Safety Barrier Certificates for Collisions-Free Multirobot Systems

Li Wang, *Student Member, IEEE*, Aaron D. Ames, *Member, IEEE*, and Magnus Egerstedt, *Fellow, IEEE*

***Abstract*—This paper presents safety barrier certificates that ensure scalable and provably collision-free behaviors in multirobot systems by modifying the nominal controllers to formally satisfy safety constraints. This is achieved by minimizing the difference between the actual and the nominal controllers subject to safety constraints. The resulting computation of the safety controllers is done through a quadratic programming problem that can be solved in real-time and in this paper, we describe a series of problems of increasing complexity. Starting with a centralized formulation, where the safety controller is computed across all agents simultaneously, we show how one can achieve a natural decentralization whereby individual robots only have to remain safe relative to nearby robots. Conservativeness and existence of solutions as well as deadlock-avoidance are then addressed using a mixture of relaxed control barrier functions, hybrid braking controllers, and consistent perturbations. The resulting control strategy is verified experimentally on a collection of wheeled mobile robots whose nominal controllers are explicitly designed to make the robots collide.**

***Index Terms*—Barrier certificates, collision avoidance, control barrier function, multirobot systems.**

I. Introduction

THE design of multirobot coordination strategies is typically concerned with realizing primary, global behaviors such as achieving and maintaining formations, covering areas of interest, environmental exploration, and boundary tracking, see, e.g., [1]–[5] and references therein. Collision avoidance and other safety considerations are then added as secondary objectives, resulting in a hierarchical composition of multiple objectives, e.g., [6]. Thus, what is ultimately deployed on the system is a combination of a "formally" designed nominal controller together with a "hand-crafted" collision avoidance algorithm, as discussed in [4], [5], [7], with the basic idea being that the multirobot system executes the nominal, primary controller most of the time, while the secondary, collision avoidance controller takes over when the robots are too close to each other, e.g., following the behavior-based design paradigm [6]. This approach is typically successful when the nominal controller, e.g., for covering an area or assembling and maintaining a formation, is allowed to dominate most of the time. However, as the robot "density" increases, e.g., when the number of agents increases, the collision avoidance strategy starts to dominate the behavior, with the robots spending most of the time avoiding collisions and, as a result, they do not progress toward achieving the primary objectives, e.g., [8].

The reason to use cbf

A remedy to this problem is to design a collision avoidance controller that is minimally invasive in the sense that it only modifies the nominal controller when collision is truly imminent. This idea was pursued in [9] for pairs of aircraft, where the aircraft switches between normal operation modes and evasive maneuvers that were guaranteed to be safe no matter what actions the other aircraft were taking. Technically, this becomes a differential game problem and the computational cost associated with solving the corresponding Hamilton–Jacobi–Bellman–Isaacs Equation can be prohibitive, even for two agents and, as such, does not lend itself to real-time applications. A similar idea was pursued in [10], [11], where a so-called "velocity obstacle" method was used to calculate the optimal, safe velocity. Although computationally cheap, the underlying, infinite acceleration and constant velocity assumptions make it not suitable for highly dynamical collision avoidance behaviors. A mixed-integer quadratic program approach was proposed in [12] to plan robot trajectories that avoid both inter-robot collisions and static obstacles. But the computational complexity of this approach prohibits its application to large swarms of robots.

In this paper, we follow this line of inquiry, whereby the nominal controller is allowed to influence the system as much as possible. However, as the robots are about to collide, barrier certificates, e.g., [13], and [14], are employed to ensure that the collisions are avoided. Barrier certificates were used for safety verification of dynamical systems [13]. The existence of barrier certificates for safe dynamical systems was established in [15], and [16]. Romdlony and Jayawardhana [17] merged control barrier function with control Lyaponov function for safe system stabilization. However, these barrier certificates are often overly restrictive by requiring the barrier function to always descend. A more permissive control barrier function was proposed in [14], [18], and [19] with correctness guarantees. In particular, in this paper, we use this type of more permissive control barrier functions to synthesize safety certificates—these provide the minimum modification necessary to formally guarantee safety. In fact, barrier certificates for multirobot collision avoidance

Manuscript received April 5, 2016; revised October 28, 2016; accepted January 11, 2017. Date of publication February 15, 2017; date of current version June 5, 2017. This paper was recommended for publication by Associate Editor H. Kress-Gazit and Editor T. Murphey upon evaluation of the reviewers' comments. The work of L. Wang and M. Egerstedt was supported by the U.S. National Science Foundation under Grant 1544332 and the work of A. D. Ames was supported by the U.S. National Science Foundation under Grant 1239055.

L. Wang and M. Egerstedt are with the School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA 30332 USA (e-mail: liwang@gatech.edu; magnus@gatech.edu).

A. D. Ames is with the Department of Mechanical and Civil Engineering, California Institute of Technology, Pasadena, CA 91125 USA (e-mail: ames@caltech.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TRO.2017.2659727

1552-3098 © 2017 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

were introduced in [20], and [21] and was extended to heterogeneous multirobot systems in [22]. The results in this paper extend and generalize these previous ideas by supporting a nonconservative decentralization of the certificates, guaranteeing that safe controllers do indeed exist by establishing feasibility, and providing approaches for deadlock avoidance.

In particular, in this paper, a safe set is defined over the joint state space of the multirobot team modeled as double integrators, which yields barrier certificates, i.e., control barrier functions, that are used to ensure that the robots remain in this set for all time. As a result, if a controller satisfies the safety barrier certificates, it is safe. At the same time, the actual control action is taken to be as close as possible to the nominal control action, i.e., it should be minimally invasive. However, the admissible control space permitted by the safety barrier certificates might be small or even empty. Therefore, there is a need to expand the admissible control space so that the nominal controller has the authority to perform desired control actions. The proposed barrier certificates have augmented admissible control space by using the so-called relaxed zeroing control barrier functions (ZCBF) as compared to past certificates. Additionally, the proposed barrier certificates utilize ZCBFs (as opposed to past schemes which used reciprocal barrier functions), with the result being both invariance and robust stabilization of the safe set.

The second main contribution in this paper is the establishment of feasibility of the underlying QP problem by combining the ZCBF controllers with an emergency braking maneuver. More specifically, multiple safety barrier constraints are combined as inequality constraints in the QP. If the QP is feasible, it guarantees that all barrier functions can be satisfied. In the case of infeasibility, we propose a method that ensures safety: Emergency braking. It will be shown that the emergency braking maneuver always constitutes a feasible action to avoid collisions and, as such, the feasible set is nonempty.

Finally, as the safety barrier constraints are designed to be decentralized and use only local sensing information, the lack of a central coordination signal might lead to deadlock among multiple robots with conflicting primary objectives. In fact, the potential for deadlock behavior is well-established, e.g., [23]–[25], whereby the robots no longer progress toward the realization of the primary objective. The solution in this paper is based on a novel deadlock-detection scheme in combination with a consistent perturbation method that is inspired by symmetry-breaking traffic rules.

The decentralized safety barrier certificates for collision avoidance were implemented experimentally on a multirobot system consisting of multiple differential-drive mobile robots. The nominal controllers, designed to make agents swap positions or to execute leader-follower maneuvers, were explicitly designed to cause collisions in confined workspace. With the decentralized safety barrier certificates, all robots successfully achieved their goals, and avoided potential collisions while staying close to the nominal controllers. The remainder of this paper is organized as follows: In Section II, we briefly recall the ZCBF construction from [14], [26]. In Section III, the safety barrier certificates are first formulated in a centralized manner in order to call out what the information requirements really are to make safety barrier certificates effective. A decentralized formulation is proposed by assigning the admissible control space to different agents in Section IV followed by several modifications, namely enlargement of the admissible control space of the safety barrier certificates by introducing a Relaxed ZCBF in Section V, feasibility for safety critical system in Section VI, and deadlock detection and resolution in Section VII. In Section VIII, the safety barrier certificates are validated on an actual multirobot system, where the results show strong agreement with the simulation. A summary of the work is presented in Section IX.

II. Background: Control Barrier Functions

In this section, the fundamentals for ZCBF are briefly recalled. The basic idea is to define a set of safe states and then use the ZCBF to formally guarantee the forward invariance of the desired set, i.e., if the system starts in the safe set, it stays in the safe set [18], [26]. Conceptually, ZCBFs are similar to control Lyapunov functions in that they both can ensure certain properties of the system without explicitly calculating the forward reachable set.

We first review some of the basic ideas behind ZCBFs and then retool them to ensure that teams of robots that start out collision-free will indeed remain collision-free. The main robot model under consideration in this paper is the double integrator. However, for the sake of generality, we first consider dynamical systems on control affine form

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

where the state $x \in \mathbb{R}^n$ and control $u \in U \subset \mathbb{R}^m$, f and g are locally Lipschitz continuous. The system (1) is assumed to be forward complete for simplicity, i.e., $x(t)$ is defined $\forall t \geq 0$. Note that the state x is taken to be an aggregate of all the robot states in the team and f and g encode the collective robot dynamics—these entities will be made concrete in subsequent sections when we specialize the system to the particular multirobot system under investigation.

Let the set $\mathscr{C} \subset \mathbb{R}^n$ be the safe set, i.e., the set where we wish that the aggregate robot states should stay. The task is then to design a controller that guarantees the forward invariance of set $\mathscr{C}$, i.e., if $x(0) \in \mathscr{C}$, then $x(t) \in \mathscr{C}$, $\forall t \geq 0$. And, to arrive at a mathematically manipulable formulation, we encode $\mathscr{C}$ through the super-level set of a ZCBF candidate function $h : \mathbb{R}^n \to \mathbb{R}$

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

We note that the time derivative of $h(x)$ along the state trajectories is

$$\frac{dh(x)}{dt} = \frac{\partial h(x)}{\partial x}\dot{x} = \frac{\partial h(x)}{\partial x}(f(x) + g(x)u)$$

or, using the Lie derivative formalism

$$\frac{dh(x)}{dt} = L_f h(x) + L_g h(x)u.$$

Definition II.1: A continuous function $\boldsymbol{\kappa} : (-b, a) \to \mathbb{R}$ for some $a, b > 0$ is called an extended class-$\mathscr{K}$ function if it is strictly increasing and $\boldsymbol{\kappa}(0) = 0$.

Definition II.2: Given a dynamical system (1) and a set $\mathscr{C} \subset \mathbb{R}^n$ defined by (2) for a smooth function $h : \mathscr{D} \to \mathbb{R}$, with $\mathscr{C} \subseteq \mathscr{D} \subset \mathbb{R}^n$. The function h is called a ZCBF, if there exists an extended class-$\mathscr{K}$ function $\boldsymbol{\kappa}$ such that

$$\sup_{u\in U}\{L_f h(x) + L_g h(x)u + \boldsymbol{\kappa}(h(x)) \geq 0\}$$

for all $x \in \mathscr{D}$.

Given a ZCBF, the admissible control space $S(x)$ can be defined as

$$S(x)=\{u \in U \mid L_f h(x) + L_g h(x)u + \boldsymbol{\kappa}(h(x)) \geq 0\}\,,\ x \in \mathscr{D}$$

Now, we need to guarantee that $\mathscr{C}$ is forward invariant, and this is ensured by the following theorem.

Theorem [26]: Given a set $\mathscr{C} \subset \mathbb{R}^n$ defined by (2) and a ZCBF h defined on $\mathscr{D}$, with $\mathscr{C} \subseteq \mathscr{D} \subset \mathbb{R}^n$, any Lipschitz continuous controller $u : \mathscr{D} \to \mathbb{R}$ such that $u \in S(x)$ for the system (1) renders the set $\mathscr{C}$ forward invariant. And $\mathscr{C}$ is asymptotically stable in $\mathscr{D}$.

The extended class-$\mathscr{K}$ function $\boldsymbol{\kappa}$ regulates how fast the state of the system can approach the boundary of $\mathscr{C}$. Therefore, different choices of $\boldsymbol{\kappa}$, e.g., $\boldsymbol{\kappa}(r) = r^P$ for any positive odd integer P, lead to different behaviors near the boundary. In this paper, the particular choice of $\boldsymbol{\kappa}(h(x)) = \gamma h^3(x)$ with $\gamma > 0$ will be adopted, which means that the controller needs to satisfy

$$L_f h(x) + L_g h(x)u + \gamma h^3(x) \geq 0 \tag{3}$$

to render the set $\mathscr{C}$ forward invariant. Note that this form of the barrier function can equivalently be stated as a *reciprocal barrier function* of the form

$$B(x) = \frac{1}{h(x)}$$

which was used in previous work [20].

III. Centralized Safety Barrier Certificates

The key to being able to ensure that the robots avoid collisions is that *all* potential, pairwise robot-to-robot collisions are accounted for. As such, a natural first attempt is to let a centralized computation keep track of all robot pairs and then dictate how the nominal controllers should be modified in order to avoid collisions. In this section, we will see how such a centralized computation should be structured. The subsequent sections will be devoted to the relaxation of the need to consider all pairs to a significantly smaller subset of pairs, and to the decentralization of the computation in order to allow the robots themselves make decisions in real-time (see Section IV).

Concretely, consider a multirobot system consisting of N planar, mobile robots, indexed by $\mathscr{M} = \{i \mid i = 1, 2, \ldots, N\}$. As the acceleration limitations play a crucial role when avoiding collisions (otherwise, the robots could just set their velocities to zero instantaneously to avoid collisions), we choose to model the robot dynamics as double integrators

$$\begin{bmatrix}\dot{\mathbf{p}}_i\\ \dot{\mathbf{v}}_i\end{bmatrix} = \begin{bmatrix}0 & I_{2\times 2}\\ 0 & 0\end{bmatrix}\begin{bmatrix}\mathbf{p}_i\\ \mathbf{v}_i\end{bmatrix} + \begin{bmatrix}0\\ I_{2\times 2}\end{bmatrix}\mathbf{u}_i \tag{4}$$

where $\mathbf{p}_i \in \mathbb{R}^2$, $\mathbf{v}_i \in \mathbb{R}^2$, and $\mathbf{u}_i \in \mathbb{R}^2$ represent the positions, velocities, and inputs (acceleration commands) of agent i, respectively. The velocity and acceleration of agent i are limited by $\|\mathbf{v}_i\|_\infty \leq \beta_i$ and $\|\mathbf{u}_i\|_\infty \leq \alpha_i$. The aggregate states and inputs of all N agents are denoted as $(\mathbf{p}, \mathbf{v}) \in \mathbb{R}^{4N}$ and $\mathbf{u} \in \mathbb{R}^{2N}$.

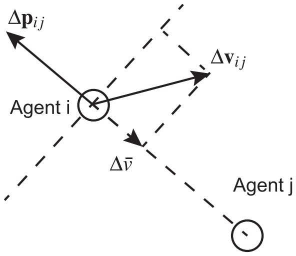


Fig. 1. Relative position and velocity between two agents.

Next, a pairwise robot-to-robot safety constraint is formulated to guarantee that a safety distance D_s between any two agents can be ensured. Algorithms to avoid imminent collision with static obstacles were developed in [27] and [28] by decelerating the agent to zero velocity with the maximum braking force. However, to avoid imminent collision with a moving agent, the relative velocity between two agents needs to be reduced to zero instead of the absolute velocity. Consider any two agents i and j, the relative position and relative velocity between them are $\Delta\mathbf{p}_{ij} = \mathbf{p}_i - \mathbf{p}_j$ and $\Delta\mathbf{v}_{ij} = \mathbf{v}_i - \mathbf{v}_j$. As illustrated in Fig. 1, the normal component of the relative velocity ($\Delta\bar{v} = \|\Delta\dot{\mathbf{p}}_{ij}\| = \frac{\Delta\mathbf{p}_{ij}^T}{\|\Delta\mathbf{p}_{ij}\|}\Delta\mathbf{v}_{ij}$) is the actual component that might lead to collision between agents i and j, while the tangent component of $\Delta\mathbf{v}_{ij}$ only leads to rotation around each other. Therefore, we need to regulate $\Delta\bar{v}$ so that imminent collisions can be avoided if the maximum relative braking force is applied.

Assuming the normal component of the relative velocity between agents i and j is $\Delta\bar{v}(t_0)$ at the current time instance t_0, it takes the time $T_b = \frac{0-\Delta\bar{v}(t_0)}{\alpha_i+\alpha_j}$ to reach $\Delta\bar{v}(t_0 + T_b) = 0$, while the maximum braking acceleration $(\alpha_i + \alpha_j)$ is applied to both robots. In order to remain farther away from the safety distance D_s, the following safety constraint needs to be satisfied

$$\|\Delta\mathbf{p}_{ij}\| + \int_{t_0}^{t_0+T_b} \Delta\bar{v}(t_0 + t)\,\mathrm{d}t \geq D_s, \quad \forall i \neq j$$

where $\Delta\bar{v}(t_0 + t) = \Delta\bar{v}(t_0) + (\alpha_i + \alpha_j)t$, which means that

$$\|\Delta\mathbf{p}_{ij}\| - \frac{(\Delta\bar{v})^2}{2(\alpha_i + \alpha_j)} \geq D_s, \quad \forall i \neq j. \tag{5}$$

Note that this safety constraint only needs to be enforced when agents are moving closer to each other, i.e., when $\Delta\bar{v} \leq 0$. It is always considered safe when the agents are moving away from each other, i.e., when $\Delta\bar{v} > 0$. By combining this observation with the constraint in (5) gives

$$-\frac{\Delta\mathbf{p}_{ij}^T}{\|\Delta\mathbf{p}_{ij}\|}\Delta\mathbf{v}_{ij} \leq \sqrt{2(\alpha_i + \alpha_j)(\|\Delta\mathbf{p}_{ij}\| - D_s)} \quad \forall i \neq j.$$

As such, the pairwise safe set $\mathscr{C}_{ij}$ is defined as

$$\begin{aligned}\mathscr{C}_{ij} &= \{(\mathbf{p}_i, \mathbf{v}_i) \in \mathbb{R}^4 \,|h_{ij}(\mathbf{p}, \mathbf{v}) \geq 0\} \quad \forall i \neq j,\\ h_{ij}(\mathbf{p}, \mathbf{v}) &= \sqrt{2(\alpha_i + \alpha_j)(\|\Delta\mathbf{p}_{ij}\| - D_s)} + \frac{\Delta\mathbf{p}_{ij}^T}{\|\Delta\mathbf{p}_{ij}\|}\Delta\mathbf{v}_{ij}\end{aligned} \tag{6}$$

where $h_{ij}(\mathbf{p},\mathbf{v})$ is the level set function of the set $\mathscr{C}_{ij}$ as well as the ZCBF candidate used to ensure the forward invariance of $\mathscr{C}_{ij}$. For ease of notation, we use h_{ij} to denote $h_{ij}(\mathbf{p},\mathbf{v})$. As seen, the forward invariance of $\mathscr{C}_{ij}$ requires the satisfaction of the ZCBF constraint in (3). Combining (6) with (3), the safety barrier constraint can be written as

$$-\Delta\mathbf{p}_{ij}^T\Delta\mathbf{u}_{ij} \le \gamma h_{ij}^3\|\Delta\mathbf{p}_{ij}\| - \frac{(\Delta\mathbf{v}_{ij}^T\Delta\mathbf{p}_{ij})^2}{\|\Delta\mathbf{p}_{ij}\|^2} + \|\Delta\mathbf{v}_{ij}\|^2 + \frac{(\alpha_i+\alpha_j)\Delta\mathbf{v}_{ij}^T\Delta\mathbf{p}_{ij}}{\sqrt{2(\alpha_i+\alpha_j)(\|\Delta\mathbf{p}_{ij}\|-D_s)}}, \quad \forall i \neq j. \tag{7}$$

Despite the seemingly complex form of (7), it should be noted that the safety barrier constraint can in fact be written as a linear constraint in $\mathbf{u}_i$ and $\mathbf{u}_j$, which in turn can be represented as $A_{ij}\mathbf{u} \le b_{ij}$, with

$$A_{ij} = [0,\ldots,\underbrace{-\Delta\mathbf{p}_{ij}^T}_{\text{agent } i},\ldots,\underbrace{\Delta\mathbf{p}_{ij}^T}_{\text{agent } j},\ldots,0]$$

and $b_{ij} = \gamma h_{ij}^3\|\Delta\mathbf{p}_{ij}\| - \frac{(\Delta\mathbf{v}_{ij}^T\Delta\mathbf{p}_{ij})^2}{\|\Delta\mathbf{p}_{ij}\|^2} + \frac{(\alpha_i+\alpha_j)\Delta\mathbf{v}_{ij}^T\Delta\mathbf{p}_{ij}}{\sqrt{2(\alpha_i+\alpha_j)(\|\Delta\mathbf{p}_{ij}\|-D_s)}} + \|\Delta\mathbf{v}_{ij}\|^2$.

We denote all pairwise safety barrier constraints in (7) as the *centralized safety barrier certificates* for the multirobot system. The centralized safety barrier certificates define the admissible control space S_u as the intersection of different half-spaces

$$S_u = \{\mathbf{u} \in \mathbb{R}^{2N} \mid A_{ij}\mathbf{u} \le b_{ij} \quad \forall i \neq j\}. \tag{8}$$

Note that as the swarm size N grows, S_u might become empty, which is a feasibility problem that will be addressed in Section VI.

The safe set $\mathscr{C}$ for the overall system is now formally defined as the intersection of all possible pairwise safe sets, i.e., *all* potential pairwise collisions are eliminated from $\mathscr{C}$

$$\mathscr{C} = \prod_{i\in\mathscr{M}}\left\{\bigcap_{\substack{j\in\mathscr{M}\\ j\neq i}}\mathscr{C}_{ij}\right\}$$

where the product is the Cartesian product over the state space of all agents.

Definition III.1: The multirobot system indexed by $\mathscr{M}$, with dynamics given in (4), is *safe* if the ensemble state $(\mathbf{p}(t),\mathbf{v}(t)) \in \mathscr{C}$, $\forall t \ge 0$.

With definitions for safety and the safety barrier certificates, the following result is presented to ensure the safety of the multirobot system.

Theorem III.1: Given a multirobot system indexed by $\mathscr{M}$ with dynamics in (4), if the controller $\mathbf{u}(t)$ satisfies the centralized safety barrier certificates in (7), and $(\mathbf{p}(0),\mathbf{v}(0)) \in \mathscr{C}$, then the multirobot system is guaranteed to be *safe*.

Proof: If the controller satisfies the centralized safety barrier certificates, then $\mathbf{u}(t)$ is always constrained inside the admissible control space S_u and satisfies all the pairwise safety barrier constraints in (7). As ensured by the ZCBFs in [26], $\mathscr{C}_{ij}$ is forward invariant for all $i \neq j$, i.e., $\mathscr{C}$ is forward invariant. Since $(\mathbf{p}(0),\mathbf{v}(0)) \in \mathscr{C}$, $(\mathbf{p}(t),\mathbf{v}(t))$ will stay in $\mathscr{C}$ for all time. Hence, the multirobot system is guaranteed to be *safe*. ■

Note that the safety barrier certificates can also deal with static or moving obstacles. When bounded with circles, the obstacles can be treated as agents with no control inputs. For example, consider a moving obstacle k with radius R_k centered at $\mathbf{p}_k$, a ZCBF $\bar{h}_{ik}(\mathbf{p},\mathbf{v})$ similar to (6) can be designed to ensure that agent i does not collide with obstacle k

$$\bar{h}_{ik}(\mathbf{p},\mathbf{v}) = \sqrt{2\alpha_i(\|\Delta\mathbf{p}_{ik}\| - (\frac{D_s}{2}+R_k))} + \frac{\Delta\mathbf{p}_{ik}^T}{\|\Delta\mathbf{p}_{ik}\|}\Delta\mathbf{v}_{ik}$$

where obstacle k is assumed to be moving at a constant velocity $\mathbf{v}_k$.

A. Minimally Invasive Collision Avoidance Using a QP-Based Controller

In order to ensure that the nominal, primary control objective is respected to the highest degree possible, the collision avoidance strategy needs to be minimally invasive in the sense that it should modify the nominal controller as little as possible. And, as we have seen that the safety constraint is linear in the control signal, we can add a quadratic cost that penalizes deviations from the nominal controller (in the least-squares sense), resulting in a quadratic programming (QP) problem. As a consequence, the QP-based controller minimizes the difference between the actual control command $\mathbf{u}_i$ and nominal control command $\hat{\mathbf{u}}_i$, while ensuring safety using the *centralized safety barrier certificates* discussed in the prior section

$$\begin{aligned} \mathbf{u}^* = \underset{\mathbf{u}\in\mathbb{R}^{2N}}{\operatorname{argmin}} \quad & J(\mathbf{u}) = \sum_{i=1}^{N}\|\mathbf{u}_i - \hat{\mathbf{u}}_i\|^2 \\ \text{s.t.} \quad & A_{ij}\mathbf{u} \le b_{ij}, \qquad \forall i \neq j \\ & \|\mathbf{u}_i\|_\infty \le \alpha_i, \qquad \forall i \in \mathscr{M}. \end{aligned} \tag{9}$$

The resulting controller $\mathbf{u}$ mimics the nominal controller $\hat{\mathbf{u}}$ completely when the system is safe, and only modifies its behavior when collisions are truly imminent, which is what we mean by minimally invasive safe modification of the nominal controller $\hat{\mathbf{u}}$.

This QP-based controller requires central computation and coordination, which suffers from poor scalability, reactiveness, and robustness as the number of agents grows. But it provides a starting point toward decentralized safety barrier certificates.

B. Reduced Neighborhoods

The *centralized safety barrier certificates* in the previous section considers *all* pairs of robots, which is potentially a very large number. Topologically speaking, that requires all-to-all interactions, i.e., a complete graph. As a result, the associated computation and sensing requirements will increase significantly as the number of robots increases. Motivated by the fact that agents sufficiently far apart will not collide within a finite time horizon, a neighborhood notion should be developed that reduces the required information structure to a disk graph, i.e., only pairs of nearby (within a certain distance) robots are needed, as shown

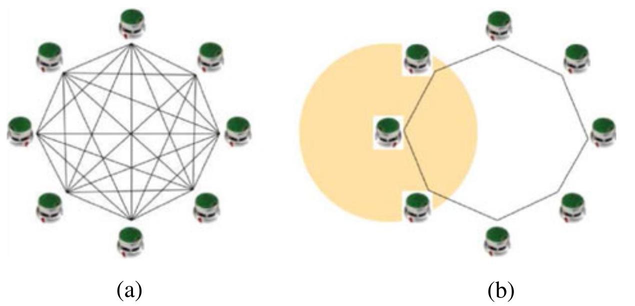


Fig. 2. Reduced information requirement graph (a) complete graph (b) disk graph.

in Fig. 2. The neighborhood set of agent i is thus defined as

$$\mathscr{N}_i = \{j \in \mathscr{M} \mid \|\Delta \mathbf{p}_{ij}\| \leq D^i_{\mathscr{N}}, j \neq i\} \tag{10}$$

where

$$D^i_{\mathscr{N}} = D_s + \frac{1}{2(\alpha_i + \alpha_{\min})}\left(\sqrt[3]{\frac{2(\alpha_i+\alpha_{\max})}{\gamma}} + \beta_i + \beta_{\max}\right)^2$$

is the choice of the radius of the neighborhood which will be elucidated later, $\alpha_{\min} = \min\limits_{j\in\mathscr{M}}\{\alpha_j\}$ and $\alpha_{\max} = \max\limits_{j\in\mathscr{M}}\{\alpha_j\}$ are lower and upper bounds of all agents' acceleration limits, and $\beta_{\max} = \max\limits_{j\in\mathscr{M}}\{\beta_j\}$ is the upper bound of all agents' speed limits.

With this notion of neighborhood, we will say that each agent only needs to consider its nearby agents to avoid collision, even though the computation, so far, is done by a centralized unit. This, however, will be related in subsequent sections. A similar notion for agents with identical acceleration limits was derived in [20], and in this paper safety barrier certificates are synthesized with ZCBFs for agents with different acceleration limits.

Theorem III.2: Agent $i \in \mathscr{M}$ only needs to form ZCBFs with its neighbors, as defined in (10), to guarantee safety.

Proof: Consider any agent k that is not a neighbor of agent i, i.e., $D_{ik} = \|\Delta \mathbf{p}_{ik}\| > D^i_{\mathscr{N}}$. We will prove that agent k is guaranteed to satisfy the pairwise safety barrier constraint (7) with agent i no matter what control action is taken.

Since $\dot{D}_{ik} = \|\Delta \dot{\mathbf{p}}_{ik}\| = \frac{\Delta \mathbf{p}_{ik}^T}{\|\Delta \mathbf{p}_{ik}\|}\Delta \mathbf{v}_{ik}$, h_{ik} in (6) can be reformulated in terms of D_{ik} and $\dot{D}_{ik}$

$$h_{ik} = \dot{D}_{ik} + \sqrt{2(\alpha_i+\alpha_k)(D_{ik}-D_s)}.$$

The derivative of h_{ik} is given by

$$\dot{h}_{ik} = \ddot{D}_{ik} + \sqrt{\frac{\alpha_i+\alpha_k}{2(D_{ik}-D_s)}}\dot{D}_{ik}.$$

With the velocity and acceleration limits of both agents, the lower bounds of h_{ik} and $\dot{h}_{ik}$ can be derived by considering the worst case scenario ($\ddot{D}_{ik} = -\alpha_i - \alpha_k$, $\dot{D}_{ik} = -\beta_i - \beta_k$). Since agent k can be any agent in the multirobot system, these lower bounds can be further relaxed with the bounds on all agents' acceleration and speed limits

$$\begin{aligned} h_{ik} &\geq \sqrt{2(\alpha_i+\alpha_k)(D_{ik}-D_s)} - \beta_i - \beta_k \\ &\geq \sqrt{2(\alpha_i+\alpha_{\min})(D_{ik}-D_s)} - \beta_i - \beta_{\max} \\ \dot{h}_{ik} &\geq -\alpha_i - \alpha_k - \sqrt{\frac{\alpha_i+\alpha_k}{2(D_{ik}-D_s)}}(\beta_i+\beta_k) \\ &\geq -\alpha_i - \alpha_{\max} - \sqrt{\frac{\alpha_i+\alpha_{\max}}{2(D_{ik}-D_s)}}(\beta_i+\beta_{\max}). \end{aligned}$$

From $D_{ik} > D^i_{\mathscr{N}}$, we get $\sqrt{2(\alpha_i+\alpha_{\min})(D_{ik}-D_s)} > \beta_i + \beta_{\max}$ and $h_{ik} > \sqrt[3]{\frac{2(\alpha_i+\alpha_{\max})}{\gamma}}$. Therefore

$$\begin{aligned} \dot{h}_{ik} &\geq -\alpha_i - \alpha_{\max} - \sqrt{(\alpha_i+\alpha_{\max})(\alpha_i+\alpha_{\min})} \\ &\geq -2(\alpha_i+\alpha_{\max}) \geq -\gamma h_{ik}^3. \end{aligned}$$

This means that no matter what control action agent k takes, it always satisfies the pairwise safety barrier constraint (7), with agent i. Therefore, there is no need for agent i to consider agent k, and the result follows. ■

With Theorem III.2, the QP-based controller (9) can be simplified by only checking the safety of a multirobot system with disk information graph

$$\begin{aligned} \mathbf{u}^* = \underset{\mathbf{u}\in\mathbb{R}^{2N}}{\text{argmin}} \quad & J(\mathbf{u}) = \sum_{i=1}^{N}\|\mathbf{u}_i - \hat{\mathbf{u}}_i\|^2 \\ \text{s.t.} \quad & A_{ij}\mathbf{u} \leq b_{ij} \qquad \forall i \in \mathscr{M},\ \forall j \in \mathscr{N}_i \\ & \|\mathbf{u}_i\|_\infty \leq \alpha_i \qquad \forall\, i \in \mathscr{M}. \end{aligned} \tag{11}$$

The radius $D^i_{\mathscr{N}}$ of the disk information graph can be designed by choosing appropriate γ, such that $D^i_{\mathscr{N}}$ is no larger than the sensing range of agents. Note that this notion of neighborhood is still valid when the *safety barrier certificates* are distributed to individual agent.

C. Simulated Centralized Safety Barrier Certificates

The centralized safety barrier certificates are validated on a simulated multirobot system consisting of 20 mobile robots modeled with double integrator dynamics. As illustrated in Fig. 3(a), all agents were equally spaced on a circle at the beginning. The goal of the nominal controller is to make all agents swap their positions with the agents on the opposite side of the circle. The nominal controller is designed as a simple PD controller $\hat{\mathbf{u}}_i = -k_1(\mathbf{p}_i - \mathbf{r}_i) - k_2\mathbf{v}_i$, which drives the robot to the desired position $\mathbf{r}_i$ with zero velocity. Intuitively, all agents will collide at the center of the circle if no collision avoidance strategy is executed. This test case generates a "crowded" and collision-prone scenario to validate the effectiveness of the safety barrier certificates.

The result of running the nominal position swapping controller wrapped with the centralized safety barrier certificates is illustrated in Fig. 3. When collisions were not imminent, all agents moved toward the center and followed the nominal controllers closely [see Fig. 3(b)]. As robots became too close to each other, the safety barrier certificates modified the controller as little as possible to keep the desired safety distance [see Fig. 3(c)]. In the end, all robots successfully navigated

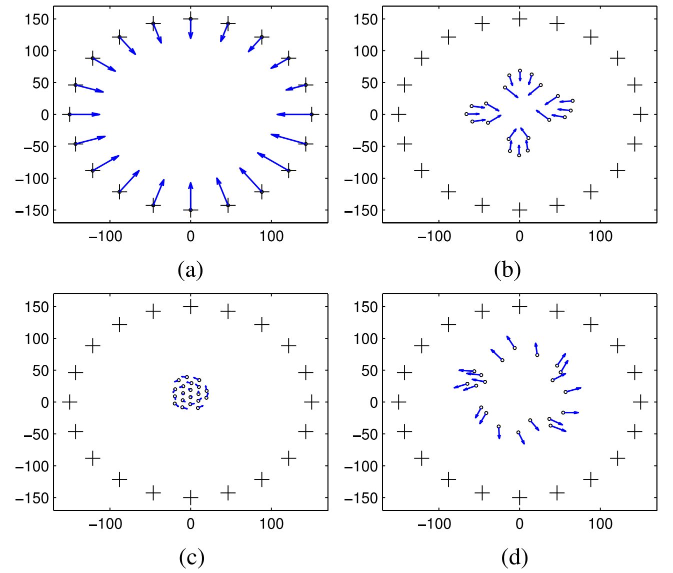


Fig. 3. Simulation results of a multirobot position swapping task regulated by the centralized safety barrier certificates. The circles and arrows represent the current positions and velocities of the agents. The safety distance $D_s = 10$. (a) Time step 3. (b) Time step 325. (c) Time step 770. (d) Time step 1000.

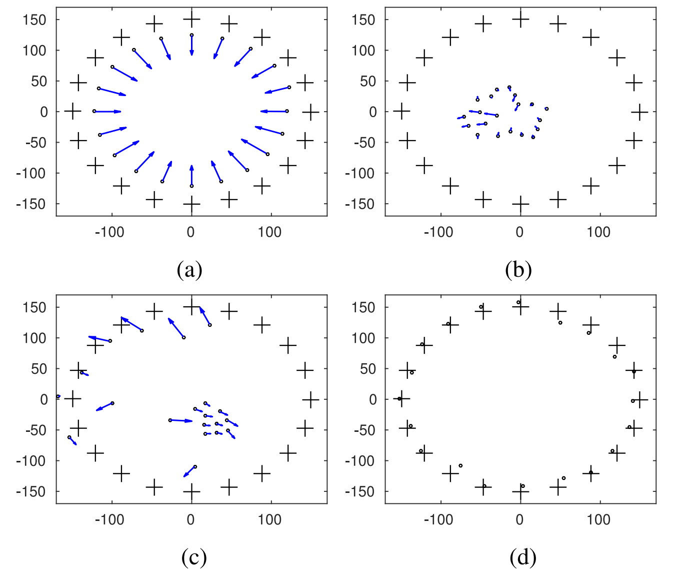


Fig. 4. Simulation results of a multirobot position swapping task regulated by the decentralized safety barrier certificates. The circles and arrows represent the current positions and velocities of the agents. The safety distance $D_s = 10$. Note that the agents moves nonsymmetrically due to different controller gains used by different agents. (a) Time step 135. (b) Time step 965. (c) Time step 2358. (d) Time step 5000.

through the “crowded” region and headed straightly toward the opposite side of the circle without colliding into each other [see Fig. 3(d)].

IV. Decentralized Safety Barrier Certificates

The centralized safety barrier certificates proposed in Section III ensure provably safe multirobot coordination, while the reliance on a central coordination unit potentially compromises the multirobot system’s scalability, reactiveness, and robustness. To address those issues caused by the centralized coordination, the safety barrier certificates will be distributed to individual agents without losing the safety guarantee in this section.

The safety barrier certificates define the admissible control space, which can be partitioned to smaller subsets. Each agent only needs to stay within its own subset to remain safe. Motivated by the fact that agents with larger acceleration limits are more agile in performing collision avoidance maneuvers, the admissible control space is partitioned based on the agents’ acceleration limits. For more details about decentralized safety barrier certificates for heterogeneous multirobot systems, we refer the reader to [22]. More specifically, the pairwise safety barrier constraint (7) between agent i and agent j is distributed to each agent as

$$-\Delta \mathbf{p}_{ij}^T \mathbf{u}_i \le \frac{\alpha_i}{\alpha_i + \alpha_j} b_{ij}$$

$$\Delta \mathbf{p}_{ij}^T \mathbf{u}_j \le \frac{\alpha_j}{\alpha_i + \alpha_j} b_{ij}.$$

With the decentralized safety barrier constraints and the notion of neighborhood, the sensing and computation tasks are completely distributed to each individual agent. Each agent $i \in \mathscr{M}$ runs their own version of QP-based controller

$$\begin{aligned} \mathbf{u}_i^* = \underset{\mathbf{u}_i \in \mathbb{R}^2}{\operatorname{argmin}} \quad & J(\mathbf{u}_i) = \|\mathbf{u}_i - \hat{\mathbf{u}}_i\| \\ \text{s.t.} \quad & \bar{A}_{ij}\mathbf{u}_i \le \bar{b}_{ij}, \qquad \forall j \in \mathscr{N}_i \\ & \|\mathbf{u}_i\|_\infty \le \alpha_i \end{aligned} \tag{12}$$

where $\bar{A}_{ij} = -\Delta \mathbf{p}_{ij}^T$, $\bar{b}_{ij} = \frac{\alpha_i}{\alpha_i + \alpha_j} b_{ij}$.

When the safety barrier certificates are distributed to each individual agents, the safety of the multirobot system is still guaranteed by the following result.

Theorem IV.1: Given a multirobot system indexed by $\mathscr{M}$ with dynamics in (4), if the controller $\mathbf{u}_i$ satisfies the decentralized safety barrier certificates in (12) for all agent $i \in \mathscr{M}$, and $(\mathbf{p}(0), \mathbf{v}(0)) \in \mathscr{C}$, then the multirobot system is guaranteed to be safe.

Proof: If all agents’ controllers satisfy the decentralized safety barrier certificates, then $\mathscr{C}_{ij}$ is forward invariant $\forall i \in \mathscr{M}$, $j \in \mathscr{N}_i$, as ensured by the ZCBFs. When $j \notin \mathscr{N}_i$, $(\mathbf{p}_i, \mathbf{v}_i)$ still stays in $\mathscr{C}_{ij}$ due to Theorem III.2. Therefore, $\mathscr{C}$ is forward invariant, and this completes the proof. ■

Although safety is still ensured, the collision avoidance interventions enforcing the decentralized safety certificates have to happen earlier than the centralized case due to the lack of central coordination.

A. Simulation for Decentralized Safety Barrier Certificates

The decentralized safety barrier certificates are validated on a simulated multirobot system for the same position swapping task described in Section III-C. As illustrated in Fig. 4, all agents successfully navigated through the crowded center region to the opposite side of the circle without collision. Meanwhile,

TABLE I
COMPUTATIONAL COMPLEXITY OF THE CERTIFICATES

Type of certificate	M_d	M_c (worst case)
Centralized	$2N$	$\frac{N(N-1)}{2}$
Centralized with neighborhood	$2N$	$\frac{N}{2}\min\left\{N-1, \text{ceiling}\left(\frac{D_{\mathscr{N}}^2}{D_s^2}\right)\right\}$
Decentralized	2	$N-1$
Decentralized with neighborhood	2	$\min\left\{N-1, \text{ceiling}\left(\frac{D_{\mathscr{N}}^2}{D_s^2}\right)\right\}$

TABLE II
COMPUTATION TIME OF THE CERTIFICATES

	Computation time per iteration (ms)		
Type of certificate	$N = 20$	$N = 60$	$N = 100$
Centralized with neighborhood	11.8	28.8	238.3
Decentralized with neighborhood	6.00	5.99	8.05

the agents were much slower in the vicinity of other agents and spent more time to complete the same task without central coordination.

B. Computational Complexity and Solver Details

The computational complexity of the QP for enforcing the safety barrier certificates is analyzed in this section. Numbers of decision variables M_d and linear constraints M_c are two important factors that determine the computation complexity of the QP. As shown in Table I, the decentralization of the safety barrier certificates and the neighborhood reduction significantly decrease M_d and M_c (in the worst case). In particular, for the decentralized safety barrier certificates with reduced neighborhood, each robot only needs to run a QP with two decision variables and at most ceiling$(\frac{D_{\mathscr{N}}^2}{D_s^2})$ constraints for arbitrary swarm size N. Here the ceiling$(\cdot)$ function maps a real number to the smallest integer no less than itself. In the worst case scenario, the neighborhood disk of the robot is densely populated with at most ceiling$(\frac{\pi D_{\mathscr{N}}^2}{\pi D_s^2})$ robots. Thus the decentralized safety barrier certificates are scalable to arbitrarily large groups of robots. Note that since the actual value of $D_{\mathscr{N}}$ depends both on D_s and γ according to (10), the sensing range of the robot can be used to replace $D_{\mathscr{N}}$ if a more specific upper bound is required.

The actual computation times of the certificates per iteration are listed in Table II. The centralized barrier certificates can handle up to 60 robots with an update rate of more than 30 Hz. Meanwhile, the decentralized barrier certificates can handle more than 100 robots with an update rate of more than 100 Hz. All the computations are performed on an Ubuntu laptop with a 2.60 GHz Intel Core i5 processor using the MATLAB *quadprog* solver.

V. RELAXED SAFETY BARRIER CERTIFICATES

The decentralized safety barrier certificates remove the need for central computation at the cost of using a more conservative collision avoidance strategy, i.e., the nominal controller might be modified earlier than truly necessary to avoid collisions. Motivated by the need to enlarge the admissible control space, the relaxed ZCBF is introduced through a relaxation parameter $k_r(t) \in [1, \infty)$, which is continuous in t.

The idea now is to insist that $\dot{h}(x) \geq -k_r(t)\boldsymbol{\kappa}(h(x))$, and prove that we can still formally guarantee the forward invariance of the set $\mathscr{C}$ when the ZCBF constraint is relaxed with $k_r(t)$. And the relaxed, feasible control set $S_r(x)$ becomes

$$S_r(x) = \{u \in U \mid L_f h(x) + L_g h(x) u + k_r(t)\boldsymbol{\kappa}(h(x)) \geq 0\}, \quad x \in \mathscr{D}.$$

Note that, in general, $S(x) \subseteq S_r(x)$ if $k_r(t) \geq 1, \forall t \geq 0$. And, the relaxed ZCBF reverts to the nominal ZCBF when $k_r(t) = 1, \forall t \geq 0$. The result for relaxed ZCBFs can be stated as

Theorem V.1: Given a set $\mathscr{C} \subset \mathbb{R}^n$ defined by (2) and a ZCBF h defined on $\mathscr{D}$ with $\mathscr{C} \subseteq \mathscr{D} \subset \mathbb{R}^n$, any Lipschitz continuous controller $u : \mathscr{D} \rightarrow \mathbb{R}$ such that $u \in S_r(x)$ for the system (1) renders the set $\mathscr{C}$ forward invariant.

Proof: Consider the nominal dynamical system $\dot{z} = -\boldsymbol{\kappa}(z)$. We know that the solution is of the form $z(t) = \sigma(z(0), t)$, where $\boldsymbol{\kappa}$ is a class-$\mathscr{K}$ function and σ is a class-$\mathscr{K}\mathscr{L}$ function, as shown, for example, in [29].

The corresponding "relaxed" dynamical system is given by $\dot{z}' = -k_r(t)\boldsymbol{\kappa}(z')$, and the solution to this system is

$$z'(t) = \sigma\left(z'(0), \int_0^t k_r(s)\,\mathrm{d}s\right),$$

which can be verified by taking the time derivative of $\sigma(u(t), v(t))$, where $u(t) = z'(0)$, $v(t) = \int_0^t k_r(s)\,\mathrm{d}s$, yielding

$$\dot{z}' = \frac{\partial \sigma}{\partial u}\frac{\mathrm{d}u}{\mathrm{d}t} + \frac{\partial \sigma}{\partial v}\frac{\mathrm{d}v}{\mathrm{d}t} = 0 - \boldsymbol{\kappa}(z')k_r(t).$$

Comparing the relaxed system with $\dot{h} \geq -k_r(t)\boldsymbol{\kappa}(h)$ (use the Comparison Lemma [29]), we have that

$$h(x(t)) \geq \sigma\left(h(0), \int_0^t k_r(s)\mathrm{d}s\right).$$

Note that $\int_0^t k_r(s)\,\mathrm{d}s \geq 0$ and $\int_0^\infty k_r(s)\,\mathrm{d}s = \infty$. And, by the properties of class-$\mathscr{K}\mathscr{L}$ functions ($\sigma(u, v)$ is strictly increasing under fixed v, and $\sigma(0, v) = 0$; $\sigma(u, v)$ is monotonically decreasing under fixed u, and $\sigma(u, \infty) = 0$), if $x(0) \in \mathscr{C}$, it follows that $h(x(t)) \geq 0, \forall t \geq 0$. Thus, the forward invariance of the set $\mathscr{C}$ is still preserved if the control signal stays in the relaxed control space $S_r(x)$. ■

Since $k_r(t)$ can be freely adjusted online in response to changing environmental conditions, it is selected as an optimization decision variable in the QP-based controller.

For agent i, the pairwise safety barrier constraint (7) with agent j is relaxed with a relaxation factor k_{rj}

$$-\Delta\mathbf{p}_{ij}^T \Delta\mathbf{u}_{ij} \leq k_{rj}\gamma h_{ij}^3 \|\Delta\mathbf{p}_{ij}\| - \frac{(\Delta\mathbf{v}_{ij}^T \Delta\mathbf{p}_{ij})^2}{\|\Delta\mathbf{p}_{ij}\|^2} + \|\Delta\mathbf{v}_{ij}\|^2 + \frac{(\alpha_i + \alpha_j)\Delta\mathbf{v}_{ij}^T \Delta\mathbf{p}_{ij}}{\sqrt{2(\alpha_i + \alpha_j)(\|\Delta\mathbf{p}_{ij}\| - D_s)}} \quad \forall i \neq j$$

which are assembled into the relaxed safety barrier certificates.

In the nominal case, there is no relaxation on the safety barrier constraints, i.e., $\hat{K}_r = [\hat{k}_{r1}, \hat{k}_{r2}, \ldots, \hat{k}_{rM}]^T = \mathbf{1}$ with $M = |\mathscr{N}_i|$ being the cardinality of agent i's neighborhood set. The actual relaxation vector $K_r = [k_{r1}, k_{r2}, \ldots, k_{rM}]^T$ is added to the cost of the QP-based controller to penalize deviation from $\hat{K}_r$

$$\begin{aligned}(\mathbf{u}_i^*, K_r^*) = \underset{\mathbf{u}_i \in \mathbb{R}^2, K_r \in \mathbb{R}^M}{\text{argmin}} \quad & J(\mathbf{u}_i, K_r) = \|\mathbf{u}_i - \hat{\mathbf{u}}_i\|^2 \\ & \quad + c_K \|K_r - \hat{K}_r\|^2 \\ \text{s.t.} \quad & \tilde{A}_{ij}\mathbf{u}_i \le \frac{\alpha_i}{\alpha_i + \alpha_j}\tilde{b}_{ij}, \quad \forall j \in \mathscr{N}_i \\ & \|\mathbf{u}_i\|_\infty \le \alpha_i,\end{aligned}$$

where c_K is the weight for the cost of relaxation, $\tilde{A}_{ij} = [-\mathbf{p}_{ij}^T]$

$$\begin{aligned}\tilde{b}_{ij} = \ & k_{rj}\gamma h_{ij}^3 \|\Delta\mathbf{p}_{ij}\| - \frac{(\Delta\mathbf{v}_{ij}^T \Delta\mathbf{p}_{ij})^2}{\|\Delta\mathbf{p}_{ij}\|^2} + \|\Delta\mathbf{v}_{ij}\|^2 \\ & + \frac{(\alpha_i + \alpha_j)\Delta\mathbf{v}_{ij}^T\Delta\mathbf{p}_{ij}}{\sqrt{2(\alpha_i + \alpha_j)(\|\Delta\mathbf{p}_{ij}\| - D_s)}}.\end{aligned}$$

Let the optimization parameter be $\mu_i = [\mathbf{u}_i^T, \sqrt{c_K} K_r^T]^T$, the QP-based controller can be rearranged as

$$\begin{aligned}\mu_i^* = \underset{\mu_i \in \mathbb{R}^{2+M}}{\text{argmin}} \quad & J(\mu_i) = \|\mu_i - \hat{\mu}_i\|^2, \\ \text{s.t.} \quad & \bar{A}_{ij}\mu_i \le \frac{\alpha_i}{\alpha_i + \alpha_j}\bar{b}_{ij}, \quad \forall j \in \mathscr{N}_i, \\ & \|\mathbf{u}_i\|_\infty \le \alpha_i \end{aligned} \tag{13}$$

where $\hat{\mu}_i = [\ \hat{\mathbf{u}}_i^T, \sqrt{c_K}\hat{K}_r^T\ ]^T$, $\bar{A}_{ij} = [-\mathbf{p}_{ij}^T, 0, \ldots, -\frac{\alpha_i \gamma h_{ij}^3 \|\Delta\mathbf{p}_{ij}\|}{(\alpha_i + \alpha_j)\sqrt{c_K}}, \ldots 0]$,

$$\begin{aligned}\bar{b}_{ij} = & -\frac{(\Delta\mathbf{v}_{ij}^T \Delta\mathbf{p}_{ij})^2}{\|\Delta\mathbf{p}_{ij}\|^2} + \|\Delta\mathbf{v}_{ij}\|^2 \\ & + \frac{(\alpha_i + \alpha_j)\Delta\mathbf{v}_{ij}^T\Delta\mathbf{p}_{ij}}{\sqrt{2(\alpha_i + \alpha_j)(\|\Delta\mathbf{p}_{ij}\| - D_s)}}.\end{aligned}$$

By putting K_r into the optimization cost, the QP-based controller will automatically expand the admissible control space for those agents that are unnecessarily constrained.

It should be noted that the continuity of K_r is guaranteed when using the QP-based controller, as long as the nominal controller $\hat{\mathbf{u}}_i$ is Lipschitz continuous as shown in [30].

VI. Feasible Safety Barrier Certificates

The safety barrier certificates ensure safety as long as all agents' control actions stay within the admissible control space. However, the admissible control space might become empty in extreme scenarios, where the QP-based controller becomes infeasible, i.e., no solution exists and safety cannot be guaranteed. This can be remedied by the robustness feature of ZCBF, which means the ZCBF will force the states back to the safe set $\mathscr{C}$ when violation occurs [26]. But it is important to synthesize safety barrier certificates with guaranteed feasibility such that a solution to the QP-based controller always exists.

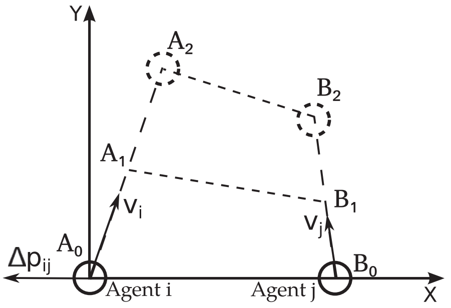


Fig. 5. Braking mode of two agents i ands j.

A. Safety Barrier Certificates With Guaranteed Feasibility

Intuitively, if all agents can decelerate to zero velocity without colliding into each other, then they can stay static to remain safe thereafter. Therefore, we can synthesize a ZCBF which ensures that decelerating to zero velocity is always a feasible safe control action. This is achieved by designing two modes of operations: normal mode and braking mode. The safety barrier certificates operate in the normal mode when the QP-based controller is feasible, and switch to the braking mode when it is infeasible. In the braking mode, the agent will apply the maximum braking force until it stops.

Assume that two agents start to apply their maximum braking force at any arbitrary time t_0, they both reach zero velocity at time t_s. Note that two agents might not stop at the same time, in which case t_s is the time instance when the latter agent reaches zero velocity. As illustrated in Fig. 5, A_0 and B_0 are the positions of agent i and j at time t_0, A_2 and B_2 are the positions of agent i and j at time t_s, A_1 and B_1 are the middle points of $\overline{A_0A_2}$ and $\overline{B_0B_2}$, where $\|\overline{A_0A_2}\| = \frac{\|\mathbf{v}_i\|^2}{2\alpha_i}$, $\|\overline{B_0B_2}\| = \frac{\|\mathbf{v}_j\|^2}{2\alpha_j}$.

The agents always need to be further away than a safety distance D_s while decelerating to zero velocity. Therefore the following safety constraint needs to be satisfied

$$\|\Delta\mathbf{p}_{ij}(t)\| \ge D_s, \quad \forall t \in [t_0, \quad t_s]. \tag{14}$$

Synthesizing safety barrier constraints for every time instance $t \in [t_0, \quad t_s]$ might not be done online efficiently. Instead, we propose the following simplified constraint that ensures (14)

$$\begin{aligned}\hat{h}_{ij}(\mathbf{p}, \mathbf{v}) = \ & \|\overline{A_1B_1}\|^2 - \left(D_s + \frac{\|\mathbf{v}_i\|^2}{4\alpha_i} + \frac{\|\mathbf{v}_j\|^2}{4\alpha_i}\right)^2 \ge 0 \\ \|\overline{A_1B_1}\|^2 = \ & \left(\frac{\|\mathbf{v}_i\|^2}{4\alpha_i}\cos\theta_i - \frac{\|\mathbf{v}_j\|^2}{4\alpha_j}\cos\theta_j + \|\Delta\mathbf{p}_{ij}\|\right)^2 \\ & + \left(\frac{\|\mathbf{v}_i\|^2}{4\alpha_i}\sin\theta_i - \frac{\|\mathbf{v}_j\|^2}{4\alpha_j}\sin\theta_j\right)^2 \end{aligned} \tag{15}$$

where $\theta_i = \measuredangle(\Delta\mathbf{p}_{ij}, \mathbf{v}_i), \theta_j = \measuredangle(\Delta\mathbf{p}_{ij}, \mathbf{v}_j)$.

Lemma VI.1: If both agent i and j start to apply the maximum braking force when $\hat{h}_{ij} \ge 0$, then the safety constraint (14) is satisfied for any time $t \in [t_0, \ t_s]$.

Proof: Pick any two points $A_p \in \overline{A_0A_2}$ and $B_p \in \overline{B_0B_2}$. Since A_1 and B_1 are the middle points of $\overline{A_0A_2}$ and $\overline{B_0B_2}$, we have $\|\overline{A_1A_p}\| \le \frac{\|\mathbf{v}_i\|^2}{4\alpha_i}$ and $\|\overline{B_1B_p}\| \le \frac{\|\mathbf{v}_j\|^2}{4\alpha_j}$. Because $\|\overline{A_pB_p}\| \ge \|\overline{A_1B_1}\| - \|\overline{B_1B_p}\| - \|\overline{A_1A_p}\|$ and $\hat{h}_{ij} \ge 0$, we can get

$$\|\overline{A_pB_p}\| \geq D_s, \quad \forall A_p \in \overline{A_0A_2},\ \forall B_p \in \overline{B_0B_2}.$$

Note that if agent i and j start to apply the maximum braking force from time t_0, $\mathbf{p}_i(t) \in \overline{A_0A_2}$ and $\mathbf{p}_j(t) \in \overline{B_0B_2}$ for any time $t \in [t_0, \quad t_s]$. Therefore, $\hat{h}_{ij} \geq 0$ implies the safety constraint (14) is satisfied. ■

Lemma VI.1 suggests that we only need to look ahead at one point to guarantee safety at all future time span $t \in [t_0,\ t_s]$. Next we will formulate a new safety barrier constraint for the safe set $\hat{\mathscr{C}}_{ij} = \{(\mathbf{p}_i, \mathbf{v}_i)\,|\,\hat{h}_{ij}(\mathbf{p}, \mathbf{v}) \geq 0\}$, i.e., $-\dot{\hat{h}}_{ij} \leq \gamma\hat{h}_{ij}^3$. This safety barrier constraint combined with (15) gives

$$\begin{aligned}&\Bigg(\frac{\|\mathbf{v}_i\|\|\mathbf{v}_j\|}{8\alpha_i\alpha_j}\mathbf{v}_j + \frac{\mathbf{v}_i\mathbf{v}_j\|\mathbf{v}_j\|}{8\alpha_i\alpha_j\|\mathbf{v}_i\|}\mathbf{v}_i - \frac{\Delta\mathbf{p}_{ij}\mathbf{v}_i}{2\alpha_i\|\mathbf{v}_i\|}\mathbf{v}_i - \frac{\|\mathbf{v}_i\|\Delta\mathbf{p}_{ij}}{2\alpha_i}\\ &\qquad + \frac{D_s\mathbf{v}_i}{\alpha_i} + \frac{\|\mathbf{v}_j\|^2\mathbf{v}_i}{4\alpha_i\alpha_j}\Bigg)\mathbf{u}_i + \Bigg(\frac{\|\mathbf{v}_i\|\|\mathbf{v}_j\|}{8\alpha_i\alpha_j}\mathbf{v}_i\\ &\qquad + \frac{\mathbf{v}_i\mathbf{v}_j\|\mathbf{v}_i\|}{8\alpha_i\alpha_j\|\mathbf{v}_j\|}\mathbf{v}_j + \frac{\Delta\mathbf{p}_{ij}\mathbf{v}_j}{2\alpha_j\|\mathbf{v}_j\|}\mathbf{v}_j\\ &\qquad + \frac{\|\mathbf{v}_j\|\Delta\mathbf{p}_{ij}}{2\alpha_j} + \frac{D_s\mathbf{v}_j}{\alpha_j} + \frac{\|\mathbf{v}_i\|^2\mathbf{v}_j}{4\alpha_i\alpha_j}\Bigg)\mathbf{u}_j\\ &\leq 2\Delta\mathbf{p}_{ij}\Delta\mathbf{v}_{ij} + \frac{\|\mathbf{v}_i\|}{2\alpha_i}\Delta\mathbf{v}_{ij}\mathbf{v}_i - \frac{\|\mathbf{v}_j\|}{2\alpha_j}\Delta\mathbf{v}_{ij}\mathbf{v}_j + \gamma\hat{h}_{ij}^3.\end{aligned}$$

The decentralized QP-based controller using the feasible safety barrier certificates can be written as

$$\begin{aligned}\mathbf{u}_i^* = \ &\underset{\mathbf{u}_i\in\mathbb{R}^2}{\text{argmin}} && J(\mathbf{u}_i) = \|\mathbf{u}_i - \hat{\mathbf{u}}_i\|\\ &\text{s.t.} && \hat{A}_{ij}\mathbf{u}_i \leq \hat{b}_{ij}, \qquad \forall j \neq i,\\ &&& \|\mathbf{u}_i\|_\infty \leq \alpha_i \end{aligned} \tag{16}$$

where $\hat{A}_{ij} = \frac{\|\mathbf{v}_i\|\|\mathbf{v}_j\|}{8\alpha_i\alpha_j}\mathbf{v}_j + \frac{\mathbf{v}_i\mathbf{v}_j\|\mathbf{v}_j\|}{8\alpha_i\alpha_j\|\mathbf{v}_i\|}\mathbf{v}_i - \frac{\Delta\mathbf{p}_{ij}\mathbf{v}_i}{2\alpha_i\|\mathbf{v}_i\|}\mathbf{v}_i$ $- \frac{\|\mathbf{v}_i\|\Delta\mathbf{p}_{ij}}{2\alpha_i} + \frac{D_s\mathbf{v}_i}{\alpha_i}$ $+ \frac{\|\mathbf{v}_j\|^2\mathbf{v}_i}{4\alpha_i\alpha_j}$, $\hat{b}_{ij} = \Delta\mathbf{p}_{ij}\Delta\mathbf{v}_{ij} + \frac{\|\mathbf{v}_i\|}{2\alpha_i}\Delta\mathbf{v}_{ij}\mathbf{v}_i + \frac{1}{2}\gamma\hat{h}_{ij}^3$.

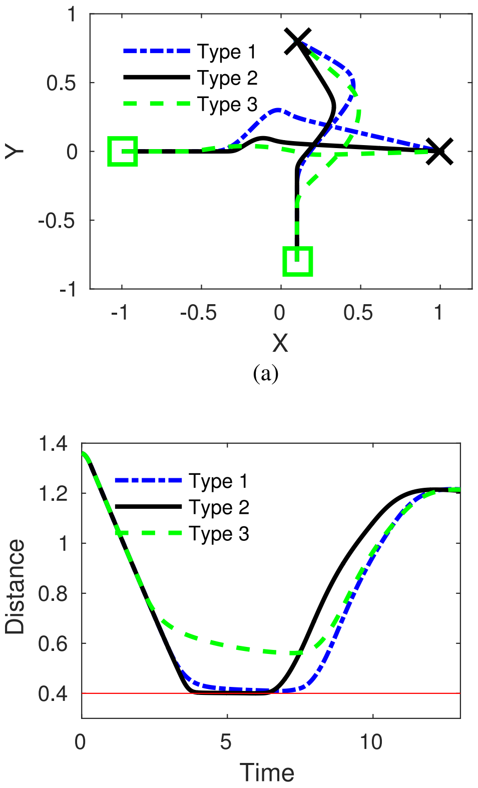
Note that if $\mathbf{v}_i = \mathbf{0}$ (agent i is stationary), the limit of $\hat{A}_{ij}$ is used, i.e., $\lim_{\mathbf{v}_i\to\mathbf{0}} \hat{A}_{ij} = 0$ and $\lim_{\mathbf{v}_i\to\mathbf{0}} \hat{A}_{ji} = -\frac{\Delta\mathbf{p}_{ji}\mathbf{v}_j}{2\alpha_j\|\mathbf{v}_j\|}\mathbf{v}_j - \frac{\|\mathbf{v}_j\|\Delta\mathbf{p}_{ji}}{2\alpha_j} + \frac{D_s\mathbf{v}_j}{\alpha_j}$. Intuitively, this means that agent i is free to choose its control action when it is stationary, while other agents can always react fast enough to avoid colliding with it.

This QP-based controller might still face feasibility issues. However, the agent can always decelerate to zero velocity safely due to Lemma VI.1. This results in the following hybrid braking controller

$$\mathbf{u}_i = \begin{cases}\mathbf{u}_i^*, & \text{if } \mathbf{u}_i^* \text{ exists},\\ -\alpha_i\frac{\mathbf{v}_i}{\|\mathbf{v}_i\|}, & \text{if } \mathbf{u}_i^* \text{ does not exist, and } \|\mathbf{v}_i\| \neq 0\\ 0, & \text{if } \mathbf{u}_i^* \text{ does not exist, and } \|\mathbf{v}_i\| = 0.\end{cases} \tag{17}$$

Theorem VI.2: If the robot agent i starts safe, the hybrid braking controller (17) guarantees that it always remains safe.

Proof: When the QP-based controller (16) is feasible, agent i always stays safe as ensured by the ZCBF. When the QP-based controller becomes infeasible, agent i would apply the maximum braking force until it stops. Then it remains static until the QP-based controller become feasible again. Thus, agent i will stay safe regardless of the feasibility of (16). ■



(a)

(b)

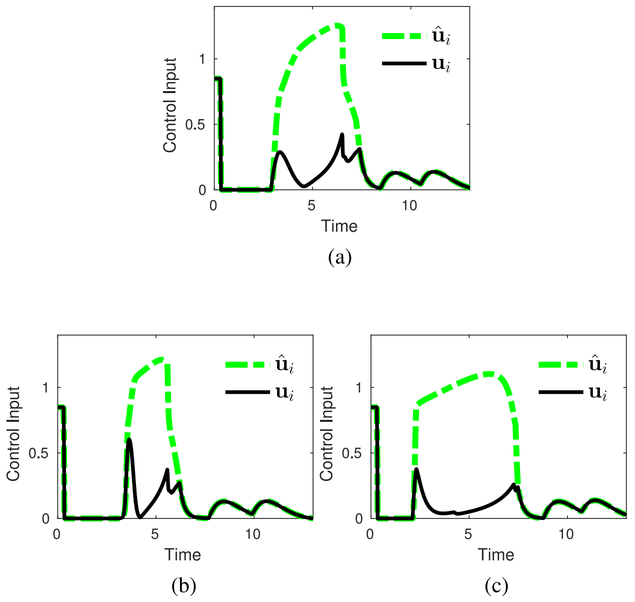
Fig. 6. Simulation results of three types of decentralized safety barrier certificates. Type 1 is the nominal safety barrier certificates in (12), Type 2 is the relaxed safety barrier certificates in (13), and Type 3 is the feasible safety barrier in certificates in (16). (a) Trajectories of two agents. The square and cross markers represent the initial and final positions of the two agents, respectively. (b) Evolution of the distance between two agents. The horizontal line is the safety distance ($D_s = 0.4$).

The braking mode constitutes an always feasible solution for the agents to safely decelerate to zero velocity ($\mathbf{v}_i = 0$), as opposed to decelerate to zero relative approaching velocity ($\Delta\bar{\mathbf{v}} = 0$) in (5). In order for such a braking mode to exist, the ZCBF is synthesized more conservatively than the nominal cases in (6). As such, the guaranteed feasible safety barrier certificates reduce the size of the admissible control space to ensure the existence of feasible control actions.

B. Simulation Results for Three Types of Decentralized Safety Barrier Certificates

Up until now, three types of decentralized safety barrier certificates, namely the nominal safety barrier certificates in (12), the relaxed safety barrier certificates in (13), and the feasible safety barrier certificates in (16), have been developed in this paper. Their performance is briefly compared here through simulations of two agents executing go-to-goal controllers regulated by three types of decentralized safety barrier certificates.

As illustrated in Fig. 6(a), two agents started simultaneously from the square markers, and successfully reached the cross markers in all three cases. Safety was always guaranteed as



(a)

(b)

(c)

Fig. 7. Magnitudes of nominal and actual control inputs for controllers regulated by three types of decentralized safety barrier certificates. (a) Nominal safety barrier certificates, the total time of intervention is 4.5 s. (b) Relaxed safety barrier certificates, the total time of intervention is 2.9 s. (c) Feasible safety barrier certificates, the total time of intervention is 5.4 s.

TABLE III
COMPARISON ACROSS THREE TYPES OF DECENTRALIZED SAFETY BARRIER CERTIFICATES

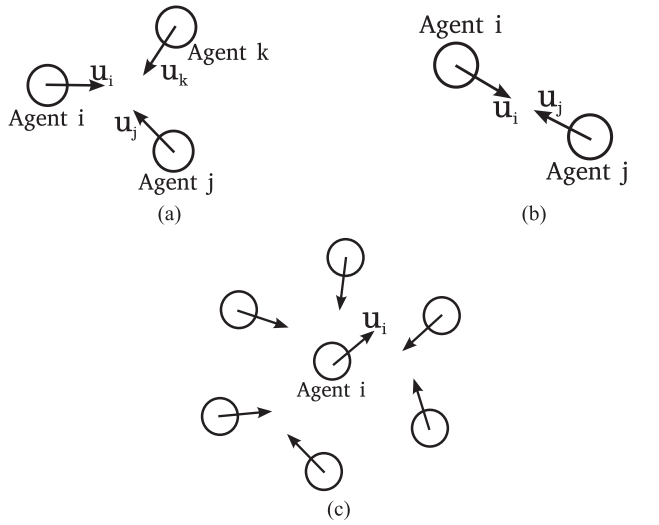
Type of certificate	Optimization parameters	Guaranteed safety	Guaranteed feasibility	Admissible control space
Nominal	$\mathbf{u}_i$	✓	×	Standard
Relaxed	$\mathbf{u}_i, K_r$	✓	×	Enlarged
Feasible	$\mathbf{u}_i$	✓	✓	Shrunken

shown by the evolution of the distance between two agents in Fig. 6(b). Among these three types of safety barrier certificates, the relaxed safety barrier certificates allow the largest admissible control space, and therefore agents can approach the safety distance more aggressively. The feasible safety barrier certificates have stricter constraints on the admissible control space to guarantee feasibility, which leads to more conservative behaviors when agents are approaching the safety distance.

The nominal and actual control signals during the simulation are illustrated in Fig. 7. Comparing the total time of intervention, it can be concluded that the relaxed safety barrier certificates are the least invasive to the nominal controller with only 2.9 s of intervention.

Based on the simulation results and earlier discussions in this paper, a comparison across the three types of decentralized safety barrier certificates is listed in Table III.

VII. Deadlock Detection and Resolution



(a)

(b)

(c)

Fig. 8. Three types of Deadlocks for robot agent i in a multirobot system. (a) Type 1 deadlock. (b) Type 2 deadlock. (c) Type 3 deadlock.

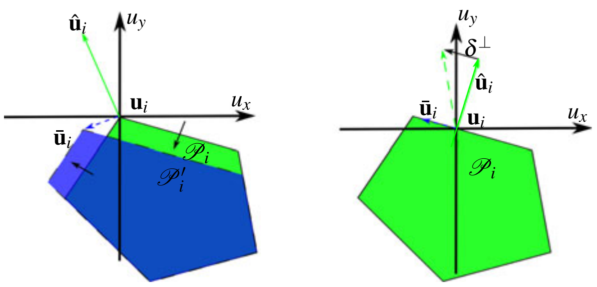


Fig. 9. Deadlock resolution methods, where $\hat{\mathbf{u}}_i$, $\mathbf{u}_i = 0$ and $\bar{\mathbf{u}}_i$ are the nominal, original and adjusted control commands respectively. (a) Type 1 deadlock, green($\mathscr{P}_i$) and blue($\mathscr{P}'_i$) polygons represent original and perturbed feasible control space. (b) Type 2 deadlock, the green polygon ($\mathscr{P}_i$) is the feasible control space.

When the objectives of multiple agents conflict with the safety barrier certificates, the agents might get stuck into a *deadlock*. In the deadlock scenario, the agents are safe but their tasks cannot be completed. Deadlock occurs because the safety barrier certificates are designed to take the local information only. In order to detect and resolve the deadlock issue, we first come up with a definition of the deadlock.

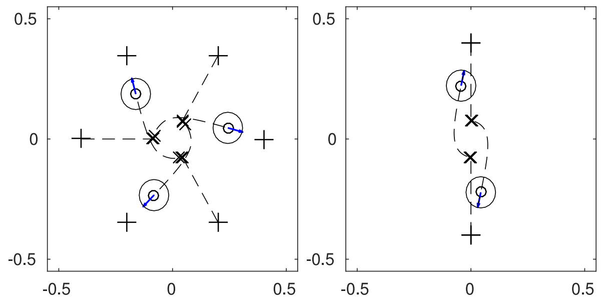
Definition VII.1: A robot agent i is said to be stuck in a deadlock, if it remains stationary ($u_i = 0$ and $v_i = 0$) and the nominal control command $\hat{u}_i \neq 0$.

With this definition, the deadlock scenarios can be further classified into three types based on the solution to the QP problem in (12). The admissible control space for the QP problem is a convex polygon $\mathscr{P}_i$ defined as the intersection of multiple half spaces, i.e.,

$$\mathscr{P}_i = \{\mathbf{u}_i \in \mathbb{R}^2 \mid \bar{A}_{ij}\mathbf{u}_i \leq \bar{b}_{ij}, \quad \forall i \neq j\}$$

where $\mathscr{P}_i$ is a decentralized counterpart of the centralized admissible control space S_u in (8). The size of the feasible control space, termed the *width of the feasible set* [30], can be evaluated with a Linear Program (LP)

$$\begin{aligned} \min_{\mathbf{u}_i \in \mathbb{R}^2, \delta_{LP} \in \mathbb{R}} \quad & \delta_{LP} \\ \text{s.t.} \quad & \bar{A}_{ij}\mathbf{u}_i \leq \bar{b}_{ij} + \delta_{LP}, \qquad \forall i \neq j, \\ & \|\mathbf{u}_i\|_\infty \leq \alpha_i. \end{aligned}$$



(a) (b)

Fig. 10. Simulated deadlock resolution. The circles, arrows, and dashed lines represent the current positions, velocities, and trajectories of different agents respectively. The cross markers represent the places where the deadlock occurs and the deadlock resolution algorithm is active. (a) Type 1 deadlock. (b) Type 2 deadlock.

The solution of the LP characterizes how much control margin is left for the strictest safety barrier constraint. If $\delta_{LP} \leq 0$, the corresponding QP is solvable. A more negative δ_{LP} indicates larger admissible control space. Otherwise, no feasible control option is available, and the admissible control space is empty.

The deadlock scenarios are categorized into the following three cases based on the relation between u_i and $\mathscr{P}_i$

1) Type 1 deadlock: $\delta_{LP} < 0, u_i = 0 \in \text{vertex}(\mathscr{P}_i)$;
2) Type 2 deadlock: $\delta_{LP} < 0, u_i = 0 \in \text{edge}(\mathscr{P}_i)$; and
3) Type 3 deadlock: $\delta_{LP} \geq 0$.

It should be noted that these three types of deadlock comprise all possible types of deadlocks. This is because u_i is either on the edge or the vertex of $\mathscr{P}_i$, when the optimal solution of the constrained QP controller ($u_i = 0$) is different from the unconstrained optimal solution ($\hat{u}_i \neq 0$), due to Karush–Kuhn–Tucker (KKT) conditions [31].

More intuitively, these three deadlock scenarios are illustrated in Fig. 8.

One way to resolve the deadlock scenarios is to perturb the QP controller so that the robot agents can move around each other when they get stuck. The QP controller needs to be perturbed consistently, because multiple robot agents might still be acting against each other with random perturbations. Inspired by the traffic rule used in transportation to resolve conflicts [32], [33], the following consistent perturbation method is proposed to resolve different deadlocks

1) Type 1 deadlock: As illustrated in Fig. 9(a), the left barrier constraint is relaxed ($k_{\gamma(\text{left})} > 1$) and the right barrier constraint is compressed ($k_{\gamma(\text{right})} < 1$).
2) Type 2 deadlock: As illustrated in Fig. 9(b), $\hat{\mathbf{u}}_i$ is perturbed with $\delta^{\perp} = k_{\delta}\left[\begin{smallmatrix}0 & -1\\ 1 & 0\end{smallmatrix}\right]\hat{u}_i$, which is a normal perturbation to the left of $\hat{u}_i$.
3) Type 3 deadlock: $\mathbf{u}_i = 0$, no perturbation is performed since no admissible perturbation is available.

Note that the online relaxation of the left and right barrier constraints is enabled by the relaxed ZCBF introduced in Section V.

The deadlock resolution strategies are consistent for different types of deadlocks in that a clockwise motion will emerge for all agents involved in the deadlock. Therefore when multiple agents get into a deadlock, they will be perturbed to give way to the agent on the right side as if traffic rules are enforced.

Proposition VII.1: Type 1 and Type 2 deadlocks are resolved with the Decentralized Deadlock Detection Resolution algorithm.

Proof: For Type 1 Deadlock, $u_i = 0 \in \text{vertex}(\mathscr{P}_i)$. When the left barrier constraint is relaxed and the right barrier constraint is compressed, $u_i = 0 \notin \text{vertex}(\mathscr{P}_i')$ as shown in Fig. 9(a). Therefore, the optimal control command to the perturbed QP-based controller is $\bar{\mathbf{u}}_i \neq 0$.

For Type 2 deadlock, $u_i = 0 \in \text{edge}(\mathscr{P}_i)$. Due to KKT conditions, $\hat{\mathbf{u}}_i$ is perpendicular to the edge of $\mathscr{P}_i$, otherwise $u_i = 0$ is not the optimal solution for the QP. When $\delta^{\perp}$, a perturbation normal to $\hat{\mathbf{u}}_i$, is applied, $u_i = 0$ is no longer the optimal control command to the perturbed QP-based controller. Therefore, the actual control command $\bar{\mathbf{u}}_i \neq 0$ as shown in Fig. 9(b).

Algorithm 1: Decentralized_Deadlock_Detection_Resolution.

```
Input: $\mathbf{u}_i, \hat{\mathbf{u}}_i, \mathbf{v}_i$
Output: $\bar{\mathbf{u}}_i, Flag\_lock$
    Initialization : $Flag\_lock = False$
$\delta_{LP} = Decentralized\_LP$
if $\|\hat{\mathbf{u}}_i\| \neq 0$ AND $\|\mathbf{u}_i\| == 0$ AND $\|\mathbf{v}_i\| == 0$ then
    $Flag\_lock = True$
end if
if $Flag\_lock == True$ then
    if $\delta_{LP} > 0$ AND $\mathbf{u}_i \in$ vertex($\mathscr{P}_i$) then
        DType = 1
    else if $\delta_{LP} > 0$ AND $\mathbf{u}_i \in$ edge($\mathscr{P}_i$) then
        DType = 2
    else
        DType = 3
    end if
    switch (DType)
    case 1:
        $\bar{\mathbf{u}}_i$ = Decentralized_QP($k_{\gamma(\text{left})} > 1$,
        $k_{\gamma(\text{right})} < 1$)
    case 2:
        $\bar{\mathbf{u}}_i$ = Decentralized_QP($\hat{\mathbf{u}}_i + \delta^{\perp}$)
    default:
        $\bar{\mathbf{u}}_i = \mathbf{u}_i$
    end switch
else
    $\bar{\mathbf{u}}_i = \mathbf{u}_i$
end if
return $\bar{\mathbf{u}}_i, Flag\_lock$
```

Combining the two cases, the adjusted control command $\bar{\mathbf{u}}_i$ is nonzero without compromising the safety guarantee. Therefore, Type 1 and Type 2 Deadlocks are resolved. ■

In Fig. 10, the decentralized deadlock detection resolution algorithm is validated against different deadlock scenarios. The algorithm successfully perturbed agents away from deadlock scenarios in a consistent way (clockwise rotation around each other emerges in both cases).

The consistent perturbation approach provides solutions to resolve all types of deadlocks except Type 3 deadlocks. In

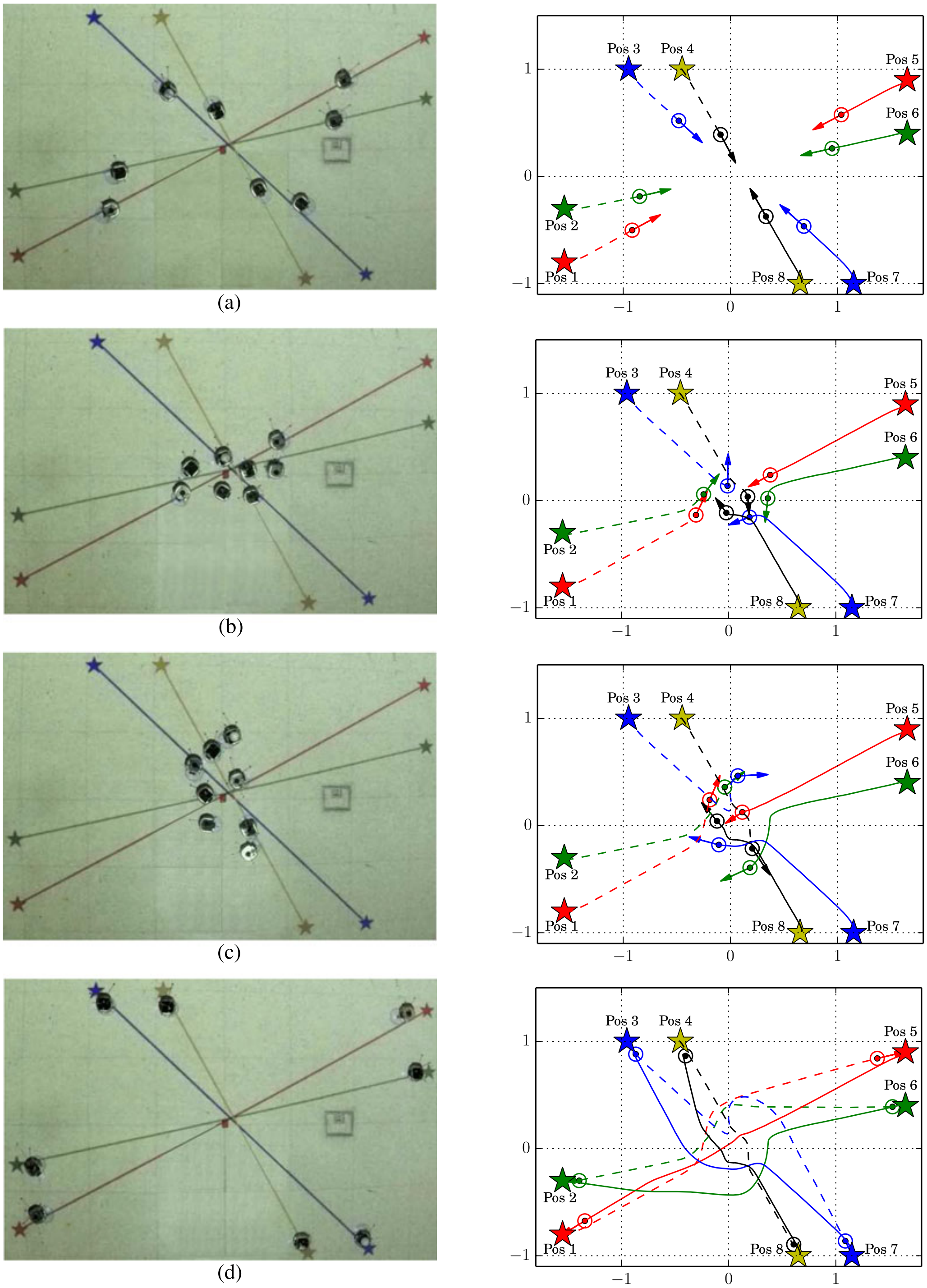


Fig. 11. Experiment of eight Khepera robots swapping positions in a confined workspace. The pictures on the left are taken with an overhead camera. The stars and lines representing the target positions and pairs of swapped positions are projected onto the floor using a projector. The figures on the right illustrate the actual positions, velocities and trajectories of the robots. (a) Agents at 3.0 s. (b) Agents at 6.0 s (c) Agents at 8.0 s. (d) Agent at 15.0 s

addition, livelock might still exist even if deadlock is resolved. This is because the safety barrier certificates in this paper is proposed to provide safety guarantee regardless of the purpose of the nominal controller. As the safety controller is not informed about what the nominal controller is ultimately trying to achieve, livelock becomes a somewhat diffuse concept. One possible idea could be to combine the safety certificates with navigation functions, such that the robots only move in the directions where the navigation function decreases.

VIII. Experimental Results

The decentralized safety barrier certificates were implemented on a multirobot system consisting of multiple

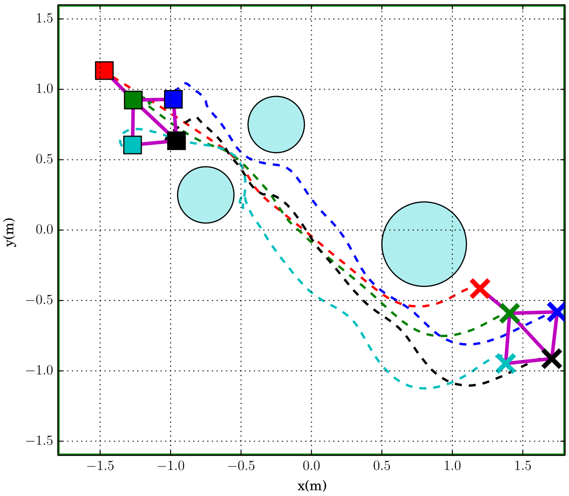


Fig. 12. Plot of actual robot trajectories from the experiment data. The cross and square markers represent the initial and final positions of five Khepera robots, respectively.

Khepera III robots. Two experiments were performed to demonstrate collision avoidance with static and moving obstacles while executing multirobot coordination strategies. A diffeomorphism controller similar to [34] is used to approximate the unicycle dynamics of the robots with double integrator dynamics.

A. Multirobot Swapping Positions in Confined Workspace

The higher level goal of the first experiment was to make all robots in the multirobot system swap positions with each other in a confined workspace, where collisions were very likely to occur. A nominal controller $\hat{\mathbf{u}}_i = -k_1(\mathbf{p}_i - \mathbf{r}_i) - k_2\mathbf{v}_i$ was executed on each robot assuming no collision would happen.

During the experiment, the decentralized safety barrier certificates were wrapped around the nominal controller using the QP based controller (12). Fig. 11 are several snapshots of the robots' positions taken by an overhead camera during the experiment. All agents started on Pos $i,\ i = 1, 2 \ldots, 8$, and moved straightly toward their goal positions following the nominal controller [see Fig. 11(a)]. As they moved closer to the center of the workspace, the safety barrier certificates started modifying the nominal control commands as little as possible to avoid collisions [see Fig. 11(b) and (c)]. Then all robots successfully navigated out of the "crowded" region and swapped their positions [see Fig. 11(d)]. A video of the experiment can be found online [35].

B. Leader-Follower Formation Among Static Obstacles

A multirobot leader-follower formation traveled through an environment populated with static obstacles in the second experiment. The leader-follower formation controller is adapted from a translational invariant formation controller for double integrators in [5].

As shown in Fig. 12, the leader-follower formation of five robots successfully traveled from the initial position to the final position without colliding with each other or the static obstacles. Notice that the shape of the formation was deformed by the safety certificates in the vicinity of obstacles. A video of the experiment is available online [35].

IX. Conclusion

A general framework of minimally invasive collision avoidance for multirobot systems was formally synthesized using control barrier functions. The computation and sensing requirements were reduced significantly by distributing safety barrier certificates to each individual agents and only considering neighboring agents without losing the safety guarantee. Then a series of problems related to safety barrier certificates, i.e., the conservativeness of the certificates, the feasibility of the QP-based controller and deadlock-avoidance, were addressed. The proposed safety barrier certificates were validated through various simulations, and then implemented on a real multirobot system consisting of multiple Khepera robots.

References

[1] T. Balch and R. C. Arkin, "Behavior-based formation control for multi-robot teams," *IEEE Trans. Robot. Autom.*, vol. 14, no. 6, pp. 926–939, Dec. 1998.

[2] J. Cortes, S. Martinez, T. Karatas, and F. Bullo, "Coverage control for mobile sensing networks," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2002, vol. 2, pp. 1327–1332.

[3] W. Burgard, M. Moors, D. Fox, R. Simmons, and S. Thrun, "Collaborative multi-robot exploration," in *Proc. IEEE Int. Conf. Robot. Autom*, 2000, vol. 1, pp. 476–481.

[4] F. Bullo, J. Cortés, and S. Martinez, *Distributed Control of Robotic Networks: A Mathematical Approach to Motion Coordination Algorithms*. Princeton, NJ, USA: Princeton Univ. Press, 2009.

[5] M. Mesbahi and M. Egerstedt, *Graph Theoretic Methods in Multiagent Networks*. Princeton, NJ, USA: Princeton Univ. Press, 2010.

[6] R. C. Arkin, *Behavior-Based Robotics*. Cambridge, MA, USA: MIT Press, 1998.

[7] W. Ren and R. W. Beard, *Distributed Consensus in Multi-Vehicle Cooperative Control*. New York, NY, USA: Springer, 2008.

[8] S. I. Roumeliotis and M. J. Mataric, "Small-world" networks of mobile robots," in *Proc. 17th Nat. Conf. Artif. Intell. 12th Conf. Innovative Appl. Artif. Intell.*, 2000, Paper 1093.

[9] C. Tomlin, G. J. Pappas, and S. Sastry, "Conflict resolution for air traffic management: A study in multiagent hybrid systems," *Proc. IEEE Trans. Automat. Control*, vol. 43, no. 4, pp. 509–521, Apr. 1998.

[10] J. van den Berg, J. Snape, S. J. Guy, and D. Manocha, "Reciprocal collision avoidance with acceleration-velocity obstacles," in *IEEE Int. Conf. Robot. Autom.*, 2011, pp. 3475–3482.

[11] J. Alonso-Mora , A. Breitenmoser, M. Rufli, P. Beardsley, and R. Siegwart, "Optimal reciprocal collision avoidance for multiple non-holonomic robots," in *Distributed Autonomous Robotic Systems*. New York, NY, USA: Springer, 2013, pp. 203–216.

[12] D. Mellinger, A. Kushleyev, and V. Kumar, "Mixed-integer quadratic program trajectory generation for heterogeneous quadrotor teams," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2012, pp. 477–483.

[13] S. Prajna, A. Jadbabaie, and G. J. Pappas, "A framework for worst-case and stochastic safety verification using barrier certificates," *IEEE Trans. Autom. Control*, vol. 52, no. 8, pp. 1415–1428, Aug. 2007.

[14] A. D. Ames, J. W. Grizzle, and P. Tabuada, "Control barrier function based quadratic programs with application to adaptive cruise control," in *Proc. IEEE 53rd Annu. Conf. Decis. Control*, Dec. 2014, pp. 6271–6278.

[15] S. Prajna and A. Rantzer, "Convex programs for temporal verification of nonlinear dynamical systems," *SIAM J. Control Optim.*, vol. 46, no. 3, pp. 999–1021, 2007.

[16] R. Wisniewski and C. Sloth, "Converse barrier certificate theorem," in *Proc. IEEE 52nd Annu. Conf. Decis. Control*, 2013, pp. 4713–4718.

[17] M. Z. Romdlony and B. Jayawardhana, "Stabilization with guaranteed safety using control Lyapunov–barrier function," *Automatica*, vol. 66, pp. 39–47, 2016.

[18] X. Xu, J. W. Grizzle, P. Tabuada, and A. D. Ames, "Correctness guarantees for the composition of lane keeping and adaptive cruise control," 2016. arXiv:1609.06807.

[19] Q. Nguyen and K. Sreenath, "Exponential control barrier functions for enforcing high relative-degree safety-critical constraints," in *Proc. Am. Control Conf. (ACC)*, 2016, pp. 322–328.

[20] U. Borrmann, L. Wang, A. D. Ames, and M. Egerstedt, "Control barrier certificates for safe swarm behavior," in *Proc. IFAC Conf. Anal. Des. Hybrid Syst.*, Oct. 2015, pp. 68–73.

[21] L. Wang, A. D. Ames, and M. Egerstedt, "Multi-objective compositions for collision-free connectivity maintenance in teams of mobile robots," in *Proc. Decisions Control Conf.*, 2016, pp. 2659–2664.

[22] L. Wang, A. D. Ames, and M. Egerstedt, "Safety barrier certificates for heterogeneous multi-robot systems," in *Proc. Amer. Control Conf.*, Jul. 2016, pp. 5213–5218.

[23] J. Van den Berg, M. Lin, and D. Manocha, "Reciprocal velocity obstacles for real-time multi-agent navigation," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2008, pp. 1928–1935.

[24] H. Yamaguchi, "A cooperative hunting behavior by mobile-robot rroops," *Int. J. Robot. Res.*, vol. 18, no. 9, pp. 931–940, 1999.

[25] S. Reveliotis and E. Roszkowska, "Conflict resolution in free-ranging multivehicle systems: A resource allocation paradigm," *IEEE Trans. Robot.*, vol. 27, no. 2, pp. 283–296, Apr. 2011.

[26] 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. Des. Hybrid Syst.*, Oct. 2015, pp. 54–61.

[27] D. Fox, W. Burgard, and S. Thrun, "The dynamic window approach to collision avoidance," *IEEE Robot. Autom. Mag.*, vol. 4, no. 1, pp. 23–33, Mar. 1997.

[28] P. Ogren and N. E. Leonard, "A convergent dynamic window approach to obstacle avoidance," *IEEE Trans. Robot.*, vol. 21, no. 2, pp. 188–195, Apr. 2005.

[29] H. K. Khalil, *Nonlinear Systems*, vol. 3. Englewood Cliffs, NJ, USA: Prentice-Hall, 1996.

[30] B. Morris, M. J. Powell, and A. D. Ames, "Sufficient conditions for the Lipschitz continuity of QP-based multi-objective control of humanoid robots," in *Proc. IEEE 52nd Annu. Conf. Decis. Control*, 2013, pp. 2920–2926.

[31] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, MA, USA: Cambridge Univ. Press, 2004.

[32] L. Perera, J. Carvalho, and C. G. Soares, "Fuzzy logic based decision making system for collision avoidance of ocean navigation under critical collision conditions," *J. Mar. Sci. Technol.*, vol. 16, no. 1, pp. 84–99, 2011.

[33] K. Nagel, D. E. Wolf, P. Wagner, and P. Simon, "Two-lane traffic rules for cellular automata: a systematic approach," *Phys. Rev. E*, vol. 58, no. 2, 1998, Art. no. 1425.

[34] J. R. Lawton, R. W. Beard, and B. J. Young, "A decentralized approach to formation maneuvers," *IEEE Trans. Robot. Autom.*, vol. 19, no. 6, pp. 933–941, Dec. 2003.

[35] Online, "Safety barrier certificates for collision-free multi-robot systems," 2016. [Online]. Available: http://tinyurl.com/htfsmnl

Li Wang (S'16) received the B.S. degree in mechanical engineering from Huazhong University of Science and Technology, Wuhan, China, in 2012 and the M.S. degree in electrical and computer engineering from Clemson University, SC, USA, in 2014. He is currently working toward the Ph.D. degree in electrical and computer engineering at the Georgia Institute of Technology, Atlanta, GA, USA.

His research interests include multirobot systems, control theory, and robotics.

Aaron D. Ames (M'16) received the B.S. degree in mechanical engineering and the B.A. degree in mathematics from University of St. Thomas, St Paul, MN, USA, in 2001. He received the M.A. degree in mathematics and the Ph.D. degree in electrical engineering and computer sciences from University of California, Berkeley, CA, USA, in 2006.

He is the Bren Professor of mechanical and civil engineering and control and dynamical systems at Caltech, Pasadena, CA, USA. Prior to joining Caltech in 2017, he was an Associate Professor in the Woodruff School of Mechanical Engineering and the School of Electrical & Computer Engineering, Georgia Tech, Atlanta, GA, USA. He served as a Postdoctoral Scholar in Control and Dynamical Systems, Caltech from 2006 to 2008, and began his faculty career at Texas A&M University in 2008. His research interests include the areas of robotics, nonlinear control and hybrid systems, with a special focus on applications to bipedal robotic walking both formally and through experimental validation. His lab designs, builds and tests novel bipedal robots, humanoids, and prostheses with the goal of achieving human-like bipedal robotic locomotion and translating these capabilities to robotic assistive devices.

Dr. Ames, at UC Berkeley, received the 2005 Leon O. Chua Award for achievement in nonlinear science and the 2006 Bernard Friedman Memorial Prize in Applied Mathematics, and he received the NSF CAREER award in 2010 and the 2015 Donald P. Eckman Award.

Magnus Egerstedt (S'99–M'00–SM'05–F'12) received the B.A. degree in philosophy from Stockholm University, Stockholm, Sweden, in 1996. He received the M.S. degree in engineering physics and the Ph.D. degree in applied mathematics from Royal Institute of Technology, Stockholm, Sweden, in 1996 and 2000, respectively.

He is the Executive Director for the Institute for Robotics and Intelligent Machines, Georgia Institute of Technology, Atlanta, GA, USA. He is a Professor and the Julian T. Hightower Chair in Systems and Controls in the School of Electrical and Computer Engineering, with secondary appointments in the Woodruff School of Mechanical Engineering, the School of Interactive Computing, and the Guggenheim School of Aerospace Engineering. He was a Postdoctoral Scholar at Harvard University. He is the Director of the Georgia Robotics and Intelligent Systems Laboratory (GRITS Lab), where he conducts research in the areas of control theory and robotics, with particular focus on control and coordination of complex networks, such as multirobot systems, mobile sensor networks, and cyber-physical systems.

Dr. Egerstedt received a number of teaching and research awards, including the Ragazzini Award from the American Automatic Control Council, the Outstanding Doctoral Advisor Award and the HKN Outstanding Teacher Award from Georgia Tech, the Alumnus of the Year Award from the Royal Institute of Technology, and the CAREER Award from the U.S. National Science Foundation.