origin at $(0, 0)$ to $(1, 0)$ on the xy-plane. The robot has scanned a few potential threats in the area with positions among

$$(x_{\text{obs}}, y_{\text{obs}}) = \{(0.8, 0.1),\ (0.3, -0.2),\ (0.2, 0.04), (0.68, 0.3),\ (0.5, 0.12)\}.$$

Ideally, the robot would like to stay away from these locations while, at the same time, would like to traverse the shortest path possible to its goal. To describe its reference trajectory, the robot uses a sum of sinusoidal functions as,

$$y = \sum_{i=1}^{10} a_i \sin(i\pi x),\ \ x \in [0, 1],$$

where $a \in \mathbb{R}^{10}$ are the amplitudes to be optimized. The objective functions for avoiding obstacles and for measuring path length are given by

$$f_{\text{obs},j}(a) = -\max_{x\in[0,1]} \left\{ (y_{\text{obs},j} - y)^2 + (x_{\text{obs},j} - x)^2 \right\},$$
$$f_{\text{len}}(a) = \int_0^1 \sqrt{1 - (\sum_{i=1}^{10} i\pi a_i \cos(i\pi x))^2}\, dx,$$

for $j \in [5]$ and $a \in \mathbb{R}^{10}$. In order to find the amplitudes that best describe the most desired reference trajectory, a human works with the robot in the multiobjective optimization problem with objective functions $f_{\text{obs},1}, \ldots, f_{\text{obs},5}, f_{\text{len}}$ by evaluating risks of the likely threats and providing the robot with gradient information. In practice, the human preference function is not known, and the gradient information can only be estimated, perhaps through asking the human to rate the importance of each objective functions at a given point. For the purpose of the simulation, we use the following function to represent the human preferences,

$$c(f) = \frac{f_{\text{len}}^2}{10} + \sum_{j=1}^{5} \frac{q_j}{f_{\text{obs},j}^2},$$

where $q = [0.2, 0.5, 0.03, 0.1, 0.3]$. The weights captured in q represent how the human assesses the threats. The third potential threat, for example, is an order of magnitude lower than the others. This can represent, for instance, how the human knows that the third object is a friendly entity and does not pose much risk besides a potential crash. We assume the human provides the exact value of the gradient information and focus on how we can apply the results in this paper to accommodate the constraints in human performance, as discussed in Section 7.3.

We run our simulations in MATLAB on a desktop with a 3.5GHz Intel Core i5-6600K

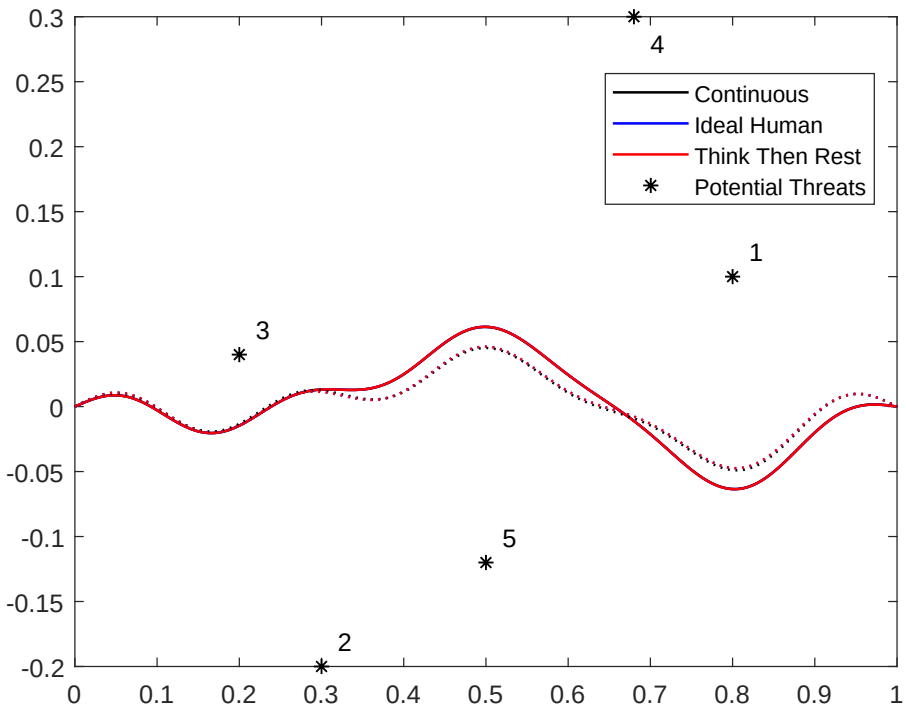


Figure 7.2: Optimized trajectories. The diagram shows the trajectories of the optimized amplitudes after 10000 robot iterations with the initial condition of a straight path, $a = 0$. Shown in the dotted lines are the trajectories at 5000 robot iterations for each respective model. The proposed triggers for updating human gradient information shows convergence towards the desired (continuous case) trajectory in all the cases.

quad-core CPU and 16GB of RAM. For comparison, we generate a reference optimized trajectory having the robot use gradient information at all times. We refer to this as the "continuous" case. To simulate the continuous dynamics, we use an Euler discretization with a stepsize of 1×10^{-5}. Note that the timescales of the robot, over which the dynamics runs, and of the human are not necessarily the same, cf. Remark 7.3.10. In fact, in our platform, each iteration of this discretization takes the robot roughly one second to compute. Therefore, continuous queries by the robot would mean that the human needs to respond every second. From an operational point of view, this amount of time can be too little for the human to work with. Instead, the results of this paper allow the robot to efficiently query the human in an opportunistic fashion to continue its operation and also allow the

human to gain more time to work between consecutive queries.

Even though the resulting optimization problem is not convex, we employ our event-triggered law (7.3) with $\sigma = 0.5$ to find a local optimizer via human-robot interactive gradient descent (cf. Remark 7.2.5). Under the ideal human model of Section 7.2, the trajectory from the resulting optimized amplitudes after 10000 iterations is plotted in Figure 7.2. Using (7.10), the lower bound to the interexecution time is 5.2×10^{-4}, i.e., 52 robot iterations. As a result, the human does not need to respond at every iteration to ensure convergence to the desired trajectory. In addition, Figure 7.3 shows the number of iterations elapsed before the human responds, which is lower bounded by the aforementioned value. Note that the optimal trajectory ends up closer to the obstacle on the left, compared to others. This is expected because the object corresponds to the threat location with weight q_3, which has lower potential risk than the others.

Next, we consider a scenario where the human may take time to provide gradient information and may need some time to rest between consecutive queries, as described under the "need to think then rest" model of Section 7.3.3. With the notation of that section, we select $T_{\text{thk}} = 5 \times 10^{-5}$, which corresponds to 5 robot iterations – i.e., the robot has to wait for up to 5 iterations in the execution of its gradient dynamics before receiving a response from the human to its query. In order to determine the allowable resting times in this scenario, we use (7.27) from Proposition 7.3.9 to plot in Figure 7.4 the design space of pairs $(\sigma, T_{\text{rest}})$ for which the above thinking time D is feasible. To obtain the same guaranteed convergence rate as in the ideal human case, we select $\sigma = 0.5$ and then, based on Figure 7.4, we pick $T_{\text{rest}} = 1 \times 10^{-4}$ to be in the interior of the feasible option. Note that this selection corresponds to 10 iterations of the robot. In practice, the algorithm might still converge with much longer resting times because of the various bounds involved in obtaining our guarantee. In fact, in our simulations, cf. Figure 7.3, we observe that the human actually has the minimum

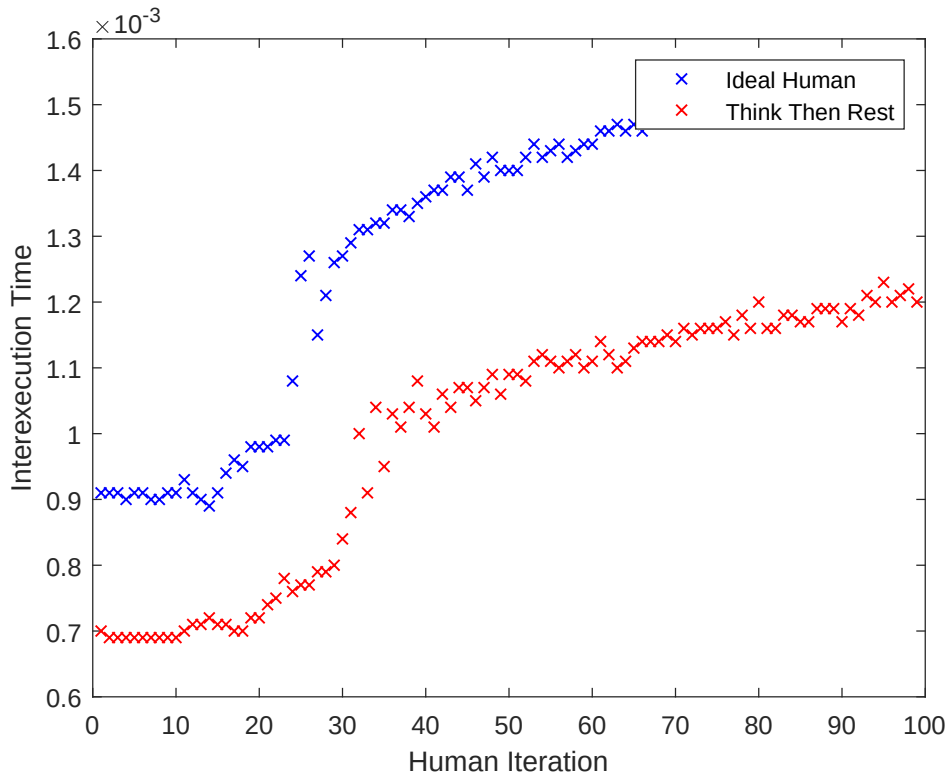


Figure 7.3: Interexecution times. The diagram shows the time elapsed between each request from the robot to the human to update the gradient value. The interevent times are uniformly lower bounded in both cases, as guaranteed by our analysis. The robot runs its dynamics with stepsize 1×10^{-5}, which means that the human only has to respond after at least 65 iterations.

resting time of 6.5×10^{-4}, which corresponds to 65 iterations. According to Proposition 7.3.9, we select $\sigma' = 0.42$ to satisfy the hypotheses and implement the event-triggered law (7.23). The result shows that the resting and thinking time constraints are respected and, as expected, the interevent times are reduced to accommodate the delay, cf. Figure 7.3. Figure 7.5 shows the evolution of the Lyapunov function, where one can see that the same level of performance as in the ideal human case is attained.

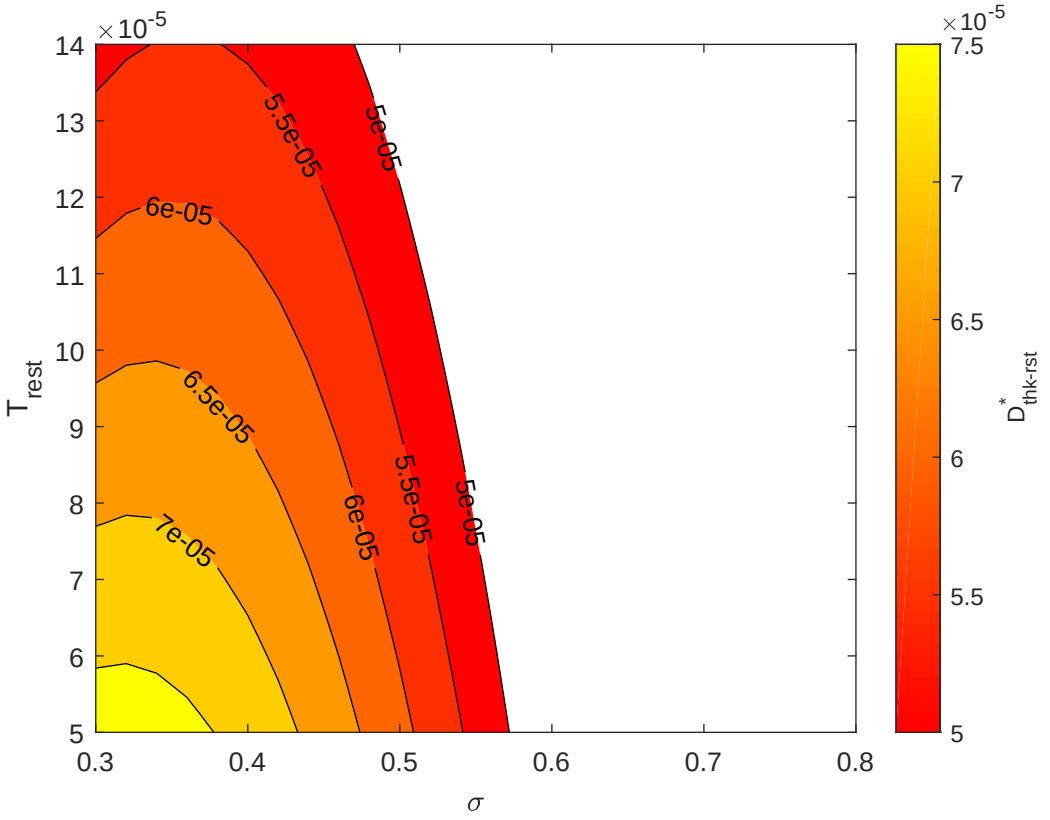


Figure 7.4: Design space for "need to think then rest human." The diagram shows the region where the pairs $(\sigma, T_{\text{rest}})$ that can accommodate the thinking time $T_{\text{thk}} = 5 \times 10^{-5}$. The colored region represents the sublevel sets of maximum allowable thinking time $D^*_{\text{tnk}-\text{rst}}$, starting with $D^*_{\text{tnk}-\text{rst}} \geq 5 \times 10^{-5}$, and the brightness denotes higher values.

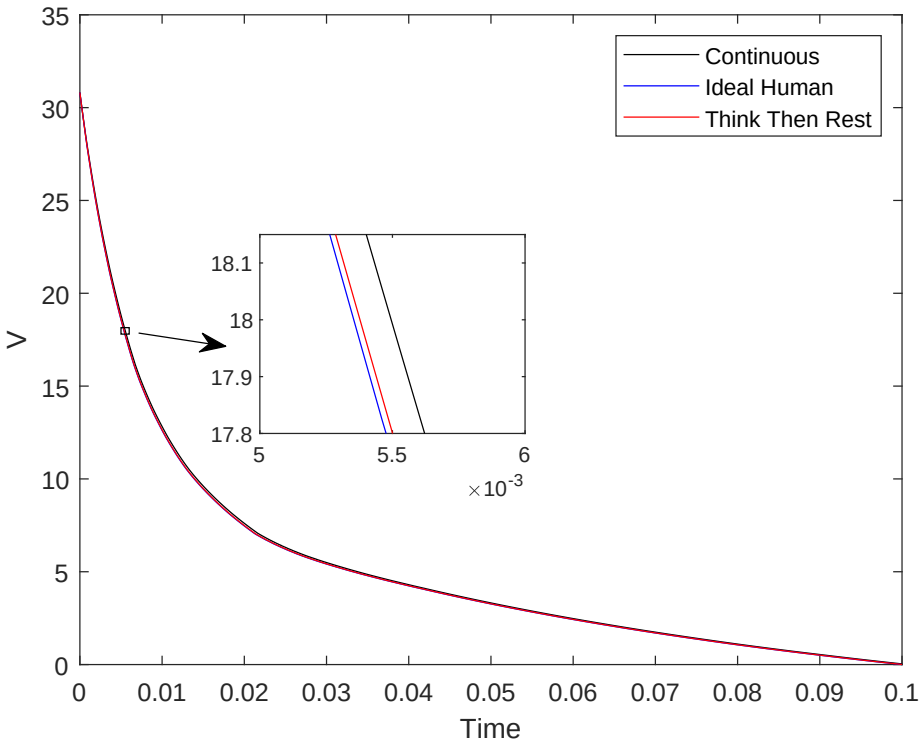


Figure 7.5: Convergence of the cost function. The plot shows the evolution of the Lyapunov function using our event-triggered design for different models. The design parameters σ for the "need to think then rest" model is chosen to match the convergence guarantee of the ideal human case.

Acknowledgements

This chapter, in part, is a reprint of the material [OC21a] as it appears in 'Opportunistic robot control for interactive multiobjective optimization under human performance limitations' by P. Ong and J. Cortés, in Automatica, 2021. The dissertation author was the primary investigator and author of this paper. The work in this chapter was partially supported by NSF Award ECCS-1917177 and CNS-1329619.

Chapter 8

Conclusions

Balancing multiple control objectives in a multi-robot systems is the overarching theme of this dissertation. While special emphasis has been put on stability and safety, we have also discussed concepts like connectivity maintenance and smoothness of the feedback controllers as the control goals for the design for multi-robot systems. We have also looked at the resource-efficient methods of implementation of the controller using event-triggered control. We have developed an event-triggered control framework for stability in distributed systems, safety, and interactive multi-objective optimization.

8.1 Summary

Chapter 3 considers the problem of maintaining network connectivity in multi-robot systems while satisfying nominal requirements that encode desired control objectives. Our solution employs the algebraic connectivity of the interconnection topology as a nonsmooth control barrier function to produce additional constraints for the optimization-based synthesis of the controller

that guarantee it is continuous and maintains network connectivity. The technical approach fully embraces the nonsmooth nature of the algebraic connectivity and other spectral functions of the Laplacian matrix corresponding to the interconnection graph. This has led us to define two different continuous set-valued constraint maps, one that reasons with the merged lower bound of all the eigenvalues' rate of change at once and another, less conservative, that instead reasons over merged lower bounds of an increasing number of eigenvalues' rate of change. We have illustrated the effectiveness of our approach in both simulation and experiment in a resource gathering multi-robot scenario.

While Chapter 3 uncovers issues when CBFs is nonsmooth, Chapter 4 looks at the problems when CBFs are smooth. Particularly, we have pointed out that existing controller design methodologies do not retain smoothness properties and have formulated a feedback controller for smooth safe stabilization. Given a CLF and a CBF, the formula considers the associated admissible control set, and calculates the weighted centroids with normal distribution weights of different sets. By combining the centroids in a smooth way, the given feedback controller retains the smoothness property of the CLF-CBF pair. Also, by manipulating the "standard deviation," the controller can be found continuous at the origin when the small control property holds.

Chapter 5 turns attention towards the implementation of stabilizing controllers. We have developed a novel framework for event-triggered control design that meets a prescribed performance regarding convergence. The proposed approach allows for greater flexibility in prescribing update times by allowing the certificate to gradually deviate from strictly decreasing in proportion to the performance residual. We have shown analytically how, for exponential performance specifications, the resulting trigger design exhibits an improved MIET with respect to the derivative-based approach. We have taken advantage of the flexibility of the proposed approach to design intrisically

Zeno-free triggers for network systems that rely on distributed computation and communication and are applicable for a general class of systems.

Following an event-triggered control framework for stabilizing controllers Chapter 5, we provide another framework for safeguarding controllers in Chapter 6. We have presented a novel approach for achieving the safety of a system with a resource efficient event-triggered control law using Input-to-State Safe Barrier Functions. Similarities and differences between achieving stability and safety in an event-triggered context were highlighted through an example, with a particular focus on how the behavior guaranteed with ISSf-BFs can lead to interevent times that are not lower bounded. This insight is used to propose a trigger law that renders the system input-to-state safe and guarantees a MIET for the system.

Chapter 7 takes a step back and explores how a system can strike balance between many objectives. Motivated the vision of human-robot symbiotic relationship, we have developed event-triggered strategies for human-robot interactive multiobjective optimization. Our design seeks to minimize human workload by having the robot require her involvement in an opportunistic fashion when it is necessary to ensure the asymptotic correctness of the robot dynamics. We have shown how different human performance limitations can be accommodated, such as the human requiring some time between consecutive queries, requiring some time before producing a response, and a combination thereof. For each model, we show that the corresponding event-triggered strategy is provably correct and Zeno-free, with uniformly lower bounded inter-event times.

8.2 Future Work

The contributions of this dissertation set the stage for many research possibilities, and our future work will reflect that. In this section, we outline our short-term goals and the long-term direction for our research focus building on the results provided in this dissertation.

8.2.1 Extensions

Inexpensive distributed computation of eigenvalue problems. Motivated by the connectivity maintenance problem, we wish to explore ways to compute eigenvalues and eigenvectors in a distributed fashion. This, in itself, should already be possible because we can use distributed optimization methods to solve for an eigenvalue, and use it to solve the linear equation for its eigenvector, also in a distributed way. In fact, existing works in the literature [YFG+10, ASD+14] have already explored the distributed computation of algebraic connectivity. However, we intend to extend those work to the computation of all eigenvalues, and we will characterize errors by use our understanding of how eigenvalues and eigenvectors change to do a perturbation analysis. From that point, we aim to use event-triggered control to reduce computational loads by using the size of perturbation as a triggering mechanism.

Smoothness and event-triggered control implementation: In this dissertation, we have explored the design of a smooth controller, but our implementation method does not take advantage of the smoothness. Our future research will investigate the benefits from smoothness in a controller. Because the current event-triggered control framework is agnostic of the smoothness properties, we want to develop a framework that capture them in some way. For instance, we may design a trigger mechanism that, in addition to monitoring performance criteria, limits how much the control signal

can change at each trigger, and use smoothness to bound its minimum-interevent time.

Performance-based event-triggered control co-design: The performance-barrier-based trigger framework we have developed opens up a whole new set of possibilities. We are particularly interested in its application to the event-triggered control co-design problem. Controllers are not usually designed with a consideration of an event-triggered implementation. Common control design methodology, such as optimal control, may lose its effectiveness when event-triggered implementation is applied after the design process. Co-designing a feedback controller together with a trigger condition aims to solve this issue. Nevertheless, guaranteeing a desirable property for the controller is often very difficult because the controller is aperiodically updated. We believe that the performance guarantee provided by the performance-barrier-based trigger framework can be exploited in a variety of scenarios to facilitate the analysis when co-designing controllers that opportunistically update.

8.2.2 The Big Picture

In the big picture, we are motivated by the multiobjective aspect in control systems. We believe that the control objectives are performance-driven, meaning that the objectives are not absolute. Instead, each control objective has an acceptable level of performance, e.g., convergence speed in asymptotic stability and safety level by a certificate, with some diminishing return. At the same time, we believe humans have intuitions and desires in these performance criteria but cannot express them, at least not mathematically. In the long term, we want to explore how performance criteria can dictate control, what parameters are important for each objective, what kind of information can be obtained from humans, and how we should interface the human. These questions

also have a connection to research fields that we have not explored in this dissertation, i.e., data-driven control and machine learning. Our approaches in the dissertation has made the assumption, as a starting point, the knowledge of the system dynamics and the availability of the certificates. Because each human is different, the control systems need to be flexible in dealing with different humans. We believe that research towards combining the results in this dissertation with the data-driven control or machine learning will be exciting. When these challenges are explored and addressed, we imagine networks of truly smart control system serving humans, executing multiple tasks in the way that maximize its utility by observing and directly interacting with humans. This vision guides our future research.

Bibliography

[AB99] C. D. Aliprantis and K. C. Border. *Infinite Dimensional Analysis: A Hitchhiker's Guide*. Studies in Economic Theory. Springer, New York, 1999.

[Abb01] S. Abbott. *Understanding analysis*, volume 2. Springer, New York, 2001.

[ACE+19] A. D. Ames, S Coogan, M. Egerstedt, G. Notomista, K. Sreenath, and P. Tabuada. Control barrier functions: theory and applications. In *European Control Conference*, pages 3420–3431, Naples, Italy, June 2019.

[APDN16] M. Abdelrahim, R. Postoyan, J. Daafouz, and D. Nešić. Stabilization of nonlinear systems using event-triggered output feedback controllers. *IEEE Transactions on Automatic Control*, 61(9):2682–2687, 2016.

[Art83] Z. Artstein. Stabilization with relaxed controls. *Nonlinear Analysis*, 7(11):1163–1173, 1983.

[ASD+14] R. Aragues, G. Shi, D. V. Dimarogonas, C. Sagüés, K. H. Johansson, and Y. Mezouar. Distributed algebraic connectivity estimation for undirected graphs with upper and lower bounds. *Automatica*, 50(12):3253–3259, 2014.

[AXGT17] A. D. Ames, X. Xu, J. W. Grizzle, and P. Tabuada. Control barrier function based quadratic programs for safety critical systems. *IEEE Transactions on Automatic Control*, 62(8):3861–3876, 2017.

[BCM09] F. Bullo, J. Cortés, and S. Martinez. *Distributed Control of Robotic Networks*. Applied Mathematics Series. Princeton University Press, 2009.

[BDH16] D. P. Borgers, V. S. Dolk, and W. P. M. H. Heemels. Dynamic event-triggered control with time regularization for linear systems. In *IEEE Conf. on Decision and Control*, pages 1352–1357, Las Vegas, NV, 2016.

[BH14] D. P. Borgers and W. P. M. H. Heemels. Event-separation properties of event-triggered control systems. *IEEE Transactions on Automatic Control*, 59(10):2644–2656, 2014.

[BK17] P. Braun and C. M. Kellett. On (the existence of) control Lyapunov barrier functions, 2017.

[Bla99] F. Blanchini. Set invariance in control. *Automatica*, 35(11):1747–1767, 1999.

[BM07] F. Blanchini and S. Miani. *Set-Theoretic Methods in Control*. Birkhäuser, Boston, MA, 2007.

[BN21] J. Berneburg and C. Nowzari. Robust dynamic event-triggered coordination with a designable minimum interevent time. *IEEE Transactions on Automatic Control*, 66(8):3417–3428, 2021.

[Bor85] K. C. Border. *Fixed Point Theorems with Applications to Economics and Game Theory*. Cambridge University Press, Cambridge, UK, 1985.

[Boy06] S. Boyd. Convex optimization of graph Laplacian eigenvalues. In *Int. Congress of Mathematicians*, pages 1311–1319, Madrid, Spain, August 2006.

[BV04] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.

[BV09] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2009.

[CGC17] A. Cherukuri, B. Gharesifard, and J. Cortés. Saddle-point dynamics: conditions for asymptotic stability of saddle points. *SIAM Journal on Control and Optimization*, 55(1):486–511, 2017.

[CGDRON05] J. W. Crandall, M. A. Goodrich, Jr. D. R. Olsen, and C. W. Nielsen. Validating human-robot interaction schemes in multitasking environments. *IEEE Transactions on Systems, Man & Cybernetics. Part A: Systems & Humans*, 35(4):438–449, 2005.

[CKSD14] T. H. Cheng, Z. Kan, J. M. Shea, and W. E. Dixon. Decentralized event-triggered control for leader-follower consensus. In *IEEE Conf. on Decision and Control*, pages 1244–1249, Los Angeles, CA, 2014.

[Cla83] F. H. Clarke. *Optimization and Nonsmooth Analysis*. Canadian Mathematical Society Series of Monographs and Advanced Texts. Wiley, 1983.

[CS20] B. Capelli and L. Sabattini. Connectivity maintenance: Global and optimized approach through control barrier functions. In *IEEE Int. Conf. on Robotics and Automation*, pages 5590–5596, Paris, France, May 2020.

[DBH14] V. S. Dolk, D. P. Borgers, and W. P. M. H. Heemels. Dynamic event-triggered control: Tradeoffs between transmission intervals and performance. In *IEEE Conf. on Decision and Control*, pages 2764–2769, Los Angeles, CA, December 2014.

[DBH17] V. S. Dolk, D. P. Borgers, and W. P. M. H. Heemels. Output-based and decentralized dynamic event-triggered control with guaranteed $\mathcal{L}_p$-gain performance and Zeno-freeness. *IEEE Transactions on Automatic Control*, 62(1):34–49, 2017.

[Deb54] G. Debreu. Representation of a preference ordering by a numerical function. *Decision processes*, 3:159–165, 1954.

[DFJ12] D. V. Dimarogonas, E. Frazzoli, and K. H. Johansson. Distributed event-triggered control for multi-agent systems. *IEEE Transactions on Automatic Control*, 57(5):1291–1297, 2012.

[dGJ06] M. C. de Gennaro and A. Jadbabaie. Decentralized control of connectivity for multi-agent systems. In *IEEE Conf. on Decision and Control*, pages 3628–3633, San Diego, CA, December 2006.

[DH12] M. C. F. Donkers and W. P. M. H. Heemels. Output-based event-triggered control with guaranteed L_∞-gain and improved and decentralised event-triggering. *IEEE Transactions on Automatic Control*, 57(6):1362–1376, 2012.

[DMGC11] S. Durand, N. Marchand, and J. F. Guerrero-Castellanos. Simple Lyapunov sampling for event-driven control. *IFAC Proceedings Volumes*, 44(1):8724–8730, 2011.

[DPH17] V. S. Dolk, J. Ploeg, and W. P. M. H. Heemels. Event-triggered control for string-stable vehicle platooning. *IEEE Transactions on Intelligent Transportation Systems*, 18(12):3486–3500, 2017.

[FGW02] Anders Forsgren, Philip E Gill, and Margaret H Wright. Interior methods for nonlinear optimization. *SIAM review*, 44(4):525–597, 2002.

[Fie73] M. Fiedler. Algebraic connectivity of graphs. *Czechoslovak Mathematical Journal*, 23(2):298–305, 1973.

[FK96] R. A. Freeman and P. V. Kototovic. *Robust Nonlinear Control Design: State-space and Lyapunov Techniques*. Birkhauser Boston Inc., Cambridge, MA, USA, 1996.

[GA12] E. Garcia and P. J. Antsaklis. Decentralized model-based event-triggered control of networked systems. In *American Control Conference*, pages 6485–6490, Montreal, Canada, 2012.

[GCE17a] P. Glotfelter, J. Cortés, and M. Egerstedt. Nonsmooth barrier functions with applications to multi-robot systems. In *IEEE Conf. on Decision and Control*, pages 5237–5242, Melbourne, Australia, December 2017.

[GCE17b] P. Glotfelter, J. Cortés, and M. Egerstedt. Nonsmooth barrier functions with applications to multi-robot systems. *IEEE Control Systems Letters*, 1(2):310–315, 2017.

[GDF72] A. M. Geoffrion, J. S. Dyer, and A. Feinberg. An interactive approach for multi-criterion optimization, with an application to the operation of an academic department. *Management Science*, 19(4-part-1):357–368, 1972.

[Gir15] A. Girard. Dynamic triggering mechanisms for event-triggered control. *IEEE Transactions on Automatic Control*, 60:1992–1997, 2015.

[GLM+12] M. Guinaldo, D. Lehmann, J. S. Moreno, S. Dormido, and K. H. Johansson. Distributed event-triggered control with network delays and packet losses. In *IEEE Conf. on Decision and Control*, pages 1–6, Hawaii, USA, December 2012.

[GR01] C. D. Godsil and G. F. Royle. *Algebraic Graph Theory*, volume 207 of *Graduate Texts in Mathematics*. Springer, 2001.

[GS07] M. A. Goodrich and A. C. Schultz. Human-robot interaction: A survey. *Foundations and Trends in Human-Computer Interaction*, 1(3):203–275, 2007.

[GSU17] A. Gasparri, L. Sabattini, and G. Ulivi. Bounded control law for global connectivity maintenance in cooperative multi-robot systems. *IEEE Transactions on Robotics*, 33(3):700–717, 2017.

[HA17] C. E. Harriott and J. A. Adams. Towards reaction and response time metrics for real-world human-robot interaction. In *IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*, pages 799–804, Aug 2017.

[Hal69] J. K. Hale. *Ordinary Differential Equations*. Robert E. Krieger Publishing Company, 1969.

[HDI06] L. Hetel, J. Daafouz, and C. Iung. Stabilization of arbitrary switched linear systems with unknown time-varying delays. *IEEE Transactions on Automatic Control*, 51(10):1668–1674, Oct 2006.

[HDT11] W. P. M. H. Heemels, M. C. F. Donkers, and A. R. Teel. Periodic event-triggered control based on state feedback. In *IEEE Conf. on Decision and Control*, pages 2571–2576, Orlando, FL, 2011.

[HFO+17] L. Hetel, C. Fiter, H. Omran, A. Seuret, E. Fridman, J. P. Richard, and S. I. Niculescu. Recent developments on the stability of systems with aperiodic sampling: An overview. *Automatica*, 76:309–335, 2017.

[HJT12] W. P. M. H. Heemels, K. H. Johansson, and P. Tabuada. An introduction to event-triggered and self-triggered control. In *IEEE Conf. on Decision and Control*, pages 3270–3285, Maui, HI, 2012.

[HLP52] G. H. Hardy, J. E. Littlewood, and G. Polya. *Inequalities*. Cambridge University Press, Cambridge, UK, 1952.

[Inc56] E. L. Ince. *Ordinary Differential Equations*. Dover Publications, 1956.

[JE07] M. Ji and M. Egerstedt. Distributed control of multiagent systems while preserving connectedness. *IEEE Transactions on Robotics*, 23(4):693–703, 2007.

[KA18] S. Kolathaya and A. D. Ames. Input-to-state safety with control barrier functions. *IEEE Control Systems Letters*, 3(1):108–113, 2018.

[KAH17] B. A. Khashooei, D. J. Antunes, and W. P. M. H. Heemels. Output-based event-triggered control with performance guarantees. *IEEE Transactions on Automatic Control*, 62(7):3646–3652, 2017.

[Kat76] T. Kato. *Perturbation theory for linear operators*. Grundlehren der mathematischen Wissenschaften: a series of comprehensive studies in mathematics. Springer, Berlin, 1976.

[KCM15] S. S. Kia, J. Cortés, and S. Martinez. Distributed convex optimization via continuous-time coordination algorithms with discrete-time communication. *Automatica*, 55:254–264, 2015.

[Kha02] H. K. Khalil. *Nonlinear Systems*. Prentice Hall, 3 edition, 2002.

[KM06] Y. Kim and M. Mesbahi. On maximizing the second smallest eigenvalue of a state-dependent graph Laplacian. *IEEE Transactions on Automatic Control*, 51(1):116–120, 2006.

[KSC+19] S. S. Kia, B. Van Scoy, J. Cortés, R. A. Freeman, K. M. Lynch, and S. Martinez. Tutorial on dynamic average consensus: The problem, its applications, and the algorithms. *IEEE Control Systems*, 39(3):40–72, 2019.

[LD97] D. T. Luc and P. H. Dien. Differentiable selection of optimal solutions in parametric linear programming. *Proceedings of the American Mathematical Society*, 125(3):883–892, 1997.

[Lew96] A. S. Lewis. Group invariance and convex matrix analysis. *SIAM Journal on Matrix Analysis and Applications*, 17(4):927–949, 1996.

[LS85] A. Lechicki and A. Spakowski. A note on intersection of lower semicontinuous multifunctions. *Proceedings of the American Mathematical Society*, 95(1):119–122, 1985.

[LS91] Y. Lin and E. D. Sontag. A universal formula for stabilization with bounded controls. *Systems & Control Letters*, 16(6):393–397, 1991.

[LS95] Y. Lin and E. D. Sontag. Control-Lyapunov universal formulas for restricted inputs. *Control-Theory and Advanced Technology*, 10:1981–2004, 1995.

[Luc97] D. T. Luc. Smooth representation of a parametric polyhedral convex set with application to sensitivity in optimization. *Proceedings of the American Mathematical Society*, 125(2):555–567, 1997.

[LWL12] L. Li, X. Wang, and M. D. Lemmon. Stabilizing bit-rate of disturbed event triggered control systems. In *Proceedings of the 4th IFAC Conference on Analysis and Design of Hybrid Systems*, pages 70–75, Eindhoven, Netherlands, June 2012.

[LYW09] M. Luque, J. B. Yang, and B. Y. H. Wong. Project method for multiobjective optimization based on gradient projection and reference points. *IEEE Transactions on Systems, Man & Cybernetics. Part A: Systems & Humans*, 39(4):864–879, 2009.

[MAT09] M. Mazo Jr., A. Anta, and P. Tabuada. On self-triggered control for linear systems: Guarantees and complexity. In *European Control Conference*, pages 3767–3772, Budapest, Hungary, August 2009.

[Mic56] E. Michael. Continuous selections. i. *Annals of Mathematics*, 63(2):361–382, 1956.

[MPA13] B. Morris, M. J. Powell, and A. D. Ames. Sufficient conditions for the Lipschitz continuity of QP-based multi-objective control of humanoid robots. In *IEEE Conf. on Decision and Control*, pages 2920–2926, Florence, Italy, Dec 2013.

[MPA15] B. J. Morris, M. J. Powell, and A. D. Ames. Continuity and smoothness properties of nonlinear optimization-based feedback controllers. In *IEEE Conf. on Decision and Control*, pages 151–158, Osaka, Japan, Dec 2015.

[MRW08] K. Miettinen, F. Ruiz, and A. P. Wierzbicki. Introduction to multiobjective optimization: Interactive approaches. In Jürgen Branke, Kalyanmoy Deb, Kaisa Miettinen, and Roman Słowiński, editors, *Multiobjective Optimization: Interactive and Evolutionary Approaches*, pages 27–57. Springer, Berlin, Heidelberg, 2008.

[MT11] M. Mazo Jr. and P. Tabuada. Decentralized event-triggered control over wireless sensor/actuator networks. *IEEE Transactions on Automatic Control*, 56(10):2456–2461, 2011.

[NGC19] C. Nowzari, E. Garcia, and J. Cortés. Event-triggered control and communication of networked systems for multi-agent consensus. *Automatica*, 105:1–27, 2019.

[NMJ05] K. B. Ngo, R. Mahony, and Z. P. Jiang. Integrator backstepping using barrier functions for systems with multiple state constraints. In *IEEE Conf. on Decision and Control*, pages 8306–8312, Seville, Spain, December 2005.

[OC19] P. Ong and J. Cortés. Universal formula for smooth safe stabilization. In *IEEE Conf. on Decision and Control*, pages 2373–2378, Nice, France, December 2019.

[OC21a] P. Ong and J. Cortés. Opportunistic robot control for interactive multiobjective optimization under human performance limitations. *Automatica*, 123:109263, 2021.

[OC21b] P. Ong and J. Cortés. Performance-barrier-based event-triggered control with applications to network systems. *IEEE Transactions on Automatic Control*, 2021. Submitted.

[OCSC21] P. Ong, B. Capelli, L. Sabattini, and J. Cortés. Nonsmooth control barrier function design of continuous constraints for network connectivity maintenance. *International Journal of Robotics Research*, 2021. Submitted.

[OLV02] G. Oriolo, A. De Luca, and M. Vendittelli. WMR control via dynamic feedback linearization: design, implementation, and experimental validation. *IEEE Transactions on Control Systems Technology*, 10(6):835–852, 2002.

[PD18] S. Peitz and M. Dellnitz. A survey of recent trends in multiobjective optimal control – surrogate models, feedback control and objective reduction. *Mathematical and Computational Applications*, 23, June 2018.

[PJ04] S. Prajna and A. Jadbabaie. Safety verification of hybrid systems using barrier certificates. In *Hybrid systems: Computation and Control*, pages 477–492, Philadelphia, PA, March 2004.

[PSH19] R. Postoyan, R.G. Sanfelice, and W. P. M. H. Heemels. Inter-event times analysis for planar linear event-triggered controlled systems. In *IEEE Conf. on Decision and Control*, pages 1662–1667, Nice, France, December 2019.

[PSOB+17] S. Peitz, K. SchÃďfer, S. Ober-BlÃűbaum, J. Eckstein, U. KÃűhler, and M. Dellnitz. A multiobjective MPC approach for autonomously driven electric vehicles. *IFAC-PapersOnLine*, 50(1):8674–8679, 2017.

[PST+15] J. R. Peters, V. Srivastava, G. S. Taylor, A. Surana, M. P. Eckstein, and F. Bullo. Human supervisory control of robotic teams: Integrating cognitive modeling with engineering design. *IEEE Control Systems*, 35(6):57–80, 2015.

[PTNA15] R. Postoyan, P. Tabuada, D. Nešić, and A. Anta. A framework for the event-triggered stabilization of nonlinear systems. *IEEE Transactions on Automatic Control*, 60(4):982–996, 2015.

[RC16] D. Richert and J. Cortés. Distributed linear programming with event-triggered communication. *SIAM Journal on Control and Optimization*, 54(3):1769–1797, 2016.

[RJ16] M. Z. Romdlony and B. Jayawardhana. Stabilization with guaranteed safety using control Lyapunov-barrier function. *Automatica*, 66:39–47, 2016.

[Roc70] R. T. Rockafellar. *Convex Analysis*. Princeton University Press, 1970.

[Sak82] M. Sakawa. Interactive multiobjective decision making by the sequential proxy optimization technique: SPOT. *European Journal of Operational Research*, 9(4):386–396, 1982.

[SC09] M. D. Schuresko and J. Cortés. Distributed motion constraints for algebraic connectivity of robotic networks. *Journal of Intelligent and Robotic Systems*, 56(1-2):99–126, 2009.

[SC12] M. D. Schuresko and J. Cortés. Distributed tree rearrangements for reachability and robust connectivity. *SIAM Journal on Control and Optimization*, 50(5):2588–2620, 2012.

[SCS13] L. Sabattini, N. Chopra, and C. Secchi. Decentralized connectivity maintenance for cooperative control of mobile robotic systems. *International Journal of Robotics Research*, 32(12):1411–1423, 2013.

[SFK+06] A. Steinfeld, T. Fong, D. B. Kaber, M. Lewis, J. Scholtz, A. C. Schultz, and M. A. Goodrich. Common metrics for human-robot interaction. In *HRI*, pages 33–40, Salt Lake City, Utah, USA, March 2006.

[SG17] M. T. Shaikh and M. A. Goodrich. Design and evaluation of adverb palette: A GUI for selecting tradeoffs in multi-objective optimization problems. In *ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, pages 389–397, March 2017.

[Son89] E. D. Sontag. A universal construction of Artstein's theorem on nonlinear stabilization. *Systems & Control Letters*, 13(2):117–123, 1989.

[Son99] E. D. Sontag. Control-Lyapunov functions. In V. Blondel, E. D. Sontag, M. Vidyasagar, and J. C. Willems, editors, *Open Problems in Mathematical Systems and Control Theory*, pages 211–216. Springer London, 1999.

[Son08] E. D. Sontag. Input to state stability: Basic concepts and results. *Nonlinear and Optimal Control Theory*, 1932:163–220, 2008.

[SP94] D. Shevitz and B. Paden. Lyapunov stability theory of nonsmooth systems. *IEEE Transactions on Automatic Control*, 39(9):1910–1914, 1994.

[SP11] A. Seuret and C. Prieur. Event-triggered sampling algorithms based on a Lyapunov function. In *IEEE Conf. on Decision and Control*, pages 6128–6133, Orlando, FL, 2011.

[SS96] E. D. Sontag and H. J. Sussmann. General classes of control-Lyapunov functions. In R. Jeltsch and M. Mansour, editors, *Stability Theory*, pages 87–96. Birkhäuser Basel, 1996.

[Sti18] G. Still. Lectures on parametric optimization: An introduction. *Optimization Online*, 2018.

[SW95] E. D. Sontag and Y. Wang. On characterizations of input-to-state stability with respect to compact sets. In *Nonlinear Control Systems Design*, pages 203–208. Elsevier, 1995.

[Tab07] P. Tabuada. Event-triggered real-time scheduling of stabilizing control tasks. *IEEE Transactions on Automatic Control*, 52(9):1680–1685, 2007.

[Tal61] G. M. Tallis. The moment generating function of the truncated multi-normal distribution. *Journal of the Royal Statistical Society. Series B (Methodological)*, 23(1):223–229, 1961.

[Tal65] G. M. Tallis. Plane truncation in normal populations. *Journal of the Royal Statistical Society. Series B (Methodological)*, 27(2):301–307, 1965.

[TC14] P. Tallapragada and N. Chopra. Decentralized event-triggering for control of nonlinear systems. *IEEE Transactions on Automatic Control*, 59(12):3312–3324, 2014.

[TGT09] K. P. Tee, S. Ge, and F. Tay. Barrier Lyapunov functions for the control of output-constrained nonlinear systems. *Automatica*, 45:918–927, 2009.

[TOCA21] A. J. Taylor, P. Ong, J. Cortés, and A. Ames. Safety-critical event triggered control via input-to-state safe barrier functions. *IEEE Control Systems Letters*, 5(3):749–754, 2021.

[VMB09] M. Velasco, P. Martí, and E. Bini. On Lyapunov sampling for event-driven controllers. In *IEEE Conf. on Decision and Control*, pages 6238–6243, Shanghai, China, 2009.

[WA07] P. Wieland and F. Allgöwer. Constructive safety using control barrier functions. *IFAC Proceedings Volumes*, 40(12):462–467, 2007.

[WAJ12] J. Weimer, J. Araújo, and K. H. Johansson. Distributed event-triggered estimation in networked systems. *IFAC Proceedings Volumes*, 45(9):178–185, 2012.

[War89a] F. W. Warner. *Foundations of Differentiable Manifolds and Lie Groups*, volume 94 of *GTM*. Springer, 2 edition, 1989.

[War89b] Frank W. Warner. *Foundations of Differentiable Manifolds and Lie Groups*. Number 94 in Graduate Texts in Mathematics. Springer, New York, 2 edition, 1989.

[WL09] P. Wan and M. D. Lemmon. Event-triggered distributed optimization in sensor networks. In *Symposium on Information Processing of Sensor Networks*, pages 49–60, San Francisco, CA, 2009.

[WL11] X. Wang and M. D. Lemmon. Event-triggering in distributed networked control systems. *IEEE Transactions on Automatic Control*, 56(3):586–601, 2011.

[WRGL15] W. Wu, S. Reimann, D. Görges, and S. Liu. Suboptimal event-triggered control for time-delayed linear systems. *IEEE Transactions on Automatic Control*, 60(5):1386–1391, May 2015.

[XCC+18] B. Xin, L. Chen, J. Chen, H. Ishibuchi, K. Hirota, and B. Liu. Interactive multiobjective optimization: A review of the state-of-the-art. *IEEE Access*, 6:41256–41279, 2018.

[XTGA15] X. Xu, P. Tabuada, J. W. Grizzle, and A. D. Ames. Robustness of control barrier functions for safety critical control. *IFAC-PapersOnLine*, 48(27):54–61, 2015.

[Yan99] J. B. Yang. Gradient projection and local region search for multiobjective optimisation. *European Journal of Operational Research*, 112(2):432–459, 1999.

[YFG+10] P. Yang, R. A. Freeman, G. J. Gordon, K. M. Lynch, S. S Srinivasa, and R. Sukthankar. Decentralized estimation and control of graph connectivity for mobile sensor networks. *Automatica*, 46(2):390–396, 2010.

[YJ06] H. Yuqing and H. Jianda. Generalized point wise min-norm control based on control Lyapunov functions. In *2007 Chinese Control Conference*, pages 404–408, July 2006.

[ZP05] M. M. Zavlanos and G. J. Pappas. Controlling connectivity of dynamic graphs. In *IEEE Conf. on Decision and Control and European Control Conference*, pages 6388–6393, Seville, Spain, December 2005.

[ZP15] M. Zavlanos and G. Pappas. Connectivity of dynamic graphs. In J. Baillieul and T. Samad, editors, *Encyclopedia of Systems and Control*. Springer, New York, 2015.