Nonsmooth Control Barrier Function Design of Continuous Constraints for Network Connectivity Maintenance

Pio Ong [a]   Beatrice Capelli [b]   Lorenzo Sabattini [b]   Jorge Cortés [c]

[a] *Department of Mechanical and Civil Engineering, California Institute of Technology, USA,*
pioong@caltech.edu

[b] *Department of Sciences and Methods for Engineering, University of Modena and Reggio Emilia, Italy,*
{beatrice.capelli, lorenzo.sabattini}@unimore.it

[c] *Department of Mechanical and Aerospace Engineering, University of California, San Diego, USA,*
cortes@ucsd.edu

Abstract

This paper 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. Our solution uses the algebraic connectivity of the multi-robot interconnection topology as a control barrier function and critically embraces its nonsmooth nature. We take advantage of the understanding of how Laplacian eigenvalues behave as their multiplicities change, in combination with the flexibility provided by the concept of control barrier function, to carefully design additional constraints that guarantee the resulting optimization-based controller is continuous and maintains network connectivity. 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.

Key words: Multi-Robot Systems, Connectivity Maintenance, Nonsmooth Control Barrier Functions, Algebraic Graph Theory

1 Introduction

Multi-robot systems can accomplish a variety of tasks through coordinated behavior in many scenarios. Such systems are more versatile, more robust, and better performing than a single specialized robot. To enjoy these advantages, cooperative strategies for multi-robot systems must overcome a number of hurdles, including scalability, graceful degradation with respect to agent failures, and connectivity maintenance, which is the focus of this work. In fact, the ability to interchange information across the network is critical to accomplish emergent coordinated behavior, such as flocking, agreement, coverage, rendezvous, etc., cf. (Bullo et al., 2009; Mesbahi and Egerstedt, 2010; Cortés and Egerstedt, 2017) and references therein. Connectivity maintenance is hence a fundamental aspect of cooperative strategies which must be considered in conjunction with the objectives that the multi-robot systems seek to achieve. This integration must be carefully balanced to avoid getting robots stuck in place or display erratic changes in their motions to avoid losing connectivity.

Motivated by these observations, this paper investigates how to ensure connectivity while efficiently managing constraints related to the objective of the multi-robot system, with a special emphasis on the continuity of the resulting feedback controller.

Literature Review

Multi-robot systems rely on coordination among agents to achieve their goals. In order to be able to interchange information across the network, the interaction graph must be connected. The concept of algebraic connectivity (Godsil and Royle, 2001) of a graph, also known as Fiedler eigenvalue (Fiedler, 1973), 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 (Bullo et al., 2009; Zavlanos and Pappas, 2015), 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., Ji and Egerstedt, 2007), which limits the graph to one arrangement. This method can be improved by considering instead multiple-hops neighbors and allowing rearrangements in the edges (Zavlanos and Pappas, 2005; Schuresko and Cortés, 2012), 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 (Boyd, 2006; Kim and Mesbahi, 2006). The idea is to find a robot motion that will increase the algebraic connectivity. A decentralized implementation of this idea is explored in (de Gennaro and Jadbabaie, 2006). Nevertheless, maximizing the algebraic connectivity in all scenarios can be overly restrictive. In this regard, (Sabattini et al., 2013; Schuresko and Cortés, 2009) introduce more flexibility by allowing algebraic connectivity to decrease when its value is large.

Our connectivity maintenance solution here is based on the concept of Control Barrier Function (CBF) from the safety-critical control literature. Control Barrier Functions (Wieland and Allgöwer, 2007) build on the barrier certificate (Prajna and Jadbabaie, 2004) notion, and is used to find choices of control inputs that makes the certificate increase, guaranteeing forward invariance of a desired set. The CBF idea can be refined further by abandoning the monotonicity of the certificate. This idea is related to the concept of practical stability with Lyapunov functions, with an additional restriction on the evolution of the certificate within the desired set (Ögren et al., 2006). It is later formalized in the context of safety (Ames et al., 2019) by using Nagumo theorem (Blanchini and Miani, 2007) as the basis for set invariance. This refined version introduces the concept of letting the certificate also decrease depending on the level of safety. In the context of connectivity maintenance, CBFs flexibly allows algebraic connectivity to decrease as long as the graph does not become disconnected. CBFs are employed in both aforementioned connectivity maintenance approaches in (Egerstedt et al., 2018) and (Capelli and Sabattini, 2020), respectively. Regarding the latter, there is no guarantee on the continuity of the proposed feedback controller because of the lack of smoothness of the algebraic connectivity. Here instead, we rely on Nonsmooth Control Barrier Functions (NCBF) (Glotfelter et al., 2017), a generalization of CBF, to properly account for the nonsmoothness of algebraic connectivity and ensure the continuity of the resulting feedback controller.

Controllers that utilize CBFs are typically based on optimization formulations, (see e.g., Ames et al., 2017, 2019). For this type of controllers, there are multiple approaches to determine continuity. Using perturbation theory, the paper (Morris et al., 2015) studies the smoothness properties of optimization-based controllers with CBFs but the result is only applicable to continuously differentiable CBFs. From a set-valued theory perspective, (Freeman and Kototovic, 1996) shows continuity of minimum-norm controllers, i.e., when the objective function is a norm. For more general objective functions (like the one considered here), we rely on Berge Maximum Theorem (Aliprantis and Border, 1999), a well-known result in parametric optimization, to guarantee continuity of the feedback controller.

Statement of Contributions

This paper considers a multi-robot system with fully actuated first-order dynamics. The underlying interaction network is described by a continuously differentiable proximity graph. We address the problem of maintaining global connectivity of the multi-robot system that is operating under some nominal control constraints. The contributions of the paper are threefold. The first contribution is the synthesis of two different set-valued constraint maps for global connectivity maintenance. The proposed constraints are based on NCBFs and are able to handle, in a continuous way, the abrupt changes caused by the jumps in multiplicity of the algebraic connectivity as a function of the network state. Establishing this fact relies on a careful application of various notions and results from set-valued analysis. With our proposed constraints, the resulting optimization-based controller is continuous, which thereby guarantees the existence of a solution and avoids issues such as chattering in its discrete-time implementation. Our second contribution deals with the well-posedness of the considered problem. As we allow for the possibility of the network to have control constraints beyond connectivity maintenance, one question that we answer is in regard to the existence of a solution to our problem, i.e., a continuous controller that can both respect control constraints and maintain network connectivity. We use a generalization of Artstein's theorem (Artstein, 1983) to deduce a mild and verifiable condition that guarantees our problem is well-posed. Our final contribution are the continuity results, as a function of the network state, for any intersection of eigenspaces of the graph Laplacian. We rely on this result to study how the algebraic connectivity changes. Since the generalized gradient of algebraic connectivity is related to its associated eigenspace, the results add to the literature on regularity of algebraic connectivity. We believe our second and third contributions may have useful applications beyond the subject matter of this paper. We conclude the paper by illustrating the effectiveness of our results in a resource gathering problem, both in simulations and an experiment. The problem consists in a group of robots trying to reach their assigned target locations which cannot do so 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.

A preliminary version of this paper appeared at the IEEE Conference on Decision and Control (Ong et al., 2021). The added value of the present work is justified by the following additions: (i) a more general control synthesis problem formulation that incorporates a nominal constraint map, which results in a more challenging technical analysis; (ii) the generalization of Artstein's theorem to formulate a reasonable assumption for the well-posedness of the problem; (iii) the establishment of the continuity property of merged eigenspaces of the graph Laplacian as a function of the network state, which was only speculated in the preliminary version of the paper; (iv) the new simulation example along with a validation of the results in an experiment on four small-wheeled robots. In addition, we provide throughout the paper all the necessary background and discussions on intuitions behind the proposed ideas.

2 Preliminaries

This section introduces basic notation and key concepts from graph theory, set-valued and nonsmooth analysis, and Nonsmooth Control Barrier Functions.

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 $[m : n] = \{m, \ldots, n\}$, 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 $\mathcal{S}$ is represented by $\text{co}(\mathcal{S})$. 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 $\mathbb{S}^n = \{\mathbf{v} \in \mathbb{R}^n \mid \|\mathbf{v}\| = 1\}$. The open ball of radius $\delta > 0$ centered at $\mathbf{x}^* \in \mathbb{R}^N$ is $\mathbb{B}_\delta(\mathbf{x}^*) = \{\mathbf{x} \in \mathbb{R}^N \mid \|\mathbf{x} - \mathbf{x}^*\| < \delta\}$. 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 $\alpha(0) = 0$. Moreover, $\text{supp}(\mathbf{f})$ is the support of the function f, i.e., the set of $\mathbf{x}$ where $f(\mathbf{x}) \neq 0$.

2.2 Graphs and Laplacian Spectrum

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 $D_{ii} = \sum_{j \in V} \mathbf{A}_{ij}$. 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* (Bullo et al., 2009). In such a case, the Laplacian matrix $\mathbf{x} \mapsto \mathbf{L}(\mathbf{x})$ is then also a function of the 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.

2.3 Continuity of Set-Valued Maps

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. Throughout the paper, 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:

$$\mathcal{U}(\mathbf{x}) = \{\mathbf{u} \in \mathbb{R}^M \mid g(\mathbf{x}, \mathbf{u}) \leq 0\}. \tag{1}$$

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

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 2.1 *(Set-Valued Map Continuity (Border, 1985)): 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 $\delta > 0$ 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 is (hemi)continuous if it is (hemi)continuous for all $\mathbf{x}$. Interestingly for set-valued maps of the form (1), 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 2.2 *(UHC Requirements (Still, 2018, 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 $\mathbf{x}$.* $\square$

Lemma 2.3 *(LHC Requirements (Still, 2018, Lem 5.2)): Assume g is continuous. If $\mathcal{U}$ has a nonempty interior and is convex-valued, then $\mathcal{U}$ is LHC.* $\square$

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

Lemma 2.4 *(Intersection of UHC maps (Border, 1985, 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 $\mathbf{x}$.* $\square$

Lemma 2.5 *(Intersection of LHC maps (Lechicki and Spakowski, 1985, 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 $\mathbf{x}$.* $\square$

---

[1] Sometimes referred to as semicontinuity, see e.g., (Lechicki and Spakowski, 1985).

2.4 Nonsmooth Analysis

Here we present basic notions of nonsmooth analysis following (Clarke, 1983). 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

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

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,

$$\dot{\mathbf{x}} = f(\mathbf{x}, \mathbf{u}), \tag{2}$$

with $f : \mathbb{R}^N \times \mathbb{R}^M \to \mathbb{R}^N$ measurable and essentially locally bounded, where $\mathbf{x}$ is the state and $\mathbf{u}$ is the control input. The weak set-valued Lie derivative (Glotfelter et al., 2017; Shevitz and Paden, 1994) is

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

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), then

$$\frac{d}{dt} h(\mathbf{x}(t)) \in \mathcal{L}_f h(\mathbf{x}(t), \mathbf{u}(t)), \ a.e. \ t \geq 0. \tag{3}$$

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.

2.5 Nonsmooth Control Barrier Functions

We use Nonsmooth Control Barrier Functions (NCBF) (Glotfelter et al., 2017) to establish forward invariance of a desired set. Consider the dynamical system (2) 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 (3), 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 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.

3 Problem Statement

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

$$\dot{x}_r = u_r, \quad \forall r \in [n], \tag{4}$$

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 $\mathbf{x} = [x_1^\top, \ldots, x_n^\top]^\top \in \mathbb{R}^N$ and $\mathbf{u} = [u_1^\top, \ldots, u_n^\top]^\top \in \mathbb{R}^N$. We use the shorthand notation $f_{\mathrm{si}} : \mathbb{R}^N \times \mathbb{R}^N \to \mathbb{R}^N$ to refer compactly to the dynamics (4) 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 [3].

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 performances and asymptotic guarantees. Continuity is an important property, both from a theoretical and practical viewpoint. Regarding the former, continuity guarantees the existence of Carathéodory (in fact, classical) solutions (Hale, 1969, 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 and takes the form

$$\mathbf{k}_{\mathrm{opt}}(\mathbf{x}) = \underset{\mathbf{u} \in \mathcal{U}(x)}{\operatorname{argmin}} J(\mathbf{x}, \mathbf{u}), \tag{5}$$

where $J : \mathbb{R}^N \times \mathbb{R}^N \to \mathbb{R}$ is a cost function encoding some desirable objective (e.g., minimal deviation from a prescribed input, minimum-energy control specifications) and $\mathcal{U} : \mathbb{R}^N \rightrightarrows \mathbb{R}^N$ is a set-valued map encoding constraints on the control input at each $\mathbf{x}$ (e.g., bounds on magnitude, stability performance using control Lyapunov function). This formulation is flexible as it can address simultaneously different performance requirements: the map $\mathcal{U}$ can be itself

---

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

[3] This assumption is satisfied by commonly employed weight assignments (Schuresko and Cortés, 2009; Gasparri et al., 2017).

4

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 $\mathbf{x} \mapsto \mathcal{U}_{\mathrm{nom}}(\mathbf{x})$, defined via a function $g_{\mathrm{nom}} : \mathbb{R}^N \times \mathbb{R}^N \to \mathbb{R}^{d_{\mathrm{nom}}}$ as

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

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 (4) operating with the optimization-based controller (5). Design the constraint map $\mathcal{U}$ so that:

- the controller $\mathbf{k}_{\mathrm{opt}}$ is continuous;
- the nominal constraint map is respected, i.e., $\mathcal{U} \subseteq \mathcal{U}_{\mathrm{nom}}$;
- the underlying graph $\mathcal{G}$ remains connected at all time. ●

We make the following assumptions on the cost function J and the nominal constraint map $\mathcal{U}_{\mathrm{nom}}$ to make sure Problem 1 is solvable. First, $\mathcal{U}_{\mathrm{nom}}$ should be large enough so that, at each state, there exists a control that can simultaneously maintain connectivity and satisfy the nominal constraints (we formulate this assumption mathematically later in our technical discussion, cf. Remark 5.3). As one can expect, continuity of $\mathbf{k}_{\mathrm{opt}}$ is related to continuity of the cost function J and the constraint map $\mathcal{U}$. In this regard, Berge Maximum Theorem (Aliprantis and Border, 1999, Thm. 17.31) states that, if J and $\mathcal{U}$ are continuous, $\mathcal{U}$ is compact-valued, and the resulting $\mathbf{k}_{\mathrm{opt}}$ is single-valued, then $\mathbf{k}_{\mathrm{opt}}$ is continuous. Based on this result, we make the following continuity assumption.

Assumption 3.1 *(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 $\mathcal{U}_{\mathrm{nom}}$ 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 3.2 *(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

$$\mathcal{J}_{\bar{\mathbf{u}}}(\mathbf{x}) = \{\mathbf{u} \in \mathbb{R}^N \mid \|J(\mathbf{x}, \mathbf{u})\| \leq \|J(\mathbf{x}, \bar{\mathbf{u}}(\mathbf{x}))\| + \delta_J\} \quad (6)$$

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.

4 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 (Capelli and Sabattini, 2020) (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

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

with $\varepsilon \in \mathbb{R}_{>0}$. The parameter ε is introduced here to provide robustness and ensure the safe set is closed. Its introduction makes the safe set smaller and hence the safety specification more conservative. Let $\mathbf{x} \mapsto h(\mathbf{x}) = \lambda_2(\mathbf{x}) - \varepsilon$ be our candidate NCBF. Resorting to the discussion of Section 2.5, we specify a constraint map for the purpose of connectivity maintenance.

Lemma 4.1 *(Connectivity Maintenance Constraint Map): Consider the multi-robot system (4) 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*

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

If $\mathbf{k}(\mathbf{x}) \in \mathcal{U}_{\mathrm{cm}}(\mathbf{x})$ 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 4.1 is a direct result of using $h(\mathbf{x}) = \lambda_2(\mathbf{x}) - \varepsilon$ as a NCBF, cf. (Glotfelter et al., 2017, Thm.3). Our first

attempt to utilize the result is to design an optimization-based controller (5) naively defined with the connectivity maintenance constraint map,

$$\mathbf{k}_{\mathrm{dis}}(\mathbf{x}) := \underset{u \in \mathcal{U}_{\mathrm{cm}}(\mathbf{x})}{\mathrm{argmin}} \ J(\mathbf{x}, \mathbf{u}). \qquad (7)$$

Unfortunately, this controller is not guaranteed to be continuous. This can cause a number of undesired phenomena. For example, sample-and-hold implementations of the controller may exhibit chattering behavior because its continuous-time counterpart is not continuous (cf., Fig. 3c in Sec. 7). More importantly for our problem, the connectivity maintenance result provided by Lemma 4.1 is only guaranteed along Carathéodory solutions, and such solutions might not exist if the controller is not continuous. As such, discontinuous controllers like $\mathbf{k}_{\mathrm{dis}}$ may fail to maintain connectivity of the multi-robot system.

The discontinuity issue arises because $\mathcal{U}_{\mathrm{cm}}$ itself is not continuous and does not meet the requirement of Berge Maximum Theorem. To pinpoint the root cause of the discontinuity of $\mathcal{U}_{\mathrm{cm}}$, 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., (Schuresko and Cortés, 2009, Lem. 1) and (Lewis, 1996, 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. (Schuresko and Cortés, 2009, Thm. 1),

$$\partial\phi_m(\mathbf{L}) = \mathrm{co}\{\mathbf{v}_m \mathbf{v}_m^\top \mid \mathbf{v}_m \in \mathcal{V}_m(\mathbf{L})\}, \qquad (8)$$

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 (Clarke, 1983, Thm. 2.3.10), the expression for the weak set-valued Lie derivative (Glotfelter et al., 2017, Rmk. 2.1) of λ_m with respect to the system (4) is

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

In the constraint map $\mathcal{U}_{\mathrm{cm}}$, 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 4.2 *(Equivalent Minimization of the Eigenvalue's Set-Valued Lie Derivative): Consider the multi-robot system (4). For $m \in [N]$, let $(\mathbf{x}, \mathbf{u}) \mapsto \mu_m(\mathbf{x}, \mathbf{u})$,*

$$\mu_m(\mathbf{x}, \mathbf{u}) := \min_{\mathbf{v} \in \mathcal{V}_m(\mathbf{L}(\mathbf{x}))} \mathbf{v}^\top \Big(\sum_{i \in [N]} \frac{\partial \mathbf{L}}{\partial \mathbf{x}_i}\mathbf{u}_i\Big)\mathbf{v}. \qquad (9)$$

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

PROOF. Let $\mathbf{P} \in \partial\phi_m(\mathbf{L}(\mathbf{x}))$ be the element of the generalized gradient (8) corresponding to the minimum value

in $\mathcal{L}_{f_{\mathrm{si}}}\lambda_m(\mathbf{x}, \mathbf{u})$, i.e.,

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

From the Carathéodory theorem of convex hulls (Rockafellar, 1970, Thm. 17.1) [4], since $\mathbf{P} \in \mathbb{R}^{n \times n}$, there exists $n^2 + 1$ (not necessarily distinct) points $\{\mathbf{P}_j\}_{j=1}^{n^2+1}$, each belonging to the set $\partial\phi_m'(\mathbf{L}(\mathbf{x})) = \{\mathbf{v}_m\mathbf{v}_m^\top \mid \mathbf{v}_m \in \mathcal{V}_m(\mathbf{L}(\mathbf{x}))\}$, such that $\mathbf{P}$ is a convex combination of $\{\mathbf{P}_j\}_{j=1}^{n^2+1}$. That is, there exist $0 \leq \sigma_j \leq 1$ with $\sum_{j=1}^{n^2+1} \sigma_j = 1$ such that

$$\min \mathcal{L}_{f_{\mathrm{si}}}\lambda_m(\mathbf{x}, \mathbf{u}) = \Big(\sum_{j=1}^{n^2+1} \sigma_j \mathbf{P}_j\Big) \cdot \Big(\sum_{i \in [N]} \frac{\partial \mathbf{L}}{\partial \mathbf{x}_i}\mathbf{u}_i\Big) \qquad (10)$$

$$= \sum_{j=1}^{n^2+1} \sigma_j \underbrace{\Big(\mathbf{P}_j \cdot \Big(\sum_{i \in [N]} \frac{\partial \mathbf{L}}{\partial \mathbf{x}_i}\mathbf{u}_i\Big)\Big)}_{\geq \min \mathcal{L}_{f_{\mathrm{si}}}\lambda_m(\mathbf{x}, \mathbf{u})}.$$

From this, there must exist at least one $\mathbf{P}_j$ such that

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

(Otherwise, if $\mathbf{P}_j \cdot \Big(\sum_{i \in [N]} \frac{\partial \mathbf{L}}{\partial \mathbf{x}_i}\mathbf{u}_i\Big) > \min \mathcal{L}_{f_{\mathrm{si}}}\lambda_m(\mathbf{x}, \mathbf{u})$ for every $j \in [n^2 + 1]$, one would reach from (10) and $\sum_{j=1}^{n^2+1} \sigma_j = 1$ the contradiction that $\min \mathcal{L}_{f_{\mathrm{si}}}\lambda_m(\mathbf{x}, \mathbf{u}) > \min \mathcal{L}_{f_{\mathrm{si}}}\lambda_m(\mathbf{x}, \mathbf{u})$). Hence, it is equivalent to compute the minimization of the Lie derivative on the set $\partial\phi_m'$ rather than its convex closure $\partial\phi_m$, that is,

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

$$= \min_{\mathbf{P}' \in \partial\phi_m'(\mathbf{L}(\mathbf{x}))} \mathbf{P}' \cdot \Big(\sum_{i \in [N]} \frac{\partial \mathbf{L}}{\partial \mathbf{x}_i}\mathbf{u}_i\Big)$$

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

$$= \mu_m(\mathbf{x}, \mathbf{u}),$$

where we have used a property of the Frobenius product in the last step. This concludes the proof. □

Lemma 4.2 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 change in value simply because of the abrupt change in the minimization constraint, as illustrated

---

[4] The theorem is applicable to the matrix space $\mathbb{R}^{n \times n}$ because its vectorized version is the n^2-dimensional real space $\mathbb{R}^{n^2}$.
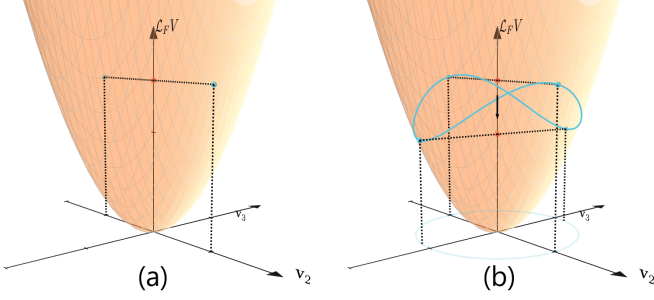
Fig. 1. A jump in the minimum value of the set-valued Lie derivative may occur when the corresponding eigenspace expands. Plot (a) shows the optimal value of a quadratic function over a normalized eigenspace in one dimension (which consists simply of two points, $\mathbf{v}_2$ and $-\mathbf{v}_2$). Plot (b) shows that the optimal value of the quadratic function abruptly drops when the normalized eigenspace increases by one dimension to a circle.

in Fig. 1. We rely on this key insight to synthesize our design in the next section.

5 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 4. 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 6 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 (9). One possible fix is to augment the eigenspace preemptively so that there is no abrupt expansion. For $\mathcal{I} \subseteq [n]$, consider

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

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

$$\mu_{\mathcal{I}}(\mathbf{x}, \mathbf{u}) := \min_{\mathbf{v} \in \mathcal{V}_{\mathcal{I}}(\mathbf{x})} \mathbf{v}^{\top} \Big(\sum_{i \in [N]} \frac{\partial \mathbf{L}}{\partial \mathbf{x}_i} \mathbf{u}_i \Big) \mathbf{v}, \qquad (11)$$

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 $\mathcal{U}_{\mathrm{cm}}$, 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 5.1 *(Strict Connectivity Constraint Map for Continuous Controller): Consider the multi-robot system (4). Given a locally Lipschitz extended class $\mathcal{K}$ function α, define the constraint map*

$$\mathcal{U}_{\mathrm{str}}(\mathbf{x}) := \big\{ \mathbf{u} \in \mathbb{R}^N \mid \mu_{[2:n]}(\mathbf{x}, \mathbf{u}) \geq -\alpha(\lambda_2(\mathbf{x}) - \varepsilon) \big\}. \quad (12)$$

If, for each $\mathbf{x}$, there exists a control input $\mathbf{u} \in \mathbb{R}^N$ that strictly satisfies the constraint map $\mathcal{U}_{\mathrm{str}} \cap \mathcal{U}_{\mathrm{nom}}(\mathbf{x})$, then under Assumptions 3.1 and 3.2, the optimization-based controller

$$\mathbf{k}_{\mathrm{str}}(\mathbf{x}) := \operatorname*{argmin}_{\mathbf{u} \in \mathcal{U}_{\mathrm{str}} \cap \mathcal{U}_{\mathrm{nom}}(\mathbf{x})} J(\mathbf{x}, \mathbf{u}) \quad (13)$$

is continuous on $\mathcal{C}_{\varepsilon}$. In addition, the closed-loop feedback $\mathbf{u} = \mathbf{k}_{\mathrm{str}}(\mathbf{x})$ 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 5.1 provides a solution to Problem 1, it is undoubtedly conservative. By design, the constraint map $\mathcal{U}_{\mathrm{str}}$ 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, for situations 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 in Section 7.

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,

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

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 5.2 *(Aggregate Connectivity Constraint Map for Continuous Controller): Consider the multi-robot system (4). Given a locally Lipschitz extended class $\mathcal{K}$ function α, define the constraint map*

$$\mathcal{U}_{\mathrm{agg}}(\mathbf{x}) := \bigcap_{m \in [2:n]} \mathcal{U}_{[2:m]}(\mathbf{x}). \quad (14)$$

If, for each $\mathbf{x}$, there exists a control input $\mathbf{u} \in \mathbb{R}^N$ that strictly satisfies the constraint map $\mathcal{U}_{\mathrm{agg}} \cap \mathcal{U}_{\mathrm{nom}}(\mathbf{x})$, then under Assumptions 3.1 and 3.2, the optimization-based controller

$$\mathbf{k}_{\mathrm{agg}}(\mathbf{x}) := \operatorname*{argmin}_{\mathbf{u} \in \mathcal{U}_{\mathrm{agg}}(\mathbf{x}) \cap \mathcal{U}_{\mathrm{nom}}(\mathbf{x})} J(\mathbf{x}, \mathbf{u}) \quad (15)$$

7

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

The idea behind the design of the aggregate constraint (14) 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 $\mathcal{U}_{[2:m]}(\mathbf{x})$ 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 $\mathcal{U}_{\mathrm{agg}}$, and the fact that it experiences no abrupt change there is enough to prevent $\mathcal{U}_{\mathrm{agg}}$ from changing abruptly at that state.

Both constraint maps (12) and (14) ensure continuity of the corresponding optimization-based controller and solve Problem 1. In general, for $m \in [2:n]$, one has $\mathcal{U}_{\mathrm{str}} \subseteq \mathcal{U}_{[2:m]}$ because $\mu_{[2:n]} \leq \mu_{[2:m]}$ and $\lambda_m \geq \lambda_2$. Therefore, $\mathcal{U}_{\mathrm{str}} \subseteq \mathcal{U}_{\mathrm{agg}}$, with equality holding on those states where $\lambda_n(\mathbf{x}) = \lambda_2(\mathbf{x})$. Consequently, $\mathcal{U}_{\mathrm{agg}}$ imposes less conservative constraints than $\mathcal{U}_{\mathrm{str}}$. This is because the aggregate constraint $\mathcal{U}_{\mathrm{agg}}$ only gradually becomes stricter as the gap between each eigenvalue and the lowest $\lambda_m - \lambda_2$ gets smaller, unlike the strict constraint $\mathcal{U}_{\mathrm{str}}$ that is agnostic to the gap.

Remark 5.3 (*Strictly Satisfying Feasible Controls Requirement*): We note that both Theorems 5.1 and 5.2 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 6.5), 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 5.1 and 5.2, 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. $\qquad\bullet$

Remark 5.4 (*Computation of Proposed Controllers*): For each $\mathbf{x}$, the computation of the controllers $\mathbf{k}_{\mathrm{str}}$ and $\mathbf{k}_{\mathrm{agg}}$ 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. (Boyd and Vandenberghe, 2009; Rockafellar, 1970), 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

$$
\mu_{[2:m]}(\mathbf{x}, \mathbf{u}) = \min_{\mathbf{v} \in \mathcal{V}_{[2:m]}(\mathbf{x})} \mathbf{v}^\top \Big(\sum_{i \in [N]} \frac{\partial \mathbf{L}}{\partial \mathbf{x}_i} \mathbf{u}_i \Big) \mathbf{v}
$$

$$
= \min_{\xi \in \mathcal{S}^{m-1}} \xi^\top [\mathbf{v}]_{2:m}^\top(\mathbf{x}) \Big(\sum_{i \in [N]} \frac{\partial \mathbf{L}}{\partial \mathbf{x}_i} \mathbf{u}_i \Big) [\mathbf{v}]_{2:m}(\mathbf{x}) \xi
$$

$$
:= \min_{\xi \in \mathbb{S}^{m-1}} \xi^\top Z_m(\mathbf{x}, \mathbf{u}) \xi,
$$

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. $\qquad\bullet$

6 Technical Analysis of the Proposed Solutions

This section provides the proofs of the results presented in Section 5. 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.

6.1 Properties of Merged Lower Bounds

We first examine the properties of functions $\mu_{\mathcal{I}}$ of the form (11) 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 6.1 (*Continuity of Normalized Merged Eigenspaces*): Let $\mathbf{L} : \mathbb{R}^N \rightarrow \mathrm{Sym}_n$ 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. $\qquad\square$

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 (Aliprantis and Border, 1999, Thm. 17.31).

Corollary 6.2 (*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 $j \notin \mathcal{I}$. $\qquad\square$

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 6.3 *(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

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

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

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.

Remark 6.4 *(More General System Dynamics):* Note that Lemma 6.3 is the only instance in our technical treatment where we have exploited the specific form of the single-integrator dynamics (4). In fact, all of our results are valid for more general system dynamics, as long as one can prove the concavity of the merged lower bound. This is the case, for instance, for control-affine systems, for which our results hold. We have kept the discussion limited to single-integrator dynamics for simplicity of presentation. $\qquad\bullet$

6.2 Equivalent Constraint Maps

In general, the constraint maps $\mathcal{U}_{\mathrm{str}}$ and $\mathcal{U}_{\mathrm{agg}}$ might not be UHC because they are unbounded. To make sure the requirements of Lemma 2.2 as well as Berge Maximum Theorem are met, we explain here how to consider, following Section 3, equivalent constraint maps that are compact-valued. This procedure involves using sublevel sets of the cost function J, which are compact due to Assumption 3.2. 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 (6). 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 5.1 and 5.2, such continuous feasible control function always exists.

Lemma 6.5 *(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 $\mathbf{u}_{\mathrm{int}}(\mathbf{x})$ denote the control input such that $g(\mathbf{x}, \mathbf{u}_{\mathrm{int}}(\mathbf{x})) < 0$. Due to continuity of g, there exists a neighborhood of $\mathbf{x}$, denoted by $\mathcal{W}(\mathbf{x})$, such that $\mathbf{u}_{\mathrm{int}}(\mathbf{x}) \in \mathcal{U}(\mathbf{x}')$ for all $\mathbf{x}' \in \mathcal{W}(\mathbf{x})$. 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. (Warner, 1989, Thm. 1.11). In other words, for each j, there exists an $\mathbf{x}$ such that $\mathrm{supp}(\psi_j)$ is a subset of $\mathcal{W}(\mathbf{x})$, each of which has an associated control $\mathbf{u}_{\mathrm{int}}^j \in \mathcal{U}(\mathbf{x})$ for $\mathbf{x} \in \mathrm{supp}(\psi_j)$. Then we define $\bar{\mathbf{u}}(\mathbf{x}) = \sum_j \psi_j(\mathbf{x}) \mathbf{u}_{\mathrm{int}}^j$, which satisfies the statement due to convexity of the map $\mathcal{U}$. $\qquad\square$

Lemma 6.5 is a generalization of Artstein's Theorem (Artstein, 1983, 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. Because the functions defining $\mathcal{U} = \mathcal{U}_{\mathrm{str}}$ are continuous, we can directly apply Lemma 6.5. On the other hand, $\mathcal{U} = \mathcal{U}_{\mathrm{agg}}$ is defined with discontinuous functions; nevertheless, from its construction, one can still employ the argument presented in the proof of Lemma 6.5 (i.e., there exists $\mathbf{u}_{\mathrm{int}}$ at each $\mathbf{x}$ belonging to $\mathcal{U}_{\mathrm{agg}}(\mathbf{x}')$ for all $\mathbf{x}'$ in a neighborhood $\mathcal{W}$ of $\mathbf{x}$, and so on). As a result, for each of the cases $\mathcal{U} = \mathcal{U}_{\mathrm{str}}$ and $\mathcal{U} = \mathcal{U}_{\mathrm{agg}}$, there exists 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 3.2. Then according to Lemmas 2.2 and 2.3, it is also continuous due to the functions $\bar{\mathbf{u}}$ and J being continuous, cf. Assumption 3.1. We then consider the intersections $\mathcal{U}_{\mathrm{str}} \cap \mathcal{U}_{\mathrm{nom}} \cap \mathcal{J}_{\bar{\mathbf{u}}}$ and $\mathcal{U}_{\mathrm{agg}} \cap \mathcal{U}_{\mathrm{nom}} \cap \mathcal{J}_{\bar{\mathbf{u}}}$, where the inclusion of $\mathcal{J}_{\bar{\mathbf{u}}}$ make these constraint maps compact-valued. For the purpose of our analysis, we equivalently define $\mathbf{k}_{\mathrm{str}}$ and $\mathbf{k}_{\mathrm{agg}}$ with these constraint maps as the constraint to the optimization.

6.3 Continuity of the Connectivity Maintenance Controllers

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

PROOF. [Proof of Theorem 5.1] Consider the constraint set $\mathcal{U}_{\mathrm{str}} \cap \mathcal{U}_{\mathrm{nom}} \cap \mathcal{J}_{\bar{\mathbf{u}}}$. We note the following: (i) all the functions defining the constraint map are continuous due to Assumption 3.1 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 2.2 and 2.3, to show continuity of this constraint map. By Berge Maximum Theorem (Aliprantis and Border, 1999, 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 4.1 ensures $\lambda_2(\mathbf{x}(t)) \geq \varepsilon$, and the proof concludes. $\qquad\square$

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. [Proof of Theorem 5.2] 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 2.2 and 2.3 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 5.1

Upper Hemicontinuity: We begin by consider the partial constraint map $\mathcal{J}_{\bar{u}} \cap \mathcal{U}_{[2:n]}$. This set-valued map is continuous on $\mathcal{C}_\varepsilon$ because of the continuity of $\mu_{[2:n]}$. Consider its intersection with $\mathcal{J}_{\bar{u}} \cap \mathcal{U}_{[2:n-1]}$. 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 $\mathcal{J}_{\bar{u}} \cap (\bigcap_{m \in [n-1:n]} \mathcal{U}_{[2:m]})(\mathbf{x})$ is exactly the same set as $\mathcal{J}_{\bar{u}} \cap \mathcal{U}_{[2:n]}(\mathbf{x})$ 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 $\mathcal{J}_{\bar{u}} \cap \mathcal{U}_{[2:n]}$, we can conclude UHC for the intersection $\mathcal{J}_{\bar{u}} \cap (\bigcap_{m \in [n-1:n]} \mathcal{U}_{[2:m]})$ at $\mathbf{x}$ where $\lambda_n(\mathbf{x}) = \lambda_{n-1}(\mathbf{x})$. Elsewhere, the intersection can be proven UHC directly via Lemma 2.2, 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.

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 6.3, 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 2.3 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 2.3, so it is LHC everywhere on $\mathcal{C}_\varepsilon$. Then using Lemma 2.5, 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

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

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 2.5, concluding the proof. $\quad\square$

7 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:

$$u_{\text{nom},r}(x_r) = v_{\text{nom}} \frac{e_r(x_r)}{\|e_r(x_r)\|}, \; \forall r \in [n], \qquad (16)$$

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 $\mathbf{k}_{\text{nom}}(\mathbf{x}) = [u_{\text{nom},1}(x_1)^\top, \ldots, u_{\text{nom},4}(x_4)^\top]^\top$, our cost function

$$J(\mathbf{x}, \mathbf{u}) = \|\mathbf{u} - \mathbf{k}_{\text{nom}}(\mathbf{x})\|^2 \qquad (17)$$

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,

$$\mathcal{U}_{\text{nom}}(\mathbf{x}) = \left\{ \mathbf{u} \in \mathbb{R}^N \mid k v_{\text{nom}}^2 - u_{\text{nom},P}^\top u_P \leq 0 \right\}, \qquad (18)$$

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 change the value of P to correspond to the index of the next robot that has yet to achieve its goal, whose task we want to prioritize.

Note that the objective (17) and the nominal constraint (18) verify both Assumptions 3.1 and 3.2 for any prioritized 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 prioritized robot changes). In addition, we can verify strict feasibility of the constraints since constraint (18) only affects the prioritized robot P, and we can have all other robots move towards the prioritized robot to maintain connectivity (because we have assumed the single-integrator dynamics (4)). We show that our proposed controllers from Theorem 5.1 and 5.2 are continuous for the duration between events when the prioritized robots achieve its goal. For both our simulations and our experiment, we use the projected saddle-point dynamics (Cherukuri et al., 2017) to solve the convex optimization problems at each given state $\mathbf{x}$ along the trajectory. The saddle-point dynamics involve performing a gradient descent in the primal variable $\mathbf{u}$, and a gradient ascent in the dual variables (projected onto the positive orthant to ensure they remain nonnegative) associated with the constraints. We chose this gradient-based method because we know how to compute the gradient of the merged lower bounds that appear in our constraints, cf. Remark 5.4. The computation of our controllers is done in MATLAB®.

7.1 Simulations

Our simulations highlight the differences among the different controllers: $\mathbf{k}_{\text{dis}}$, defined in (7), $\mathbf{k}_{\text{str}}$, defined in (13), and $\mathbf{k}_{\text{agg}}$, defined in (15), each with $\alpha(\lambda) = \lambda$. 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. 2 reports the eigenvalues of the Laplacian matrix during the simulations. It is clear how both the aggregate (Fig. 2a) and the strict controller (Fig. 2b) maintain the connectivity constraint, unlike the discontinuous one that leads to disconnection (Fig. 2c).

The overall performance corroborates our hypothesis that the aggregate controller will outperform the strict one. Figs. 3a and 3b show the continuous input produced by the aggregate and the strict controllers, and Fig. 3c shows the discontinuous one generated by the discontinuous controller. It takes the aggregate controller (1542 steps) shorter time than the strict controller (2199 steps) to complete all the tasks in the simulation. We expected this result because the conservatism in the strict controller leads to fewer eligible robot formations. On the other hand, the aggregate controller allows the robots to get in better positions for the subsequent tasks. We note that there is no guarantee that the aggregate controller will always do better. This is because the two controllers generate two different trajectories and the strict controller may take the robots to better formations despite having fewer options available. Nevertheless, we believe that the aggregate controller will outperform the stricter one in general, based on our earlier reasoning. Regarding the discontinuous controller, we observe chattering throughout most of the trajectory. Such

(a) Aggregate controller

(b) Strict controller
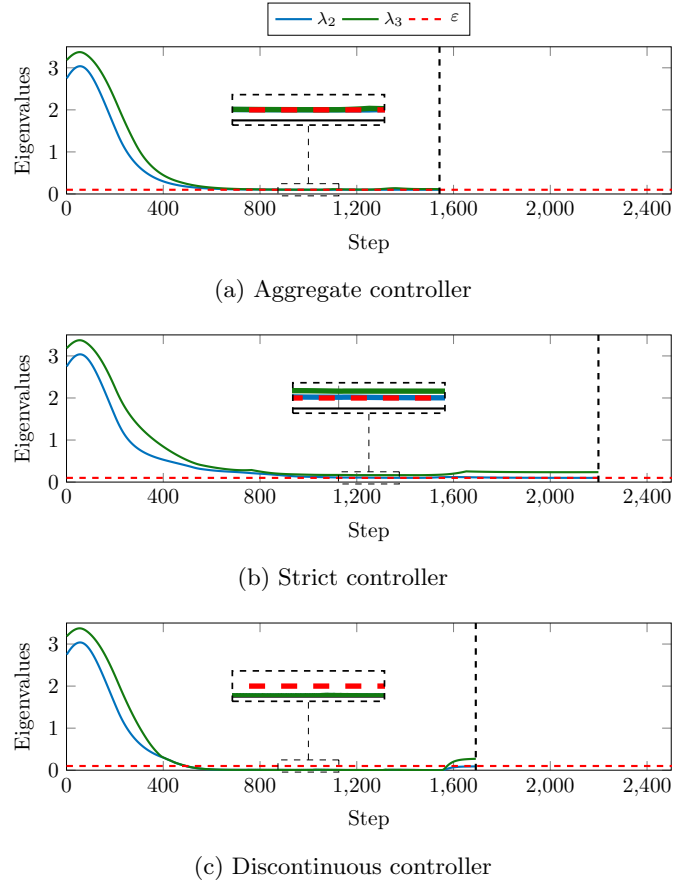
(c) Discontinuous controller

Fig. 2. Eigenvalue evolution during the simulations under the different controllers. The *dashed black* lines represent the end of the network task.

behavior is undesirable because it is difficult to implement in real systems, and it may also cause other unexpected issues. For example, despite theoretically being the least conservative out of the three controllers, the discontinuous controller does not complete the tasks before the aggregate one.

Next, Fig. 4 reports the evolution of the function defining the nominal constraint map $\mathcal{U}_{\text{nom}}$ under the aggregate and the strict 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. 3c.

7.2 Experimental Validation

We also carry out an experiment for the same resource gathering problem, cf. Fig. 5. 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 (Oriolo et al., 2002). We tested only the proposed controller $\mathbf{k}_{\text{agg}}$, as the simulations in Section 7.1 verified that it is the best both in terms of performance and
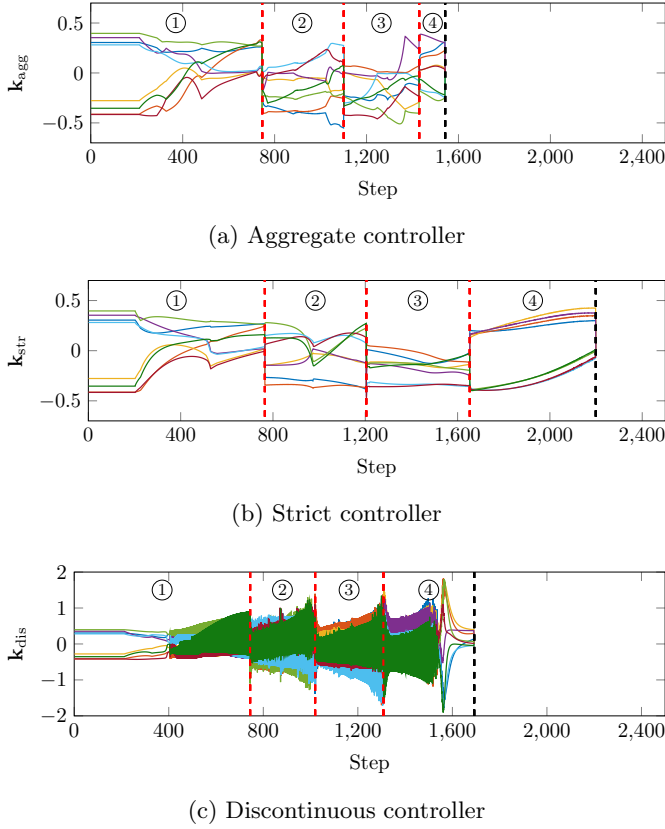
11

(a) Aggregate controller

(b) Strict controller

(c) Discontinuous controller

Fig. 3. Control inputs during the simulations of different controllers. We report all the components of the control input for each robot.

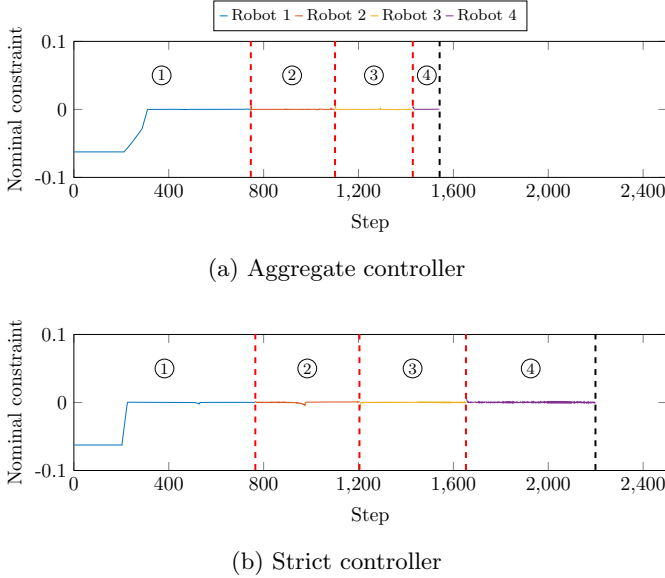(a) Aggregate controller

(b) Strict controller

Fig. 4. Nominal constraint (18) 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.

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. 6 reports the eigenvalue evolution during the experiment, further confirming the effectiveness of the proposed

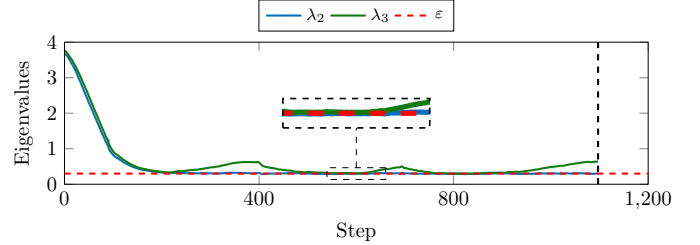Fig. 5. Experimental setup: 4 ePucks and their corresponding targets.

Fig. 6. Eigenvalue evolution during the experiment. The *dashed black* lines represents the end of the task.

method in maintaining connectivity. Fig. 7 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 to see how the robots that have already reached their target cooperate to connectivity. Fig. 8 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. For a given measurement of the state $\mathbf{x}$, we run the saddle-point dynamics as many steps as possible so that the control $\mathbf{u}$ solution is near optimal. Nevertheless, we report the computation of the gradient of the merged lower bounds as the bottleneck in the execution of the saddle-point dynamics. These are equivalent to solving eigenvalue problems, so they scale up cubically with the number of agents, $\mathcal{O}(N^3)$. In our future work, we plan to investigate ways of avoiding the calculation of these gradients when it is not necessary. 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. 9.

8   Conclusions

We have considered 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
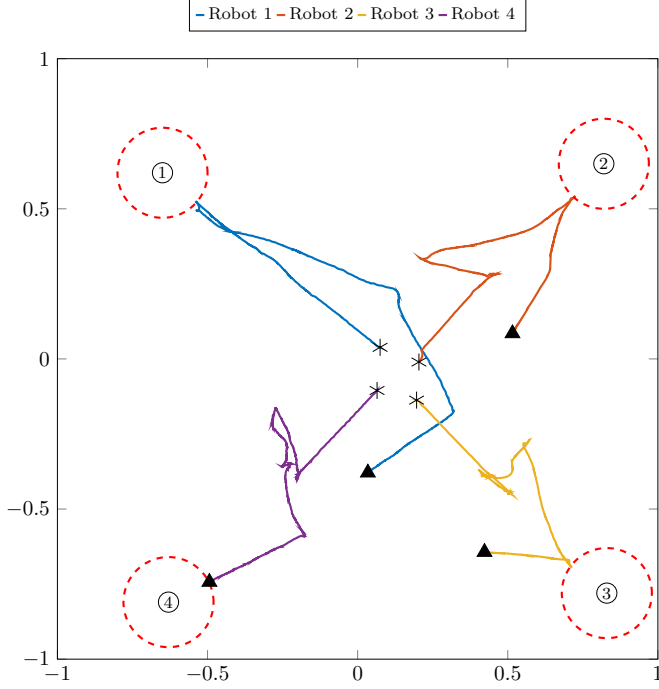
12

Fig. 7. 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.
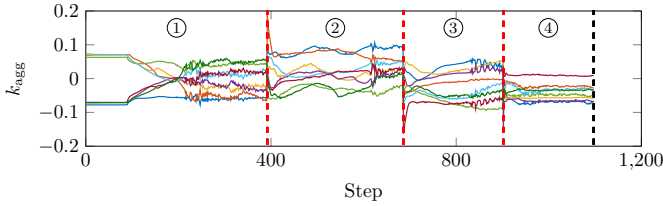

Fig. 8. Control input applied to the robots in the experiment. We report all the components of the control input for each robot. These inputs have been transformed via input-output state-feedback linearization to be executed by the robots.
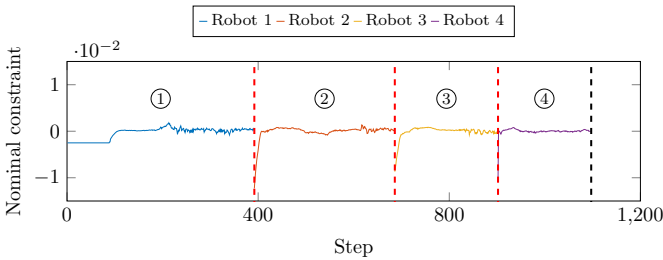

Fig. 9. Nominal constraint (18) during the experiment. The *dashed red* lines represent the instant in which the robot priority changes, due to the fact that a target has been reached. The number for each time slot corresponds to the robot with the highest priority.

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. Future work will investigate the extension of our approach to ensure Lipschitzness of the controller, the application of the methodology proposed here to the synthesis of distributed controllers for connectivity maintenance, the development of formal guarantees for robustness under perturbations and discrete-time implementations of our design, resource-aware design of aperiodic sample-and-hold implementations of the proposed controllers that do not require solving convex optimization problems at every state, and the extension of our results to more general nonlinear systems.

A   Appendix

Here we provide several results on the continuity properties of eigenspaces, with the ultimate goal of establishing that the merged eigenspaces are continuous functions of the state as long as their dimensions remain constant, cf. Theorem 6.1 [5].

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,

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

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

A.1   Upper Hemicontinuity of Merged Eigenspaces

For the analysis of (A.1), 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

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

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 $\mathbf{B}^{(i)}(\mathbf{x}) :=$

---

[5] Although this result is seemingly intuitive, we have not found it in the literature. There are results (e.g., (Kato, 1976, 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.

13

$\mathbf{D}^{(i)}(\mathbf{x}) - \lambda_i(\mathbf{x})\mathbf{I}$, we can equivalently write each eigenequation with

$$\begin{bmatrix} \mathbf{B}_{aa}^{(i)}(x) & \mathbf{B}_{ab}^{(i)}(x) \\ \mathbf{B}_{ab}^{(i)}(x)^\top & \mathbf{B}_{bb}^{(i)}(x) \end{bmatrix} \begin{bmatrix} \mathbf{w}_{i,a} \\ \mathbf{w}_{i,b} \end{bmatrix} = 0, \qquad \text{(A.2)}$$

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 A.1 *(UHC of individual eigenspaces): Consider a continuous function* $\mathbf{L}: \mathbb{R}^N \to \text{Sym}_n$. *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 $\mathbf{B}_{aa}^{(i)}(\mathbf{x}^*)$ is invertible, there exists $\bar{\delta}_\mathbf{x} > 0$ such that $\mathbf{B}_{aa}^{(i)}(\mathbf{x})$ remains invertible for each $\mathbf{x} \in \mathbb{B}_{\bar{\delta}_\mathbf{x}}(\mathbf{x}^*)$. From (A.2),

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

Because $\mathbf{B}_{aa}^{(i)}(\mathbf{x})^{-1}\mathbf{B}_{ab}^{(i)}(\mathbf{x})$ 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 $\|\mathbf{B}_{aa}^{(i)}(\mathbf{x})^{-1}\mathbf{B}_{ab}^{(i)}(\mathbf{x})\|_F < \delta_\mathbf{w}/\sqrt{2}$ for all $\mathbf{x} \in \mathbb{B}_{\delta_\mathbf{x}}(\mathbf{x}^*)$. Then,

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

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. $\square$

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

Theorem A.2 *(UHC of Merged Eigenspaces): Consider a continuous function* $\mathbf{L}: \mathbb{R}^N \to \text{Sym}_n$. *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 (A.1), 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 A.1 as follows.

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

A.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 A.3 *(LHC of Merged Eigenspaces): Consider a continuous function* $\mathbf{L}: \mathbb{R}^N \to \text{Sym}_n$. *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 (A.1) can be rewritten as

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

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,

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

so that $\mathbf{w}_{i,b}$ has the dimension of $|\mathcal{I}|$. The matrix $\mathbf{D}_{aa}^{\mathcal{I}}(\mathbf{x}^*)$ 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 $\mathbf{D}_{aa}^{\mathcal{I}}(\mathbf{x}) - \lambda_i(\mathbf{x})\mathbf{I}$ 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,

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

Due to continuity of the matrix $\mathbf{D}^{\mathcal{I}}$ and the fact that $\mathbf{D}_{ab}^{\mathcal{I}}(\mathbf{x})$ 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

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

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 (Bullo et al., 2009, 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 here 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. $\square$

References

C. D. Aliprantis and K. C. Border. *Infinite Dimensional Analysis: A Hitchhiker's Guide*. Studies in Economic Theory. Springer, New York, 1999. ISBN 9783540658542.

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.

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.

Z. Artstein. Stabilization with relaxed controls. *Nonlinear Analysis*, 7(11):1163–1173, 1983.

F. Blanchini and S. Miani. *Set-Theoretic Methods in Control*. Birkhäuser, Boston, MA, 2007. ISBN 9780817646066.

K. C. Border. *Fixed Point Theorems with Applications to Economics and Game Theory*. Cambridge University Press, Cambridge, UK, 1985. ISBN 9780521388085.

S. Boyd. Convex optimization of graph Laplacian eigenvalues. In *Int. Congress of Mathematicians*, pages 1311–1319, Madrid, Spain, August 2006.

S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, Cambridge, UK, 2009. ISBN 0521833787.

F. Bullo, J. Cortés, and S. Martinez. *Distributed Control of Robotic Networks*. Applied Mathematics Series. Princeton University Press, Princeton, NJ, 2009. ISBN 978-0-691-14195-4.

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.

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.

F. H. Clarke. *Optimization and Nonsmooth Analysis*. Canadian Mathematical Society Series of Monographs and Advanced Texts. Wiley, New York, 1983. ISBN 047187504X.

J. Cortés and M. Egerstedt. Coordinated control of multi-robot systems: A survey. *SICE Journal of Control, Measurement, and System Integration*, 10(6):495–503, 2017.

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.

M. Egerstedt, J. N. Pauli, G. Notomista, and S. Hutchinson. Robot ecology: Constraint-based control design for long duration autonomy. *Annual Reviews in Control*, 46:1–7, 2018.

M. Fiedler. Algebraic connectivity of graphs. *Czechoslovak Mathematical Journal*, 23(2):298–305, 1973.

R. A. Freeman and P. V. Kototovic. *Robust Nonlinear Control Design: State-space and Lyapunov Techniques*. Birkhäuser, Boston, MA, 1996. ISBN 0-8176-3930-6.

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.

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.

C. D. Godsil and G. F. Royle. *Algebraic Graph Theory*, volume 207 of *Graduate Texts in Mathematics*. Springer, 2001. ISBN 0387952411.

J. K. Hale. *Ordinary Differential Equations*. Wiley, New York, 1969. ISBN 9780471340904.

M. Ji and M. Egerstedt. Distributed coordination control of multiagent systems while preserving connectedness. *IEEE Transactions on Robotics*, 23(4):693–703, 2007.

T. Kato. *Perturbation Theory for Linear Operators*. Grundlehren der mathematischen Wissenschaften: a series of comprehensive studies in mathematics. Springer, Berlin, 1976.

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.

A. Lechicki and A. Spakowski. A note on intersection of lower semicontinuous multifunctions. *Proc. of the American Mathematical Society*, 95(1):119–122, 1985.

A. S. Lewis. Group invariance and convex matrix analysis. *SIAM Journal on Matrix Analysis and Applications*, 17(4):927–949, 1996.

M. Mesbahi and M. Egerstedt. *Graph Theoretic Methods in Multiagent Networks*. Applied Mathematics Series. Princeton University Press, 2010.

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.

P. Ögren, A. Backlund, T. Harryson, L. Kristensson, and P. Stensson. Autonomous UCAV strike missions using behavior control Lyapunov functions. In *AIAA Guidance, Navigation, and Control Conference and Exhibit*, page 6197, August 2006.

P. Ong, B. Capelli, L. Sabattini, and J. Cortés. Network connectivity maintenance via nonsmooth control barrier functions. In *IEEE Conf. on Decision and Control*, pages 4780–4785, Austin, TX, December 2021.

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.

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.

R. T. Rockafellar. *Convex Analysis*. Princeton University Press, Princeton, NJ, 1970. ISBN 9780691015866.

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.

M. D. Schuresko and J. Cortés. Distributed motion constraints for algebraic connectivity of robotic networks. *Journal of Intelligent and Robotic Systems*, 56(1):99–126, 2009.

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.

D. Shevitz and B. Paden. Lyapunov stability theory of nonsmooth systems. *IEEE Transactions on Automatic Control*, 39(9):1910–1914, 1994.

G. Still. Lectures on parametric optimization: An introduction. *Optimization Online*, 2018.

F. W. Warner. *Foundations of Differentiable Manifolds and Lie Groups*. Number 94 in Graduate Texts in Mathematics. Springer, New York, 2 edition, 1989. ISBN 9780387908946.

P. Wieland and F. Allgöwer. Constructive safety using control barrier functions. *IFAC Proceedings Volumes*, 40(12):462–467, 2007.

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.

M. M. Zavlanos and G. J. Pappas. Connectivity of dynamic graphs. In J. Baillieul and T. Samad, editors, *Encyclopedia of Systems and Control*. Springer-Verlag, 2015.