Distributed implementation of control barrier functions for multi-agent systems

Xiao Tan and Dimos V. Dimarogonas, *Senior Member, IEEE*

***Abstract*— In this work, we propose a distributed implementation framework for control barrier functions induced quadratic programs for multi-agent systems. The quadratic program aims at minimally modifying nominal local controllers, which relate to the underlying system tasks, while always respecting a single coupling constraint which relates to system safety. Unlike previous implementations, no approximation or pre-allocation of the coupling constraint over the agents is needed. Instead, to solve the quadratic problem exactly, an auxiliary variable is assigned to each agent and then locally updated and transmitted among agents. The proposed distributed implementation ensures that the control barrier function constraint is enforced at every time instant, and the optimal to the quadratic program control signal is achieved in finite time. The efficacy of our method is demonstrated through two numerical examples.**

***Index Terms*— Decentralized control, Constrained control, Control barrier functions**

I. INTRODUCTION

ENSURING safety for dynamical systems has been under discussion for a long time in the literature. One interpretation of system safety is through the notion of set forward invariance, i.e., the system state should always remain in a safe set once it starts inside. Control barrier functions (CBF), initially proposed by [1] and later developed in [2]–[4], provide a point-wise linear constraint on the input, and by enforcing this constraint at every state, the forward invariance of the safety set is guaranteed. In order to enforce this constraint, a computationally efficient, modular implementation leveraging quadratic programs is introduced that aims to modify a pre-designed nominal controller to be safe in a minimal invasive manner. This methodology has been widely investigated and applied with practical success.

There are many works extending the CBF framework to multi-agent systems, with various safety criteria including inter-collision avoidance [5]–[7], connectivity maintenance [7], [8], and temporal logic tasks [9]. However, in all these works, the CBF induced quadratic program is either solved in a centralized manner [6], [8], i.e, by a central module that has access to the states of every agent, or using a pre-allocation scheme [5], [9] that distributes the linear constraint among the agents involved. While the obtained solution is feasible with the empirically designed allocation scheme, the optimality of the original quadratic program is generally lost.

In essence, one can view the CBF-induced quadratic program (QP) among agents as a special case of a distributed optimization problem with time-varying coupling constraints among them. In [10], the authors propose an online ADMM-based distributed optimization scheme that solves a relaxed optimization problem with synchronized updates. Other distributed optimization algorithms [11], [12] could also be applied to this problem. Although all these optimization-based algorithms converge to the optimal solution fairly quickly, no theoretical guarantees can be asserted regarding the satisfaction of the safety-certifying constraints during the solution iterations. This can potentially lead to the system trajectory moving out of the safe set.

Moreover, [5], [9], [10] assume that each agent has access to (part of) the states of the other agents that share a same coupling constraint. [13], instead, does not assume any specific communication structure; however, it can only deal with a certain form of CBF candidates and only an approximate solution is obtained.

In this work, we consider the CBF-induced quadratic program for multi-agent systems with a general connected communication graph, and propose a distributed implementation scheme such that the modified control signal is optimal to this QP. To achieve this goal, we introduce an equivalent quadratic program with an auxiliary decision variable. The optimality condition of this auxiliary variable to the equivalent QP is characterized. The distributed implementation scheme consists of a local quadratic program and a local adaptation of the auxiliary variable such that the optimality condition is satisfied in finite time. We show that, with our proposed implementation scheme, 1) the optimal solution to the CBF-induced QP is achieved in finite time, and 2) the CBF constraint is satisfied for all time. To our best knowledge, this is the first paper that simultaneously achieves the optimal to the CBF-induced QP control signal and guarantees system safety for a multi-agent system in a distributed way. Applications to a static quadratic program problem and a consensus task with a stacked state boundedness safety criterion are demonstrated in the simulation.

This work was supported by the Swedish Research Council (VR), the Swedish Foundation for Strategic Research (SSF), the ERC CoG LEAFHOUND, the EU CANOPIES project, and the Knut and Alice Wallenberg Foundation (KAW).

Xiao Tan and Dimos V. Dimarogonas are with the Division of Decision and Control Systems, School of EECS, Royal Institute of Technology (KTH), 100 44 Stockholm, Sweden (Email: xiaotan, dimos@kth.se).

II. Preliminary and Problem formulation

Notation: The operator $\nabla : C^1(\mathbb{R}^n) \to \mathbb{R}^n$ is defined as the gradient $\frac{\partial}{\partial x}$ of a scalar-valued differentiable function with respect to $\boldsymbol{x}$. The Lie derivatives of a function $h(\boldsymbol{x})$ for the system $\dot{\boldsymbol{x}} = \mathfrak{f}(\boldsymbol{x}) + \mathfrak{g}(\boldsymbol{x})\boldsymbol{u}$, where $\mathfrak{f} : \mathbb{R}^n \to \mathbb{R}^n$, $\mathfrak{g} : \mathbb{R}^n \to \mathbb{R}^{n\times m}$, are denoted by $L_{\mathfrak{f}}h = \nabla h^\top \mathfrak{f}(\boldsymbol{x}) \in \mathbb{R}$ and $L_{\mathfrak{g}}h = \nabla h^\top \mathfrak{g}(\boldsymbol{x}) \in \mathbb{R}^{1\times m}$, respectively. The interior and boundary of a set $\mathscr{A}$ are denoted $\text{Int}(\mathscr{A})$ and $\partial\mathscr{A}$, respectively. A continuous function $\alpha : [0, a) \to [0, \infty)$ for $a \in \mathbb{R}_{>0}$ is a *class $\mathcal{K}$ function* if it is strictly increasing and $\alpha(0) = 0$. A continuous function $\alpha : (-b, a) \to (-\infty, \infty)$ for $a, b \in \mathbb{R}_{>0}$ is an *extended class $\mathcal{K}$ function* if it is strictly increasing and $\alpha(0) = 0$. $\text{blk}(\boldsymbol{g}_1, \boldsymbol{g}_2, ..., \boldsymbol{g}_n)$ denotes a block diagonal matrix with its diagonal blocks $\boldsymbol{g}_1, \boldsymbol{g}_2, ..., \boldsymbol{g}_n$, where $\boldsymbol{g}_i, i = 1, .., n$, can be either a vector or a matrix. For $x \in \mathbb{R}$, $\text{sign}(x) := \begin{cases} 1, & x>0; \\ 0, & x=0; \\ -1, & x<0. \end{cases}$ For any $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^n$, $\text{sign}(\boldsymbol{x}) := (\text{sign}(x_1), \text{sign}(x_2), ..., \text{sign}(x_n))$ and $\boldsymbol{x} \star \boldsymbol{y}$ means that $x_i \star y_i, i \in \{1, 2, ..., n\}$ for $\star \in \{>, \geq, <, \leq\}$. $\mathbf{1}_n$ is a vector in $\mathbb{R}^n$ with all entries to be one.

Consider a multi-agent system with N agents indexed by $\mathcal{I} = \{1, 2, 3, ..., N\}$. The dynamics of agent $i \in \mathcal{I}$ is given by $\dot{\boldsymbol{x}}_i = \mathfrak{f}_i(\boldsymbol{x}_i) + \mathfrak{g}_i(\boldsymbol{x}_i)\boldsymbol{u}_i$, where the state $\boldsymbol{x}_i \in \mathbb{R}^{n_i}$, and the control input $\boldsymbol{u}_i \in \mathbb{R}^{m_i}$, $\mathfrak{f}_i(\boldsymbol{x}_i)$, $\mathfrak{g}_i(\boldsymbol{x}_i)$ are locally Lipschitz functions in $\boldsymbol{x}_i$. We denote the stacked state $\boldsymbol{x} := (\boldsymbol{x}_1^\top, \boldsymbol{x}_2^\top, ..., \boldsymbol{x}_N^\top)^\top \in \mathbb{R}^n, n := \sum_{i\in\mathcal{I}} n_i$, the stacked control input $\boldsymbol{u} := (\boldsymbol{u}_1^\top, \boldsymbol{u}_2^\top, ..., \boldsymbol{u}_N^\top)^\top \in \mathbb{R}^m, m = \sum_{i\in\mathcal{I}} m_i$, the stacked vector fields $\mathfrak{f} = (\mathfrak{f}_1^\top, \mathfrak{f}_2^\top, ..., \mathfrak{f}_N^\top)^\top$ and $\mathfrak{g} = \text{blk}(\mathfrak{g}_1, \mathfrak{g}_2, ..., \mathfrak{g}_N)$. Thus, the stacked dynamics is obtained as

$$\dot{\boldsymbol{x}} = \mathfrak{f}(\boldsymbol{x}) + \mathfrak{g}(\boldsymbol{x})\boldsymbol{u}. \tag{1}$$

In this work, we assume that the communication graph $\mathcal{G} = (\mathcal{I}, E)$ among the N agents is connected and undirected. $(i, j) \in E$ represents that the agents i, j can communicate with each other. The associated Laplacian matrix [14] is denoted as L and the neighborhood set of agent i is defined as $N_i := \{j \in \mathcal{I} : (i, j) \in E\}$. Thus, the stacked locally available state $\boldsymbol{x}_{loc,i} := (\boldsymbol{x}_i^\top, \boldsymbol{x}_{j_1}^\top, ..., \boldsymbol{x}_{j_{|N_i|}}^\top)^\top, j_k \in N_i$, for $k \in \{1, 2, ..., |N_i|\}$, i.e., $\boldsymbol{x}_{loc,i}$ stores the states of agent i and all it neighboring agents $j \in N_i$.

In many applications, the stacked state $\boldsymbol{x}$ needs to be constrained due to safety concerns. Denote the safety set $\mathscr{C}$, where the system state $\boldsymbol{x}$ is expected to evolve within, as a superlevel set of a differentiable function $h : \mathbb{R}^n \to \mathbb{R}$:

$$\mathscr{C} = \{\boldsymbol{x} \in \mathbb{R}^n : h(\boldsymbol{x}) \geq 0\}. \tag{2}$$

Definition 1 (CBF). *Let set $\mathscr{C}$ be defined by* (2). *$h(\boldsymbol{x})$ is a control barrier function (CBF) for the stacked system* (1) *if there exists a locally Lipschitz extended class $\mathcal{K}$ function α such that:*

$$\sup_{\boldsymbol{u}\in\mathbb{R}^m} [L_{\mathfrak{f}}h(\boldsymbol{x}) + L_{\mathfrak{g}}h(\boldsymbol{x})\boldsymbol{u} + \alpha(h(\boldsymbol{x}))] \geq 0, \ \forall \boldsymbol{x} \in \mathbb{R}^n. \tag{3}$$

It has been shown [2]–[4] that any locally Lipschitz control input $\boldsymbol{u}$ that satisfies the CBF constraint (3) renders the set $\mathscr{C}$ forward invariant and, if $\mathscr{C}$ is compact, asymptotically stable.

Assumption 1. *The parameters in the CBF condition* (3) *are locally obtainable, i.e, the argument of* (3) *can be written in the form of*

$$\sum_{i\in\mathcal{I}} \boldsymbol{a}_i^\top(\boldsymbol{x}_{loc,i})\boldsymbol{u}_i + \sum_{i\in\mathcal{I}} b_i(\boldsymbol{x}_{loc,i}) \leq 0, \tag{4}$$

where the functions $\boldsymbol{a}_i(\boldsymbol{x}_{loc,i}), b_i(\boldsymbol{x}_{loc,i})$ can be evaluated based on locally available information to agents.

The constraint form in (4) encodes a variety of constraints including, but not limited to, the examples below:

1) $h(\boldsymbol{x}) = \sum_{i\in\mathcal{I}} h_i(\boldsymbol{x}_i)$ with h_i differentiable. One example in this case is the stacked state boundedness constraint, which is given by $h(\boldsymbol{x}) = \sum_{i\in\mathcal{I}}(r^2 - \|\boldsymbol{x}_i\|^2)$ for some constant $r > 0$.
2) $h(\boldsymbol{x}) = \sum_{l\in L} h_l(\boldsymbol{x}_{l_i}, \boldsymbol{x}_{l_j}), \ L \subset \mathbb{N}$ with $h_l(\cdot, \cdot)$ differentiable, $(l_i, l_j) \in E$. This could encode, for example, a least collective interaction level among all connected agents $h(\boldsymbol{x}) = \sum_{(i,j)\in E}(e^{-r_0\|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2} - r_1)$ for some constants $0 < r_0, 0 < r_1 < 1$.

In the following, only a single safety constraint is considered, and we assume that Assumption 1 holds. Extensions to the multiple safety constraints case will be a future work.

A. Problem formulation

In this work, we assume that nominal controllers are obtained by some distributed coordination protocol, i.e, $\boldsymbol{u}_{nom,i}(\boldsymbol{x}_{loc,i})$. The control barrier function condition (4) serves as the safety constraint in the following quadratic program (QP).

$$\begin{aligned} &\min_{\boldsymbol{u}\in\mathbb{R}^m} \sum_{i\in\mathcal{I}} \frac{1}{2}\|\boldsymbol{u}_i - \boldsymbol{u}_{nom,i}(\boldsymbol{x}_{loc,i})\|^2 \\ &s.t. \ \sum_{i\in\mathcal{I}} \boldsymbol{a}_i^\top(\boldsymbol{x}_{loc,i})\boldsymbol{u}_i + \sum_{i\in\mathcal{I}} b_i(\boldsymbol{x}_{loc,i}) \leq 0. \end{aligned} \tag{5}$$

In the following we denote for brevity $\boldsymbol{a}_i, b_i, \boldsymbol{u}_{nom,i}$ when no ambiguity occurs. Here we note that for agent i, only $\boldsymbol{a}_i, \boldsymbol{u}_{nom,i}, b_i$ is known. The intuition behind this QP is that the control signal is obtained by minimally modifying the nominal controller subject to the safety constraint.

A similar QP formulation has been widely applied in a single agent setting [2]–[4]. However, it remains unanswered how to properly extend this formulation to multi-agent systems since every agent has only local information. In essence, how to design an algorithm that yields the optimal solution to this QP, in a distributed manner, while always enforcing the coupling constraint that certifies system safety remains unsolved. We note that although many distributed optimization algorithms with coupling constraints have been proposed in the literature, few account for the satisfaction of the coupling constraints during the iterations. In the following, we aim at deriving a distributed implementation of the quadratic program in (5) while always satisfying the coupling constraint.

Note that $\boldsymbol{a}_i, b_i, \boldsymbol{u}_{nom,i}$ are defined along the system trajectory, thus their values evolve with time. In the following we start from the analysis of the frozen-time optimality condition and later on provide a scheme that converges to the time-varying optimal solution in finite time while always enforcing the coupling constraint.

III. Main result

In this section, we will analyze the explicit solutions to the QP in (5) and a distributed, yet equivalent QP, and then propose a distributed implementation that solves the original QP online while always enforcing the coupling constraint.

A. Explicit solution analysis

Defining $\bar{\boldsymbol{a}} = (\boldsymbol{a}_1^\top, \boldsymbol{a}_2^\top, ..., \boldsymbol{a}_N^\top)^\top$, $\bar{b} = \sum_{i\in\mathcal{N}} b_i$ and $\boldsymbol{u}_{nom} = (\boldsymbol{u}_{nom,1}^\top, \boldsymbol{u}_{nom,2}^\top, ..., \boldsymbol{u}_{nom,N}^\top)^\top$, we can rewrite the centralized QP in (5) in a compact form as

$$\begin{aligned} &\min_{\boldsymbol{u}\in\mathbb{R}^m} \frac{1}{2}\|\boldsymbol{u}-\boldsymbol{u}_{nom}\|^2 \\ &s.t. \quad \bar{\boldsymbol{a}}^\top \boldsymbol{u} + \bar{b} \le 0. \end{aligned} \tag{6}$$

Assumption 2. *We assume that the QP is feasible, i.e.,* $\bar{b} \le 0$ *whenever* $\bar{\boldsymbol{a}} = \boldsymbol{0}$.

If $\bar{\boldsymbol{a}}^\top \boldsymbol{u}_{nom} + \bar{b} \le 0$, then $\boldsymbol{u}^\star = \boldsymbol{u}_{nom}$ with the optimal cost 0; otherwise, the linear constraint is active, and from Assumption 2, $\bar{\boldsymbol{a}} \neq \boldsymbol{0}$. Based on the least-norm solution, we obtain $\boldsymbol{u}^\star = \boldsymbol{u}_{nom} - (\bar{\boldsymbol{a}}^\top \boldsymbol{u}_{nom} + \bar{b})/\|\bar{\boldsymbol{a}}\|^2 \bar{\boldsymbol{a}}$. Thus, the explicit solution is given as

$$\boldsymbol{u}_i^\star = \boldsymbol{u}_{nom,i} - \mu \boldsymbol{a}_i \tag{7}$$

with μ given by

$$\mu = \begin{cases} 0, & \text{if } \bar{\boldsymbol{a}}^\top \boldsymbol{u}_{nom} + \bar{b} \le 0; \\ (\bar{\boldsymbol{a}}^\top \boldsymbol{u}_{nom} + \bar{b})/\|\bar{\boldsymbol{a}}\|^2, & \textit{Otherwise.} \end{cases} \tag{8}$$

Although $\boldsymbol{u}_{nom,i}$ and $\boldsymbol{a}_i$ in (7) only require local information, the calculation of μ requires global information.

B. An equivalent QP

The QP problem in (5) can be equivalently given by

$$\begin{aligned} &\min_{(\boldsymbol{u},\boldsymbol{y})\in\mathbb{R}^{m+N}} \sum_{i\in\mathcal{I}} \frac{1}{2}\|\boldsymbol{u}_i-\boldsymbol{u}_{nom,i}\|^2 \\ &s.t.\ \boldsymbol{a}_i^\top \boldsymbol{u}_i + \sum_{j\in N_i}(y_i - y_j) + b_i \le 0,\ \forall i \in \mathcal{I}, \end{aligned} \tag{9}$$

where $\boldsymbol{y} = (y_1, y_2, ..., y_N) \in \mathbb{R}^N$ is an auxiliary decision variable. One could relate y_i with agent i and view $(\boldsymbol{x}_i, y_i)$ as an extended state of agent i. The equivalence is shown as follows: for any feasible solution $(\boldsymbol{u}', \boldsymbol{y}')$ to (9), by summing up all the constraints, we obtain that $\boldsymbol{u}'$ also satisfies the constraint in (5); for any feasible solution $\boldsymbol{u}'$ to (5), let $A = \text{blk}(\boldsymbol{a}_1^\top, \boldsymbol{a}_2^\top, ..., \boldsymbol{a}_N^\top), \boldsymbol{b} = (b_1, b_2, ..., b_N)$, and define $\boldsymbol{v} = A\boldsymbol{u}' + \boldsymbol{b}$, $\boldsymbol{w} = \frac{\boldsymbol{1}_N^\top \boldsymbol{v}}{N}\boldsymbol{1}_N$. Thus, $w_i \le 0$ for all $i = 1, 2, ..., N$ and there exists a $\boldsymbol{y}$ such that $L\boldsymbol{y} + \boldsymbol{v} = \boldsymbol{w}$ since $\text{Range}(L) = \{\boldsymbol{x} \in \mathbb{R}^N : \boldsymbol{1}_N^\top \boldsymbol{x} = 0\}$ for a connected undirected graph. This implies that such a $(\boldsymbol{u}', \boldsymbol{y})$ is also feasible to the QP in (9). Thus, they share the same set of feasible solutions with respect to $\boldsymbol{u}$. Noting that they also share the same cost function, we conclude that the two QPs are equivalent.

One nice property of (9) is that, for each constraint, the agent only needs the extended state information from itself and its connected agents. However, the QP in (9) cannot be implemented in a straightforward manner as the following local QPs for each agent i

$$\begin{aligned} &\min_{(\boldsymbol{u}_i,\boldsymbol{y})\in\mathbb{R}^{m_i+N}} \frac{1}{2}\|\boldsymbol{u}_i-\boldsymbol{u}_{nom,i}\|^2 \\ &s.t.\ \boldsymbol{a}_i^\top \boldsymbol{u}_i + \sum_{j\in N_i}(y_i - y_j) + b_i \le 0. \end{aligned}$$

Here $(\boldsymbol{u}_i, \boldsymbol{y})$ is the decision variable for agent i. Note that the $\boldsymbol{y}$s obtained by each agent may not be consistent. In what follows we first analyze the solution to (9), and the distributed implementation is discussed later.

By stacking the constraints in (9) together, we rewrite (9) as

$$\begin{aligned} &\min_{(\boldsymbol{u},\boldsymbol{y})\in\mathbb{R}^{m+N}} \frac{1}{2}\|\boldsymbol{u}-\boldsymbol{u}_{nom}\|^2 \\ &s.t. \quad A\boldsymbol{u} + L\boldsymbol{y} + \boldsymbol{b} \le \boldsymbol{0}. \end{aligned} \tag{10}$$

Here $A = \text{blk}(\boldsymbol{a}_1^\top, \boldsymbol{a}_2^\top, ..., \boldsymbol{a}_N^\top), \boldsymbol{b} = (b_1, b_2, ..., b_N), L$ is the Laplacian matrix. Let $\boldsymbol{l}_i$ be the ith row of the matrix L. The Lagrangian is thus $\mathcal{L}(\boldsymbol{u}, \boldsymbol{y}, \boldsymbol{\lambda}) = \|\boldsymbol{u} - \boldsymbol{u}_{nom}\|^2/2 + \sum_{i\in\mathcal{I}} \lambda_i(\boldsymbol{a}_i^\top \boldsymbol{u}_i + \boldsymbol{l}_i \boldsymbol{y} + b_i)$. Note that this is a convex problem and the Slater's condition holds, so the optimal solution satisfies the Karush–Kuhn–Tucker (KKT) condition, given below

$$A\boldsymbol{u} + L\boldsymbol{y} + \boldsymbol{b} \le \boldsymbol{0} \tag{11}$$

$$\boldsymbol{\lambda} \ge \boldsymbol{0}, \tag{12}$$

$$\lambda_i(\boldsymbol{a}_i^\top \boldsymbol{u}_i + \boldsymbol{l}_i \boldsymbol{y} + b_i) = 0, \quad \forall i \in \mathcal{I}, \tag{13}$$

$$(\frac{\partial \mathcal{L}}{\partial \boldsymbol{u}})^\top = \boldsymbol{u} - \boldsymbol{u}_{nom} + (\lambda_1 \boldsymbol{a}_1, ..., \lambda_N \boldsymbol{a}_N) = \boldsymbol{0}, \tag{14}$$

$$\frac{\partial \mathcal{L}}{\partial \boldsymbol{y}} = \sum_{i\in\mathcal{I}} \lambda_i \boldsymbol{l}_i = \boldsymbol{\lambda}^\top L = \boldsymbol{0}. \tag{15}$$

From (15) and that the communication graph is connected and undirected, we know $\boldsymbol{\lambda} \in \text{Null}(L^\top) = \text{Null}(L) = \{\boldsymbol{v} \in \mathbb{R}^N : \boldsymbol{v} = k\boldsymbol{1}_N, k \in \mathbb{R}\}$. From (12), we further obtain that $\boldsymbol{\lambda} = k\boldsymbol{1}_N$ for some $k \ge 0$. Substituting this to (14), we have

$$\boldsymbol{u}_i = \boldsymbol{u}_{nom,i} - k\boldsymbol{a}_i,\ \forall i \in \mathcal{I}. \tag{16}$$

We now show that $k = \mu$ with μ given in (8) in the following two cases.

1) If $\bar{\boldsymbol{a}}^\top \boldsymbol{u}_{nom} + \bar{b} \le 0$, then there exists a vector $\boldsymbol{w} \in \{\boldsymbol{x} \in \mathbb{R}^N : x_i \le 0, i = 1, 2, ..., N\}$ such that $\sum_{i\in\mathcal{I}} w_i = \bar{\boldsymbol{a}}^\top \boldsymbol{u}_{nom} + \bar{b}$, which gives $\boldsymbol{1}_N^\top(\boldsymbol{w} - (A\boldsymbol{u}_{nom} + \boldsymbol{b})) = 0$. Thus, $\boldsymbol{w} - (A\boldsymbol{u}_{nom} + \boldsymbol{b}) \in \text{Range}(L)$, i.e, there exists a $\boldsymbol{y} \in \mathbb{R}^N$ such that $L\boldsymbol{y} + A\boldsymbol{u}_{nom} + \boldsymbol{b} = \boldsymbol{w} \le \boldsymbol{0}$. By choosing such a $\boldsymbol{y}$, we thus know $(\boldsymbol{u}_{nom}, \boldsymbol{y})$ is a feasible solution to (10) with zero cost, which further implies that it is an optimal solution. Considering that the optimal $\boldsymbol{u}_i, i \in \mathcal{I}$ is given in the form of (16), we obtain $k = 0$.

2) If $\bar{\boldsymbol{a}}^\top \boldsymbol{u}_{nom} + \bar{b} > 0$, we first show that $k > 0$. Assume that $k = 0$, then we have $\boldsymbol{u}_i = \boldsymbol{u}_{nom,i}$. From (11), it implies $\boldsymbol{a}_i^\top \boldsymbol{u}_{nom,i} + \boldsymbol{l}_i \boldsymbol{y} + b_i \le 0, \forall i \in \mathcal{I}$. Summing up over $i \in \mathcal{I}$, we obtain $\bar{\boldsymbol{a}}^\top \boldsymbol{u}_{nom} + \bar{b} \le 0$, which yields a contradiction. Thus $k > 0$. From (13), we have $\boldsymbol{a}_i^\top \boldsymbol{u}_i + \boldsymbol{l}_i \boldsymbol{y} + b_i = 0$. Substituting $\boldsymbol{u}_i = \boldsymbol{u}_{nom,i} - k\boldsymbol{a}_i$, we have $\boldsymbol{a}_i^\top(\boldsymbol{u}_{nom,i} - k\boldsymbol{a}_i) + \boldsymbol{l}_i \boldsymbol{y} + b_i = 0 \Leftrightarrow k\boldsymbol{a}_i^\top \boldsymbol{a}_i = \boldsymbol{a}_i^\top \boldsymbol{u}_{nom,i} + \boldsymbol{l}_i \boldsymbol{y} + b_i$. Summing up over $i \in \mathcal{I}$, we obtain $k\bar{\boldsymbol{a}}^\top \bar{\boldsymbol{a}} = \bar{\boldsymbol{a}}^\top \boldsymbol{u}_{nom} + \bar{b}$, which gives $k = (\bar{\boldsymbol{a}}^\top \boldsymbol{u}_{nom} + \bar{b})/\|\bar{\boldsymbol{a}}\|^2$.

Thus

$$k = \begin{cases} 0, & \text{if } \bar{\boldsymbol{a}}^\top \boldsymbol{u}_{nom} + \bar{b} \le 0; \\ (\bar{\boldsymbol{a}}^\top \boldsymbol{u}_{nom} + \bar{b})/\|\bar{\boldsymbol{a}}\|^2, & \text{Otherwise.} \end{cases} \tag{17}$$

C. Distributed Implementation

In this subsection we propose a distributed implementation scheme that combines an adaptive law that locally updates y_i and a local QP with only the decision variable $\boldsymbol{u}_i$. Specifically, for each agent i, $\forall i \in \mathcal{I}$, we solve the following local QP

$$\begin{aligned} &\min_{\boldsymbol{u}_i \in \mathbb{R}^{m_i}} \frac{1}{2}\|\boldsymbol{u}_i - \boldsymbol{u}_{nom,i}\|^2 \\ &s.t.\ \boldsymbol{a}_i^\top \boldsymbol{u}_i + \sum_{j \in N_i} (y_i - y_j) + b_i \le 0, \end{aligned} \tag{18}$$

and y_i is updated locally according to some adaptation law $\dot{y}_i = v(\boldsymbol{x}_{loc,i}, \boldsymbol{y}_{loc,i})$ that will be derived later. Here $\boldsymbol{y}_{loc,i} = (y_i, y_{j_1}, y_{j_2}, ..., y_{j_{|N_i|}})$, for $j_k \in N_i, k = \{1, 2, ..., |N_i|\}$, i.e., each agent i shares $(\boldsymbol{x}_i, y_i)$ with agent $j \in N_i$.

Proposition 1. *If the local QPs given in* (18) *are feasible, i.e,* $\sum_{j \in N_i}(y_i - y_j) + b_i \le 0$ *whenever* $\boldsymbol{a}_i = \boldsymbol{0}, \forall i \in \mathcal{I}$, *then the solution* $\bar{\boldsymbol{u}}^\star = (\boldsymbol{u}_1^\star, \boldsymbol{u}_2^\star, ..., \boldsymbol{u}_N^\star)$ *to the local QPs satisfies* $\bar{\boldsymbol{a}}^\top \bar{\boldsymbol{u}}^\star + \bar{b} \le 0$ *for any value of* $\boldsymbol{y}$.

Proof. This is evident by summing up all the constraints in the local QPs in (18). □

This property is of interest because it states that whatever $\boldsymbol{y}$ is chosen, the safety guarantee is enforced whenever the local QPs are feasible.

Now we examine the conditions for the optimal $\boldsymbol{y}^\star$ corresponding to (9). Based on our previous analysis (17), we know that if $\bar{\boldsymbol{a}}^\top \boldsymbol{u}_{nom} + \bar{b} \le 0$, then $\boldsymbol{u}_i^\star = \boldsymbol{u}_{nom,i}$, and from (11), $\boldsymbol{a}_i^\top \boldsymbol{u}_{nom,i} + \boldsymbol{l}_i \boldsymbol{y}^\star + b_i \le \boldsymbol{0}$, for all $i \in \mathcal{I}$; if otherwise $\bar{\boldsymbol{a}}^\top \boldsymbol{u}_{nom} + \bar{b} > 0$, then $\boldsymbol{u}_i^\star = \boldsymbol{u}_{nom} - k\boldsymbol{a}_i, k = (\bar{\boldsymbol{a}}^\top \boldsymbol{u}_{nom} + \bar{b})/\|\bar{\boldsymbol{a}}\|^2$; substituting $\boldsymbol{u}_i^\star$ to (11), $\boldsymbol{a}_i^\top \boldsymbol{u}_{nom,i} + \boldsymbol{l}_i \boldsymbol{y}^\star + b_i = k\boldsymbol{a}_i^\top \boldsymbol{a}_i$, for all $i \in \mathcal{I}$. Thus for all $i \in \mathcal{I}$, $\boldsymbol{y}^\star$ needs to satisfy

$$\begin{cases} \boldsymbol{a}_i^\top \boldsymbol{u}_{nom,i} + \boldsymbol{l}_i \boldsymbol{y}^\star + b_i \le \boldsymbol{0}, & \text{if } \bar{\boldsymbol{a}}^\top \boldsymbol{u}_{nom} + \bar{b} \le 0; \\ \boldsymbol{a}_i^\top \boldsymbol{u}_{nom,i} + \boldsymbol{l}_i \boldsymbol{y}^\star + b_i = k\boldsymbol{a}_i^\top \boldsymbol{a}_i, & \text{if } \bar{\boldsymbol{a}}^\top \boldsymbol{u}_{nom} + \bar{b} > 0, \end{cases} \tag{19}$$

with $k = (\bar{\boldsymbol{a}}^\top \boldsymbol{u}_{nom} + \bar{b})/\|\bar{\boldsymbol{a}}\|^2$.

One sufficient condition on $\boldsymbol{y}^\star$ satisfying (19) is

$$\boldsymbol{a}_i^\top \boldsymbol{u}_{nom,i} + \boldsymbol{l}_i \boldsymbol{y}^\star + b_i = c\boldsymbol{a}_i^\top \boldsymbol{a}_i,\ \forall i \in \mathcal{I} \tag{20}$$

with $c = (\bar{\boldsymbol{a}}^\top \boldsymbol{u}_{nom} + \bar{b})/\|\bar{\boldsymbol{a}}\|^2$. Note that (20) can be rewritten in a compact form as $L\boldsymbol{y}^\star = c(\boldsymbol{a}_1^\top \boldsymbol{a}_1, \boldsymbol{a}_2^\top \boldsymbol{a}_2, ..., \boldsymbol{a}_N^\top \boldsymbol{a}_N) - A\boldsymbol{u}_{nom} - \boldsymbol{b}$. This condition on $\boldsymbol{y}^\star$ is feasible since $\text{rank}(L) = n - 1, \text{Range}(L) = \{\boldsymbol{x} : \boldsymbol{1}_N^\top \boldsymbol{x} = 0\}$ and $\boldsymbol{1}_N^\top(c(\boldsymbol{a}_1^\top \boldsymbol{a}_1, \boldsymbol{a}_2^\top \boldsymbol{a}_2, ..., \boldsymbol{a}_N^\top \boldsymbol{a}_N) - A\boldsymbol{u}_{nom} - \boldsymbol{b}) = \sum_{i \in \mathcal{I}}(c\boldsymbol{a}_i^\top \boldsymbol{a}_i - \boldsymbol{a}_i^\top \boldsymbol{u}_{nom,i} - b_i) = 0$. The sufficiency is evident as it poses the same conditions in (19) when $\bar{\boldsymbol{a}}^\top \boldsymbol{u}_{nom} + \bar{b} > 0$ and when $\bar{\boldsymbol{a}}^\top \boldsymbol{u}_{nom} + \bar{b} \le 0$, we have $c \le 0$, which further implies $\boldsymbol{a}_i^\top \boldsymbol{u}_{nom,i} + \boldsymbol{l}_i \boldsymbol{y}^\star + b_i \le 0$.

Assume that $\boldsymbol{a}_i \ne \boldsymbol{0}$ for all $i \in \mathcal{I}$. For a given $\boldsymbol{y}$, we define local variables $c_i, i \in \mathcal{I}$ as

$$c_i = \frac{1}{\boldsymbol{a}_i^\top \boldsymbol{a}_i}(\boldsymbol{l}_i \boldsymbol{y} + \boldsymbol{a}_i^\top \boldsymbol{u}_{nom,i} + b_i). \tag{21}$$

We denote $\boldsymbol{c} = (c_1, c_2, ..., c_N)$.

Proposition 2. *Assume that* $\boldsymbol{a}_i \ne \boldsymbol{0}$ *for all* $i \in \mathcal{I}$. *If* $\boldsymbol{y}$ *is chosen such that* $c_i = c_j$ *for any* $i, j \in \mathcal{I}$, *then the condition in* (19) *is satisfied.*

Proof. Since $\boldsymbol{a}_i \ne \boldsymbol{0}$, and the input set is unbounded, the local QPs given in (18) are feasible. From (21), $\boldsymbol{a}_i^\top \boldsymbol{a}_i c_i = \boldsymbol{l}_i \boldsymbol{y} + \boldsymbol{a}_i^\top \boldsymbol{u}_{nom,i} + b_i$. Noting $c_i = c_j$ and summing up over $\mathcal{I}$, we obtain $c_i = c_j = (\bar{\boldsymbol{a}}^\top \boldsymbol{u}_{nom} + \bar{b})/\|\bar{\boldsymbol{a}}^\top\|^2 = c, \forall i, j \in \mathcal{I}$. Thus, if $\boldsymbol{y}$ is chosen such that $c_i, \forall i \in \mathcal{I}$ reach a consensus, then the condition in (19) is satisfied. □

From (21), we have

$$K_A \boldsymbol{c} = L\boldsymbol{y} + A\boldsymbol{u}_{nom} + \boldsymbol{b} \tag{22}$$

with $K_A := \text{blk}(\|\boldsymbol{a}_1\|^2, \|\boldsymbol{a}_2\|^2, ..., \|\boldsymbol{a}_N\|^2)$. In the following we derive an adaptive law for $\boldsymbol{y}$ such that $\boldsymbol{c}$ reaches a consensus in finite time.

Note that $\boldsymbol{a}_i, \boldsymbol{u}_{nom,i}, \boldsymbol{b}_i$ are state-dependent and thus evolve with time. In the following a finite time consensus law that is inspired by [15] is designed and analyzed.

Proposition 3. *Assume that* $\boldsymbol{a}_i, \boldsymbol{u}_{nom,i}, \boldsymbol{b}_i$ *are slowly time-varying in the sense that* $\|\dot{K}_A \boldsymbol{c}(t) + \frac{d}{dt}(A\boldsymbol{u}_{nom} + \boldsymbol{b})\|_1 \le D$ *for some* $D > 0$ *and* $a_{min} \le \boldsymbol{a}_i^\top \boldsymbol{a}_i \le a_{max}$ *for some positive constants* a_{min}, a_{max} *for all* $i \in \mathcal{I}$. *If the discontinuous adaptive law*

$$\dot{\boldsymbol{y}} = -k_0 \text{sign}(L\boldsymbol{c}), \tag{23}$$

is applied, where $\boldsymbol{c}$ *is defined in* (21) *and the gain* k_0 *satisfies*

$$k_0 \ge a_{max}(2\delta_{max} D / a_{min} + \epsilon), \tag{24}$$

where $\delta_{max} := \max_{i \in \mathcal{I}} |N_i|$, ϵ *is a positive constant, then* $\boldsymbol{c}$ *achieves a consensus, i.e.,* $c_i = c_j, \forall i, j \in \mathcal{I}$, *within a finite time* $t_r \le \frac{\|L\boldsymbol{c}(0)\|_1}{\epsilon}$.

Proof. Differentiating (22), we obtain $\dot{\boldsymbol{c}} = K_A^{-1}(L\dot{\boldsymbol{y}} + \frac{d}{dt}(A\boldsymbol{u}_{nom} + b) + \dot{K}_A \boldsymbol{c})$. For simplicity, denote $d := \frac{d}{dt}(A\boldsymbol{u}_{nom} + b) + \dot{K}_A \boldsymbol{c}$. Substituting (23), we have

$$\dot{\boldsymbol{c}} = K_A^{-1}(-k_0 L \text{sign}(L\boldsymbol{c}) + d). \tag{25}$$

In this proof we will interpret the solution to this differential equation in the sense of Filippov and use tools from non-smooth analysis. The existence of such solution is guaranteed by the boundedness of the right hand side. For more details about the non-smooth analysis, see [15], [16].

Define the disagreement vector $\boldsymbol{\delta} = \boldsymbol{c} - \frac{1}{N}\boldsymbol{1}_N \boldsymbol{1}_N^\top \boldsymbol{c}$, $\boldsymbol{z} = L\boldsymbol{\delta}$, and the function $\sigma : \boldsymbol{x} \mapsto \text{sign}(L\boldsymbol{x})$. The following properties hold:

1) $\sigma(\boldsymbol{c}) = \text{sign}(L\boldsymbol{c}) = \text{sign}(L\boldsymbol{\delta}) = \sigma(\boldsymbol{\delta})$;
2) If $\boldsymbol{c}$ has not reached a consensus, then $\|L\sigma(\boldsymbol{c})\| \ge 1$.

Property 2 can be verified in a similar manner as [15, Prop. 2.1] and omitted here for brevity. Consider the locally Lipschitz Lyapunov function $V(\boldsymbol{z}) = \|\boldsymbol{z}\|_1$. For $z_1 \ne 0, z_2 \ne 0, ..., z_N \ne 0$, the time derivative is calculated by $\frac{d}{dt}V(\boldsymbol{z}) = \sum_{i \in \mathcal{I}} \text{sign}(z_i)\dot{z}_i = \text{sign}(\boldsymbol{z})^\top \dot{\boldsymbol{z}}$. If any of the variables z_i is zero, let $\mathcal{I}^0 = \{i \in \mathcal{I} : z_i = 0\}$ and $\mathcal{I}^{\neq} = \{i \in \mathcal{I} : z_i \neq 0\}$. We can compute the generalized time derivative as

$$\frac{d}{dt}V(\boldsymbol{z}) \in \sum_{i\in\mathcal{I}^{\neq}} \text{sign}(z_i)\dot{z}_i + \sum_{i\in\mathcal{I}^0} \text{SIGN}(z_i)\dot{z}_i,$$

where $\text{SIGN}(z) := \begin{cases} 1, & z>0; \\ [-1,1], & z=0; \\ -1, & z<0. \end{cases}$ Since we are dealing with Filippov solutions, we can disregard the case which $z_i = 0$ holds for isolated time instants of measure zero. If $z_i = 0$ holds along an interval of time of positive measure, then, in the sense of Filippov, $\dot{z}_i$ exists at those time instants and $\dot{z}_i = 0$. Based on this fact, we have that for almost all t, $\frac{d}{dt}V(\boldsymbol{z}) = \sum_{i\in\mathcal{I}^{\neq}} \text{sign}(z_i)\dot{z}_i + \sum_{i\in\mathcal{I}^0} \text{sign}(z_i)\dot{z}_i$, Thus the generalized time derivative $\frac{d}{dt}V(\boldsymbol{z}(t))$ is now a well-defined function for almost all t and is given by

$$\frac{d}{dt}V(\boldsymbol{z}) = \sum_{i\in\mathcal{I}} \text{sign}(z_i)\dot{z}_i = \sigma(\boldsymbol{\delta})^\top \dot{\boldsymbol{z}}. \tag{26}$$

In view of Property 1 and the fact that $\dot{\boldsymbol{z}} = L\dot{\boldsymbol{\delta}} = L\dot{\boldsymbol{c}} - \frac{1}{N}L\mathbf{1}_N\mathbf{1}_N^\top\dot{\boldsymbol{c}} = L\dot{\boldsymbol{c}}$, we get $\frac{d}{dt}V(\boldsymbol{z}) = \sigma(\boldsymbol{c})^\top L\dot{\boldsymbol{c}}$. Thus from (25),

$$\frac{d}{dt}V(\boldsymbol{z}) = \sigma(\boldsymbol{c})^\top LK_A^{-1}(-k_0 L\sigma(\boldsymbol{c}) + d) \tag{27}$$

Note that $\|L\|_1 = max_{j\in\mathcal{I}} \sum_{i\in\mathcal{I}} |l_{ij}| = 2\delta_{max}$ since the graph $\mathcal{G}$ is connected and undirected, $\|K_A^{-1}\|_1 = 1/a_{min}$, $\|d\|_1 \leq D$, we have

$$\begin{aligned} |\sigma(\boldsymbol{c})^\top LK_A^{-1}d| \leq \|LK_A^{-1}d\|_1 &\leq \|L\|_1\|K_A^{-1}\|_1\|d\|_1 \\ &\leq 2\delta_{max}D/a_{min}. \end{aligned} \tag{28}$$

When $\boldsymbol{c}$ has not reached a consensus, we note that

$$-k_0(L\sigma(\boldsymbol{c}))^\top K_A^{-1}L\sigma(\boldsymbol{c}) \leq -k_0/a_{max} \tag{29}$$

in view of Property 2. Substituting (24), (28) and (29) into (27), we have $\frac{d}{dt}V(\boldsymbol{z}) \leq -\epsilon < 0$. When $\boldsymbol{c}$ reaches a consensus, i.e, $\boldsymbol{c} = \alpha\mathbf{1}_N, \alpha \in \mathbb{R}$, we get $\boldsymbol{z} = L\boldsymbol{\delta} = 0, V(\boldsymbol{z}) = 0$. Summarizing, it holds

$$\begin{aligned} &V(\boldsymbol{z}) > 0 \text{ and } \frac{d}{dt}V(\boldsymbol{z}(t)) \leq -\epsilon, \text{ for } \boldsymbol{c} \neq \alpha\mathbf{1}_N, \alpha \in \mathbb{R}; \\ &V(\boldsymbol{z}) = 0 \text{ and } \frac{d}{dt}V(\boldsymbol{z}(t)) = 0, \quad \text{for } \boldsymbol{c} = \alpha\mathbf{1}_N, \alpha \in \mathbb{R}. \end{aligned} \tag{30}$$

Therefore $\boldsymbol{c}(t)$ achieves consensus in finite time, with the transient time $t_r \leq V(\boldsymbol{z}(0))/\epsilon = \|L\boldsymbol{c}(0)\|_1/\epsilon$. □

Now we summarize our main results in the following theorem.

Theorem 1. *Consider the CBF-induced quadratic program in* (5). *Assume that the conditions in Proposition 3 hold. Then the solution to the local QPs in* (18) *with* $\boldsymbol{y}$ *locally updated according to* (23) *is identical to the solution to the QP* (5) *in finite time. Moreover, the coupling constraint in* (5) *is satisfied for all time.*

Proof. In view of the fact that $\boldsymbol{a}_i^\top\boldsymbol{a}_i \geq a_{min}$, and that the input set is unbounded, the local QPs given in (18) are feasible. Applying Proposition 1, we know that the coupling constraint is thus satisfied for all time. From Proposition 3, the variable $\boldsymbol{c}$ defined in (21) achieves a consensus in finite time. Since $\boldsymbol{a}_i^\top\boldsymbol{a}_i \geq a_{min}, \forall i \in \mathcal{I}$, Proposition 2 is applicable and $\boldsymbol{y}$ converges to the optimal $\boldsymbol{y}^\star$ in finite time. Thus, the local QPs in (18) solve the original QP problem in finite time. □

IV. Simulations

In this section we will demonstrate our results for a multi-agent system with $N = 9$ agents. The communication graph $(\mathcal{I}, E)$ is defined and depicted in Fig.1. Firstly we consider a static distributed QP problem in the form of (5), where $(\boldsymbol{a}_1^\top, \boldsymbol{a}_2^\top, ..., \boldsymbol{a}_9^\top) = (0.745, -1.03, 0.004, 0.509, 0.987, -2.70, 2.405, 2.164, 0.426, 0.347, 2.130, 1.439, -2.467, -0.633, 0.889, -2.482, -0.425, 0.879)$, $(b_1, b_2, ..., b_9) = (0.242, 0.839, -1.91, 2.082, 2.972, -2.639, 0.141, -0.677, 1.198)$, and $\boldsymbol{u}_{nom,i} = (0, 0)$ are constants. Here agent $i \in \mathcal{I}$ solves the local QP (18) and adapts its y_i value according to (23) with $k_0 = 10$, $y_i(0) = 0$. The results are shown in Fig. 2. In this case the assumptions in Theorem 1 hold and we see that 1) $\boldsymbol{c}$ reaches a consensus (and thus the optimal solution to this static QP is obtained) in finite time; 2) the coupling constraint is always enforced.

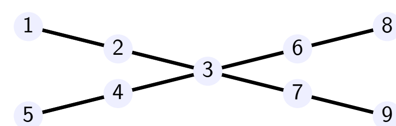
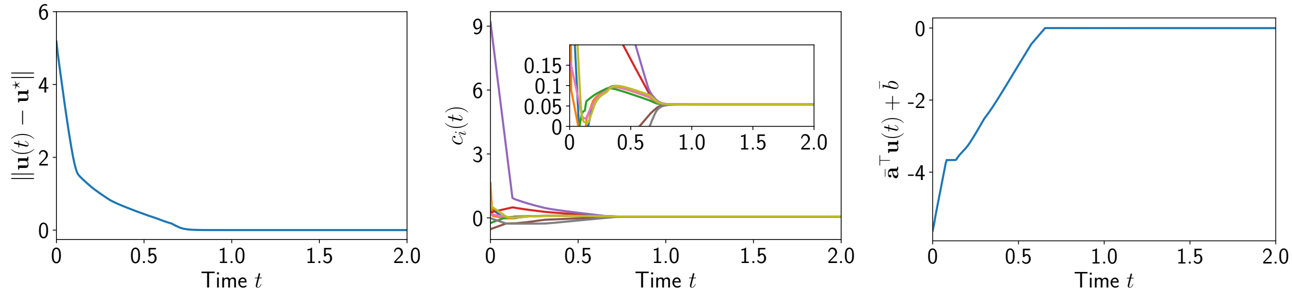


Fig. 1: Communication topology of the multi-agent system.

Now we apply our results to the same multi-agent system for a consensus control problem with a stacked state boundedness constraint given below. For $i \in \mathcal{I}, \boldsymbol{x}_i \in \mathbb{R}^2$ denotes the agent state, the agent dynamics is $\dot{\boldsymbol{x}}_i = \boldsymbol{u}_i$, and its initial condition $\boldsymbol{x}_i(0) = (2\cos(2\pi i/10) + 2, 2\sin(2\pi i/10) + 1)$, for $i \in \mathcal{I}$. Define a nominal controller $\boldsymbol{u}_{nom,i} = \sum_{j\in N_i}(\boldsymbol{x}_j - \boldsymbol{x}_i)$. It is known [14] that the multi-agent system with this controller achieves a consensus with each agent converging to $(2, 1)$, which is later verified in the simulation. In this example, we specify that the agents should meet in $\{\boldsymbol{x} \in \mathbb{R}^{18} : h(\boldsymbol{x}) = 9 - \boldsymbol{x}^\top\boldsymbol{x} \geq 0\}$. With $\alpha(v) = v, v \in \mathbb{R}$, the induced CBF condition in the form of (4) is thus $\sum_{i\in\mathcal{I}} 2\boldsymbol{x}_i^\top\boldsymbol{u}_i + \sum_{i\in\mathcal{I}}(\boldsymbol{x}_i^\top\boldsymbol{x}_i - 1) \leq 0$. Note that the constraint is initially violated $h(\boldsymbol{x}(0)) = -72.0$.

We test the following 3 controllers for this task: Case 1 the nominal control $\boldsymbol{u}_{nom,i}$; Case 2 the CBF-induced QP (5) solved in a centralized manner; Case 3 the proposed distributed method (18),(23) with $\boldsymbol{y}(0) = \mathbf{0}$ and $k_0 = 100$. The simulation results are shown in Fig. 3. We see that 1) the final state in Case 1 (Fig. 3(a)) does not fulfill the stacked state boundedness constraint; 2) the trajectories in Case 3 are bounded away from $(0, 0)$, thus $\boldsymbol{a}_i(t) = 2\boldsymbol{x}_i(t)$ is also bounded away from $(0, 0)$; 3) although the trajectories in Case 3 are in general more curly than that of Case 2, the final states in Case 2 and Case 3 are very close. Setting the simulation duration to 20s, we get $h(\boldsymbol{x}_{f,2}) = 0.0014, h(\boldsymbol{x}_{f,3}) = -0.00047$, where $\boldsymbol{x}_{f,2}, \boldsymbol{x}_{f,3}$ are the final states in Cases 2 and 3, respectively. The negativeness of $h(\boldsymbol{x}_{f,3})$ could be caused by the smooth approximation of sign function, or that the agent states are still evolving.

We acknowledge that although the proposed scheme works well in the simulations, the condition on the lower bound of

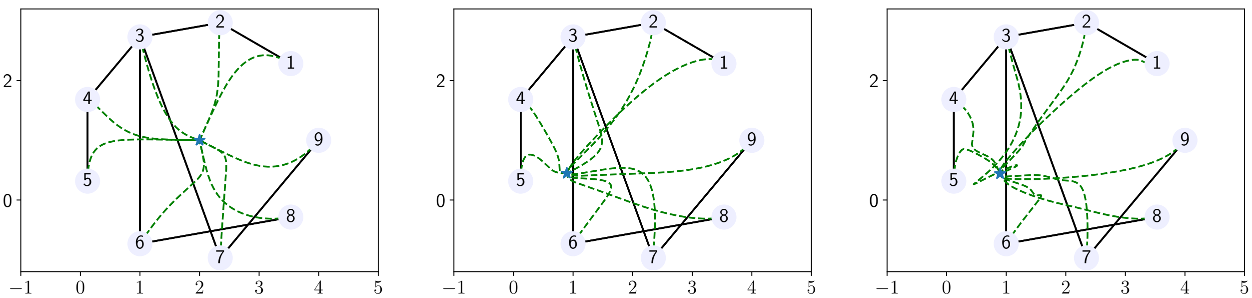


(a) Time evolution of $\|\boldsymbol{u}(t) - \boldsymbol{u}^\star\|$.

(b) Time evolution of $c_i(t)$

(c) Time evolution of $\sum_{i\in\mathcal{I}}(\boldsymbol{a}_i^\top \boldsymbol{u}_i(t) + b_i)$

Fig. 2: Numerical results involving 9 agents solving the static QP with a coupling constraint.



(a) Case 1: nominal controller.

(b) Case 2: centralized CBF controller.

(c) Case 3: distributed CBF controller.

Fig. 3: System trajectories of a multi-agent system under three different controllers. The black lines denote the communication links among the agents, and the green dash lines are the state trajectories of each agent; the blue star is the final state.

$\boldsymbol{a}_i^\top \boldsymbol{a}_i, i \in \mathcal{I}$ may be conservative and, if $\boldsymbol{a}_i^\top \boldsymbol{a}_i$ is too small, the division by it (when obtaining c_i as in (21)) may cause numerical issues. As a future work, we plan to remove the boundedness assumptions on $\boldsymbol{a}_i$. Another direction for future work is to extend the result to the multiple safety constraints case, where more general safety constraints can be tackled.

V. Conclusion

In this work, we proposed a distributed implementation scheme for CBF-induced quadratic programs for multi-agent systems, where each agent solves a local QP and locally adapts an auxiliary variable. Under the assumption that the parameters of the coupling constraint are slowly time-varying, the proposed implementation solves the CBF-induced QP in finite time and guarantees the satisfaction of the coupling constraint for all time. These two properties are of interest because they guarantee optimality of the control signal to the CBF-induced QP and safety of the multi-agent system. We also applied our results in a static QP problem and a consensus control problem with a stacked state boundedness constraint.

References

[1] P. Wieland and F. Allgöwer, "Constructive safety using control barrier functions," *IFAC Proceedings*, vol. 40, no. 12, pp. 462–467, 2007.

[2] X. Xu, P. Tabuada, J. W. Grizzle, and A. D. Ames, "Robustness of control barrier functions for safety critical control," in *Proc. IFAC Conf. Anal. Design Hybrid Syst.*, vol. 48, 2015, pp. 54–61.

[3] 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*, vol. 62, no. 8, pp. 3861–3876, 2016.

[4] X. Tan, W. Shaw Cortez, and D. V. Dimarogonas, "High-order barrier functions: robustness, safety and performance-critical control," *IEEE Transactions on Automatic Control*, Early access, 2021.

[5] L. Wang, A. Ames, and M. Egerstedt, "Safety barrier certificates for collisions-free multirobot systems," *IEEE Transactions on Robotics*, vol. 33, no. 3, pp. 661–674, 2017.

[6] P. Glotfelter, J. Cortés, and M. Egerstedt, "Nonsmooth barrier functions with applications to multi-robot systems," *IEEE Control Systems Letters*, vol. 1, no. 2, pp. 310–315, 2017.

[7] D. Panagou, D. M. Stipanović, and P. G. Voulgaris, "Distributed coordination control for multi-robot networks using lyapunov-like barrier functions," *IEEE Transactions on Automatic Control*, vol. 61, no. 3, pp. 617–632, 2015.

[8] B. Capelli and L. Sabattini, "Connectivity maintenance: Global and optimized approach through control barrier functions," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 5590–5596.

[9] L. Lindemann and D. V. Dimarogonas, "Barrier function based collaborative control of multiple robots under signal temporal logic tasks," *IEEE Transactions on Control of Network Systems*, vol. 7, no. 4, pp. 1916–1928, 2020.

[10] Y. Chen, M. Santillo, M. Jankovic, and A. D. Ames, "Online decentralized decision making with inequality constraints: an ADMM approach," *IEEE Control Systems Letters*, vol. 5, no. 6, pp. 2156–2161, 2020.

[11] A. Falsone, K. Margellos, S. Garatti, and M. Prandini, "Dual decomposition for multi-agent distributed optimization with coupling constraints," *Automatica*, vol. 84, pp. 149–158, 2017.

[12] M. Santilli, G. Oliva, and A. Gasparri, "Distributed finite-time algorithm for a class of quadratic optimization problems with time-varying linear constraints," in *2020 59th IEEE Conference on Decision and Control (CDC)*. IEEE, 2020, pp. 4380–4386.

[13] M. Manao and I. Masumi, "Consensus-based control barrier function for swarm," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2021.

[14] M. Mesbahi and M. Egerstedt, *Graph theoretic methods in multiagent networks*. Princeton University Press, 2010.

[15] M. Franceschelli, A. Pisano, A. Giua, and E. Usai, "Finite-time consensus with disturbance rejection by discontinuous local interactions in directed graphs," *IEEE transactions on Automatic Control*, vol. 60, no. 4, pp. 1133–1138, 2014.

[16] D. Shevitz and B. Paden, "Lyapunov stability theory of nonsmooth systems," *IEEE Transactions on automatic control*, vol. 39, no. 9, pp. 1910–1914, 1994.