Control Barrier Certificates for Safe Swarm Behavior

Urs Borrmann * **Li Wang** ** **Aaron D. Ames** *** **Magnus Egerstedt** **

* *Technische Universität München, 80290 München, Germany (e-mail: urs.borrmann@tum.de).*
** *Georgia Institute of Technology, Atlanta, GA 30332 USA (e-mail: liwang,magnus@gatech.edu).*
*** *Texas A&M University, College Station, TX 77843 USA (e-mail: aames@tamu.edu).*

Abstract: Multi-agent robotics involves the coordination of large numbers of robots, which leads to significant challenges in terms of collision avoidance. This paper generates provably collision free swarm behaviours by constructing swarm safety control barrier certificates. The safety barrier, implemented via an optimization-based controller, serves as a low level safety controller formally ensuring the forward invariance of the safe operating set. In addition, the proposed method naturally combines the goals of collision avoidance and interference with the coordination laws in a unified and computationally efficient manner. The centralized version of safety barrier certificate is designed on double integrator dynamic model, and then a decentralized formulation is proposed as a less computationally intensive and more scalable solution. The safety barrier certificate is validated in simulation and implemented experimentally on multiple mobile robots; the proposed optimization-based controller successfully generates collision free control commands with minimal overall impact on the coordination control laws.

1. INTRODUCTION

The literature on multi-robot coordination strategies traditionally focuses on the design of localized coordination rules with provable, global properties such as achieving and maintaining formation, covering areas, or tracking boundaries, e.g., Bullo et al. (2009); Mesbahi and Egerstedt (2010). However, what is actually deployed on teams of robots must also be safe in the sense that collisions are avoided, which typically calls for a secondary, low-level collision-avoidance controller that takes over the operation of the robots as they get too close to each other, e.g., Arkin (1998). The consequence of this construction is that what is tested is, in reality, a combination of the "formally" designed algorithm in conjunction with the "hand-crafted" collision-avoidance controller. Furthermore, as the number of robots increases, the "robot density" increases as well, with the result that the collision-avoidance controller starts to dominate the behavior of the robot team which means that the desired, global properties can no longer be ensured, e.g., Roumeliotis and Mataric (2000).

One solution to the problem of avoiding collisions is to make collision-avoidance an integral part of the coordinated control design. However, this significantly increases the complexity of the design-task and, more importantly, makes the many proposed design tools (see the textbooks Bullo et al. (2009); Mesbahi and Egerstedt (2010); Ren and Beard (2008) for a representative sample of these tools and techniques) no longer applicable. A remedy to this problem is to let the coordinated control design proceed without taking collisions into account and then *ensure that the safety controllers are minimally invasive* in the sense that they do as little as possible unless collisions are absolutely imminent. This idea was pursued in Tomlin et al. (1998) for pairs of kinematic aircraft and, based on global optimal control, hybrid control laws were developed that dictated when the aircraft needed to switch from their current mode of operations to an evasive maneuver in order to avoid collisions. Although elegant, the computational costs associated with solving the full-fledged Hamilton-Jacobi-Bellman Equations quickly become prohibitive when scaling up from two to more agents. Moreover, even for two agents, the problem cannot be solved in real-time; instead, the solution is viewed as a precomputed evasive maneuver that is stored and, subsequently, deployed by the aircraft.

The goal of this paper is to develop controllers that respect desired coordinated control laws as much as possible (in a least-squares sense) while simultaneously guaranteeing collision-free behavior. The key tool for producing these safety-critical controllers us to utilize control barrier certifications to prevent the robots from entering unsafe sets—naturally expressed in a minimally invasive fashion through the use of optimization-based controllers. Building upon the notion of barrier certificates proposed by Prajna et al. (2007), and adopting the control barrier function analogue recently proposed by Ames et al. (2014), safety constraints that prevent collision yield an inequality constraint affine in the control input. A given control law for coordination can then be implemented in the cost of an quadratic program (QP) based control law with constraints given by the safety barrier certificate that enforces collision free behavior. The resulting provably safe algorithm is applied to arbitrarily large teams of mobile robots in both centralized and decentralized representations.

The outline of this paper is as follows: In Section 2 we briefly recall the control barrier certificate construction from Ames et al. (2014) and show that the enabling feature is the inclusion of the barrier as a constraint in an optimization-based controller (as opposed to inclusion in the cost, as is traditionally done; see Panagou et al. (2013)). In Section 3, we show how the barrier certificates can be designed in a centralized manner, i.e., by an external computational unit that has access to the states of all robots in the swarm. In order to reduce the computational burden, it is shown that it is enough to consider robots that are sufficiently close together, i.e., barriers must be considered only between a small subset of agents. This observation is what leads to a decentralized formulation in Section 4, where the individual robots themselves compute their own barrier certificates and corresponding, safe control actions based solely on locally available information. In Section 5, the control laws are experimentally implemented on a team of mobile robots, and the concluding remarks and future directions are the topics of Section 6.

2. BACKGROUND: BARRIER CERTIFICATES

Consider a nonlinear system of the form

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

for $x \in \mathbb{R}^n$ and $u \in U \subset \mathbb{R}^m$, with f and g assumed to be locally Lipschitz. For a given set of $\mathcal{C} \subset \mathbb{R}^n$, the goal is to generate a controller that ensures invariance of the set $\mathcal{C}$, i.e., solutions to (1) that start in $\mathcal{C}$ stay in $\mathcal{C}$ for all time. Establishing invariance of $\mathcal{C}$ can be done through the use of a *barrier function* $B : \mathcal{C} \to \mathbb{R}$ (or barrier certificate; see Prajna et al. (2007)). In particular, if B satisfies the properties:

$$\inf_{x \in \mathrm{Int}(\mathcal{C})} B(x) \geq 0, \qquad \lim_{x \to \partial\mathcal{C}} B(x) = \infty \tag{2}$$

then the question becomes: how does one constraint the behavior of $\dot{B}(x,u)$ to ensure invariance of $\mathcal{C}$?

Conventional design of barrier functions assumed invariant level sets of $\mathcal{C}$, i.e. $\dot{B} \leq 0$ (Tee et al., 2009). Yet this condition is unnecessarily strict, restricting the availability of control inputs to (1). To address this, Ames et al. (2014) recently presented a novel formulation that relaxes the conditions on the change in B to only require that:

$$\dot{B} \leq \frac{\gamma}{B} \tag{3}$$

with $\gamma > 0$. It was shown that this condition still ensures invariance of $\mathcal{C}$, since B is allowed to grow at a rate proportional to the distance of the system from the boundary of $\mathcal{C}$, the set of available control inputs that keep the system safe is greatly increased. The set $\mathcal{C}$ is given by

$$\begin{aligned} \mathcal{C} &= \{x \in \mathbb{R}^n : h(x) \geq 0\}, \\ \partial\mathcal{C} &= \{x \in \mathbb{R}^n : h(x) = 0\}, \\ \mathrm{Int}(\mathcal{C}) &= \{x \in \mathbb{R}^n : h(x) > 0\}, \end{aligned} \tag{4}$$

for a smooth function $h : \mathbb{R}^n \to \mathbb{R}$. Then the condition (3) naturally leads to a notion of a control barrier function:

Definition 1: For the dynamical system (1), a function $B : \mathcal{C} \to \mathbb{R}$ is a control barrier function (CBF) for the set $\mathcal{C}$ defined by (4) for a continuously differentiable function $h : \mathbb{R}^n \to \mathbb{R}$, if there exist locally Lipschitz class $\mathcal{K}$ functions α_1, α_2 such that, for all $x \in \mathrm{Int}(\mathcal{C})$,

$$\frac{1}{\alpha_1(h(x))} \leq B(x) \leq \frac{1}{\alpha_2(h(x))} \tag{5}$$

$$\inf_{u \in U} \left[L_f B(x) + L_g B(x) u - \frac{\gamma}{B(x)} \right] \leq 0 \tag{6}$$

Given a CBF B, consider the set:

$$K_{cbf}(x) = \left\{ u \in U \ : \ L_f B(x) + L_g B(x) u - \frac{\gamma}{B(x)} \leq 0 \right\}$$

wherein it was shown in Ames et al. (2014):

Theorem [Ames et al. (2014)]. Given a set $\mathcal{C} \subset \mathbb{R}^n$ defined by (4) with associated control barrier function B, any Lipschitz continuous controller $u(x) \in K_{cbf}(x)$ for the system (1) renders the set $\mathcal{C}$ forward invariant.

Note that in Ames et al. (2014), control barrier functions were only constructed in the case when h has relative degree 1 (this was extended to higher relative degrees in Hsu et al. (2015)), and applied to adaptive cruise control. In this paper we want to explore the application of CBFs in a multi-agent environment, where each agent is constrained by their own set of CBFs.

3. CENTRALIZED SAFETY BARRIER CERTIFICATES

This section focuses on developing centralized safety barrier certificates that are less intrusive to the nominal controller, but at an expense of central coordination. Then the safety barrier certificates will be decentralized in Section 4 requiring only local information, which leads to a scalable but more conservative solution.

3.1 Problem Formulation

Let $\mathcal{M} = \{1, 2, \ldots, N\}$ be the set of N mobile agents. The dynamics of agent i in the robot swarm is given by

$$\begin{bmatrix} \dot{\mathbf{p}}_i \\ \dot{\mathbf{v}}_i \end{bmatrix} = \begin{bmatrix} 0 & I \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{p}_i \\ \mathbf{v}_i \end{bmatrix} + \begin{bmatrix} 0 \\ I \end{bmatrix} \mathbf{u}_i, \tag{7}$$

where $\mathbf{p}_i \in \mathbb{R}^2$, $\mathbf{v}_i \in \mathbb{R}^2$, and $\mathbf{u}_i \in \mathbb{R}^2$ are the position, velocity, and acceleration of agent i respectively. The velocity and acceleration limits are $\|\mathbf{v}_i\|_p \leq v_{max}$ and $\|\mathbf{u}_i\|_p \leq a_{max}$, where $\|\cdot\|_p$ is vector $p-$norm determined by actual robot model. The relative position between agent i and j is denoted as $\Delta\mathbf{p}_{ij} = \mathbf{p}_i - \mathbf{p}_j$, relative velocity is $\Delta\mathbf{v}_{ij} = \mathbf{v}_i - \mathbf{v}_j$.

The safety constraint of the robot swarm requires that all agents should always keep safety distance D_s from each other. A pairwise safety constraint on relative velocity and relative position:

$$-\frac{\Delta\mathbf{p}_{ij}^T}{\|\Delta\mathbf{p}_{ij}\|}\Delta\mathbf{v}_{ij} \leq \sqrt{2\Delta a_{max}(\|\Delta\mathbf{p}_{ij}\| - D_s)}, \forall i \neq j \tag{8}$$

is proposed to regulate the dynamics of all robot agents within admissible range. This pairwise safety constraint is inspired by the idea of always keeping safety distance while applying the maximum braking force until relative velocity equals zero, which is adopted by many classic collision avoidance literature (see Fox et al. (1997), Ogren and Leonard (2005)). The pairwise robot agent collision avoidance case is a variation of single agent case, where both agents involved are actively reacting to safety threats. As illustrated in Fig. 1, the normal component of relative velocity $\Delta\bar{v} = \frac{\Delta\mathbf{p}_{ij}^T}{\|\Delta\mathbf{p}_{ij}\|}\Delta\mathbf{v}_{ij}$ is considered as the actual component that causes collision, while the tangent component only leads to rotation around each other.

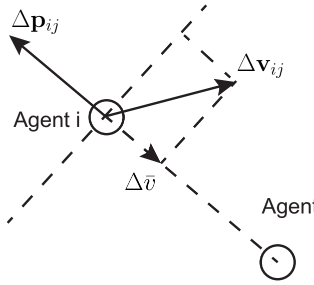


Fig. 1. Relative position and velocity between agent i, j

The maximum relative braking acceleration is $\Delta a_{max} = 2a_{max}$, because both agents are avoiding collision collaboratively in the centralized case. Assuming the relative velocity in the normal direction is $\Delta\bar{v}(t_0)$ at the current time instance, it takes $T_b = \frac{0-\Delta\bar{v}(t_0)}{\Delta a_{max}}$ to reach $\Delta\bar{v}(t_0 + T_b) = 0$, with the maximum relative braking acceleration Δa_{max}. The safety constraint requires that two agents should always keep a safety distance D_s:

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

Since $\Delta\bar{v}(t_0 + t) = \Delta\bar{v}(t_0) + \Delta a_{max} t$, we obtain:

$$\|\Delta\mathbf{p}_{ij}\| - \frac{(\Delta\bar{v})^2}{2\Delta a_{max}} \geq D_s, \quad \forall i \neq j. \tag{9}$$

Note that this constraint is only active when two agents are moving closer to each other ($\Delta\bar{v} < 0$), and no constraint is needed when two agents are moving away from each other ($\Delta\bar{v} \geq 0$). Combining Eqn. (9) with the two cases of $\Delta\bar{v}$ gives the safety constraint presented in Eqn. (8).

We construct a control barrier function B_{ij} candidate from the pairwise safety constraint as follows:

$$h_{ij} = \frac{\Delta\mathbf{p}_{ij}^T}{\|\Delta\mathbf{p}_{ij}\|}\Delta\mathbf{v}_{ij} + \sqrt{2\Delta a_{max}(\|\Delta\mathbf{p}_{ij}\| - D_s)} \tag{10}$$

$$B_{ij} := \frac{1}{h_{ij}}, \qquad \forall i \neq j, \tag{11}$$

where h_{ij} is short for $h_{ij}(\Delta\mathbf{p}_{ij}, \Delta\mathbf{v}_{ij})$, and B_{ij} is short for $B_{ij}(\Delta\mathbf{p}_{ij}, \Delta\mathbf{v}_{ij})$. Note that from h_{ij} we get corresponding sets $\mathcal{C}_{ij}$ as in Eqn. (4), and that B_{ij} satisfies (5) for $\alpha_1(r) = \alpha_2(r) = r$. Therefore, B_{ij} is a CBF if it satisfies the condition: $\dot{B}_{ij} \leq \frac{\gamma}{B_{ij}}$, in which case the forward invariance of the safety operating sets $\mathcal{C}_{ij}$ is guaranteed. $\dot{B}_{ij} \leq \frac{\gamma}{B_{ij}}$ can be reformulated as linear constraints on the control variable $\mathbf{u}$:

$$\begin{aligned} -\Delta\mathbf{p}_{ij}^T\Delta\mathbf{u}_{ij} \leq \frac{\gamma}{B_{ij}} h_{ij}^2 \|\Delta\mathbf{p}_{ij}\| - \frac{(\Delta\mathbf{v}_{ij}^T\Delta\mathbf{p}_{ij})^2}{\|\Delta\mathbf{p}_{ij}\|^2} \\ + \|\Delta\mathbf{v}_{ij}\|^2 + \frac{\Delta a_{max}\Delta\mathbf{v}_{ij}^T\Delta\mathbf{p}_{ij}}{\sqrt{2\Delta a_{max}(\|\Delta\mathbf{p}_{ij}\| - D_s)}}, \quad \forall i \neq j \end{aligned} \tag{12}$$

This safety barrier constraint in Eqn. (12) can be represented as $A_{ij}\mathbf{u} \leq b_{ij}$, where

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

and $\mathbf{u} = [\mathbf{u}_1^T,\ \mathbf{u}_2^T, ..., \mathbf{u}_N^T]^T$ and b_{ij} is the right side of (12).

3.2 Minimally Invasive Collision Avoidance using QPs

In order to develop minimally invasive collision avoidance strategies, we formulate the problem as a quadratic program (QP) that minimizes the difference between the nominal control command $\hat{\mathbf{u}}_i$ and actual control command $\mathbf{u}_i$ subjected to safety barrier constraints. Note that this is a point-wise minimizer, because future coordination control command is unknown to the low level safety program. As discussed in Section 3.1, safety barrier constraints serve as linear constraints on the control variable with the end result being the following QP based controller:

$$\begin{aligned} \mathbf{u}^* = \underset{\mathbf{u}}{\text{argmin}} \quad & J(\mathbf{u}) = \sum_{i=1}^{N} \|\mathbf{u}_i - \hat{\mathbf{u}}_i\|^2 \\ \text{s.t.} \quad & A_{ij}\mathbf{u} \leq b_{ij}, \qquad \forall i \neq j, \\ & \|\mathbf{u}_i\|_\infty \leq a_{max}, \ \ \forall i \in \mathcal{M} \end{aligned} \tag{13}$$

where ∞-norm is adopted in the acceleration limit for simplicity. Note that the online computation of a QP is very efficient, which enables real-time implementation of this algorithm on swarm robotic platform.

3.3 Notion of Neighbor

Centralized barrier certificates lead to increased computation burden and sensing requirement as robot swarm size grows, since a central brain needs to perform all the computation and each agent needs to form a CBF with every other agent in the swarm. A formal notion of neighbor is necessary to reduce computation and sensing requirements. The neighbor set of agent i is defined as

$$\begin{aligned} \mathcal{N}_i = \{j \mid \|\Delta\mathbf{p}_{ij}\| \leq D_\mathcal{N}, D_\mathcal{N} = D_s \\ + \frac{1}{2\Delta a_{max}}(\sqrt[3]{\frac{2\Delta a_{max}}{\gamma}} + \Delta v_{max})^2, j \neq i\} \end{aligned} \tag{14}$$

Since only the distance $D_{ij} = \|\Delta\mathbf{p}_{ij}\|$ between agent i and j is of interest, h_{ij} can be rewritten as:

$$h_{ij} = \dot{D}_{ij} + \sqrt{2\Delta a_{max}(D_{ij} - D_s)}, \qquad \forall i \neq j, \tag{15}$$

Notice that $\dot{D}_{ij} = \sqrt{\Delta\mathbf{p}_{ij} \cdot \Delta\mathbf{p}_{ij}}^{\boldsymbol{\cdot}} = \frac{\Delta\mathbf{p}_{ij}^T}{\|\Delta\mathbf{p}_{ij}\|}\Delta\mathbf{v}_{ij}$ and $D_{ij} \geq D_s$. The following result enables decentralized implementation of swarm control barrier certificates:

Theorem 1. Agent $i \in \mathcal{M}$ only needs to form CBFs with its neighbors defined in Eqn. (14) to guarantee safety.

Proof. For any agent $k \notin \mathcal{N}_i, k \neq i$, i.e., $D_{ik} > D_\mathcal{N}$, we will prove that it is guaranteed to satisfy the safety barrier constraint; therefore, there is no need to form CBF with any agent k outside of the safety disk $D(\mathbf{p}_i, D_\mathcal{N})$.

$$\dot{B}_{ik} \ = \ -\frac{1}{h_{ik}^2}(\ddot{D}_{ik} + \frac{\Delta a_{max}}{\sqrt{2\Delta a_{max}(D_{ik} - D_s)}}\dot{D}_{ik}) \tag{16}$$

Considering the worst case scenario, the largest $\dot{B}_{ik}$ is achieved at $\ddot{D}_{ik} = -\Delta a_{max}$ and $\dot{D}_{ik} = -\Delta v_{max}$, where two agents are heading towards each other at the maximum possible velocity and acceleration.

$$\begin{aligned} h_{ik} &\geq \sqrt{2\Delta a_{max}(D_{ik} - D_s)} - \Delta v_{max} \\ \dot{B}_{ik} &\leq \frac{\Delta a_{max}}{h_{ik}^2}(1 + \frac{\Delta v_{max}}{\sqrt{2\Delta a_{max}(D_{ik} - D_s)}}) \end{aligned}$$

Since $D_{ik} > D_{\mathcal{N}}$, we get $\sqrt{2\Delta a_{max}(D_{ik} - D_s)} > \Delta v_{max}$ and $h_{ij} > \sqrt[3]{\frac{2\Delta a_{max}}{\gamma}}$. The upper bound on $\dot{B}_{ik}$ can be further relaxed to $\dot{B}_{ik} < \frac{2\Delta a_{max}}{h_{ik}^2}$. Combining these inequalities with (11) results in:

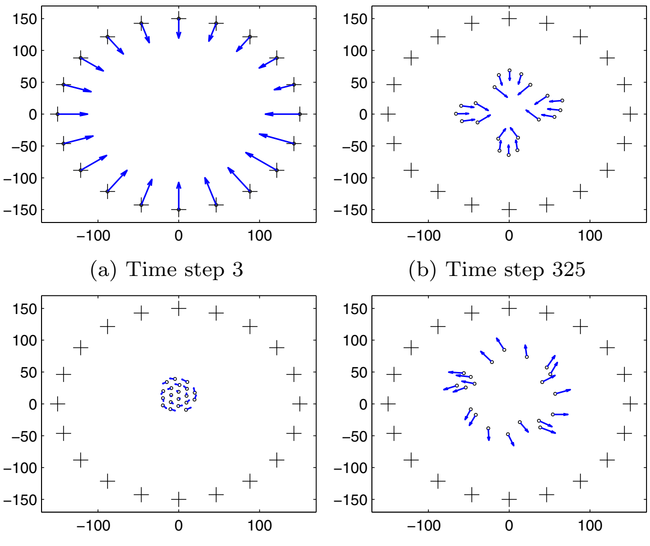
$$\dot{B}_{ik} < \frac{2\Delta a_{max}}{h_{ik}^3}\frac{1}{B_{ik}} < \frac{\gamma}{B_{ik}} \tag{17}$$

By choosing $D_{\mathcal{N}} = D_s + \frac{1}{2\Delta a_{max}}(\sqrt[3]{\frac{2\Delta a_{max}}{\gamma}} + \Delta v_{max})^2$, we can guarantee that the time derivative of the CBF $\dot{B}_{ik}$ is always bounded by $\frac{\gamma}{B_{ik}}$ and so (6) is satisfied and B_{ik} is a valid CBF. Therefore, agents outside of the safety disk $D(\mathbf{p}_i, D_{\mathcal{N}})$ are always considered safe (see Section 2). □

Note that we can design γ appropriately and ensure $D_{\mathcal{N}}$ is smaller than the sensing range to guarantee safety. Although the notion of neighbor is derived for the centralized case, the same idea also applies to decentralized case.

3.4 Simulation results of centralized barrier certificates

This section presents simulation results of centralized barrier certificates applied on a multi-robot test-bed. 20 agents modeled with double integrator dynamics operate with coordination control law $\hat{\mathbf{u}}_i = -k_1(\mathbf{p}_i - \mathbf{r}_i) - k_2\mathbf{v}_i$, which drives it to desired reference position $\mathbf{r}_i$ with final velocity of zero. This nominal controller is implemented in (13) to ensure collision avoidance. All the agents started equally spaced on the cross markers on a circle (Fig. 2a), and moved towards the opposite side of the circle (Fig. 2b). The nominal coordination control law would lead to collision at the center if no collision avoidance strategy is considered. When they got closer (Fig. 2c), the barrier certificates force the agents to keep the required safety distance. Agents rotated around the center to move to the other side of the circle. After the agents passed the "crowded" region (Fig. 2d), they started separating and heading directly toward the goal position.

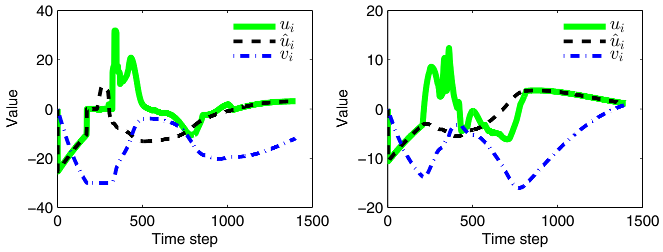


(a) Time step 3 (b) Time step 325

(c) Time step 770 (d) Time step 1000

Fig. 2. Simulation of centralized swarm barrier certificate with 20 mobile robot agents, the arrow represents current velocity of the agent, $D_s = 10$

The safety barrier certificates not only successfully avoid collision between robot agents, but also minimize the interference to a pre-specified coordination control command. As illustrated in Fig. 3, the controller adopts coordination control command when collision is not imminent. As soon as the coordination control command leads to collision, the safety barrier dominates the controller and computes a safe control closest to coordination control law. The



(a) Agent $i = 2$ (b) Agent $i = 4$

Fig. 3. Control variable and velocity of two selected agents

centralized barrier certificates execute efficiently at 50Hz for a 20-agents multi-robot simulation. This enables the safety barrier certificate to be applied in real-time on a swarm robotics platform.

4. DECENTRALIZED SAFETY BARRIER CERTIFICATES

In the previous section, we introduced a centralized swarm control barrier certificate, producing convincing results for multi-agent collision avoidance. Due to its centralized nature, the algorithm (13) must be run on a master node, which might not be feasible in many applications. Even with the notion of neighbours, the amount of computation can still be prohibitive for the centralized case. In order to decentralize the algorithm, we let each agent $i \in \mathcal{M}$ run their own QP on board:

$$\begin{aligned} \mathbf{u}_i^* = \underset{\mathbf{u}_i}{\text{argmin}} \quad & J(\mathbf{u}_i) = \|\mathbf{u}_i - \hat{\mathbf{u}}_i\|^2, \quad \forall i \in \mathcal{M} \\ \text{s.t.} \quad & A_{ij}\mathbf{u} \le b_{ij}, \qquad\qquad j \in \mathcal{N}_i, \\ & \|\mathbf{u}_i\|_\infty \le a_{max}, \end{aligned} \tag{18}$$

where $\mathbf{A}_{ij} = [0, ..., -\Delta\mathbf{p}_{ij}^T, ..., 0]$ requiring only local information. Instead of optimizing over all agents, each node optimizes its own objective, subject to CBF constraints and assuming that other agents keep a constant velocity.

4.1 Assumption on Behaviour of Other Agents

As opposed to the centralized case, there is no communication between the agents, so each agent has to make assumptions on how the other agents will react to impending violations of the safety constraints. Three representative behaviour patterns are considered: actively chasing the current agent, continuing on its own path, or actively avoiding collision. Depending on the assumed pattern, a discrete variable c_{ij} is introduced by scaling Δa_{max} in (10).

$$h_{ij} = \frac{\Delta\mathbf{p}_{ij}^T}{\|\Delta\mathbf{p}_{ij}\|}\Delta\mathbf{v}_{ij} + \sqrt{c_{ij}\Delta a_{max}(\|\Delta\mathbf{p}_{ij}\| - D_s)}, \tag{19}$$

with $c_{ij} \in \{0, 1, 2\}$. For $c_{ij} = 0$, agent i assumes that agent j will show aggressive behaviour towards it; for $c_{ij} = 1$, it assumes neutral behaviour; and for $c_{ij} = 2$, it assumes

agent j will try to avoid the collision as well.
Lower c_{ij} leads to more conservative avoidance behaviour, as shown in Fig. 4. For $c_{12} = c_{21} = 0$, the two agents skirt around each other (solid line). For $c_{12} = c_{21} = 1$, the agents first move towards each other, and then start avoidance behaviours (dash-dot line). If both agents assume collaborative collision avoidance, i.e. $c_{12} = c_{21} = 2$, they move even closer together, before sidestepping each other (dashed line). Another advantage of introducing this decision variable is that a central instance can use it to prioritize one agent over the other.

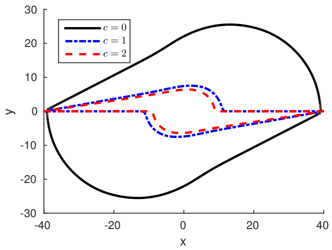
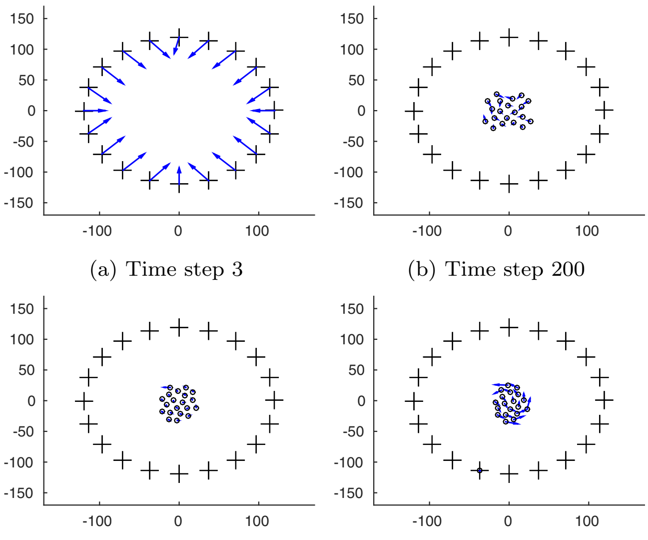


Fig. 4. Comparison of the agents for different assumptions on the reaction of the other agents.

4.2 Simulation Results

The decentralized safety barrier certificate was evaluated using the same nominal controller as introduced in 3.4. In the beginning (Fig. 5b), the agents followed the coordination control command and moved towards the opposite side of the circle. When they got closer (Fig. 5c), the safety barriers force the agents to keep required safety distance. At the same time, they started negotiating a way around each other to reach the other side of the circle. After the agents successfully passed each other (Fig. 5d), they separate and head directly towards the final position.



(a) Time step 3

(b) Time step 200

(c) Time step 600

(d) Time step 850

Fig. 5. Simulation of decentralized swarm barrier certificate with 20 mobile robot agents, the arrow represents current velocity of the agent, $D_s = 10$

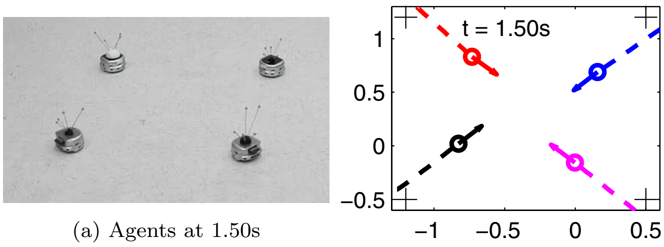
The decentralized safety barrier certificates also successfully avoid collisions. Yet compared with the centralized version, agents are more conservative, choosing slower velocities for the same distances. As there is no central coordination, the paths taken by each individual robot can be far more convoluted than in the central case.

5. EXPERIMENTAL RESULTS

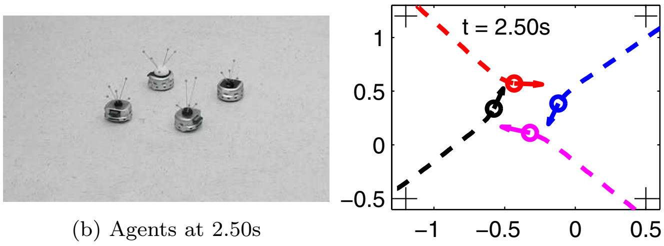
The swarm control barrier certificate, realized via the optimization-based controller, is implemented on a multi-robot platform with four Khepera III robots and Optitrack motion capture system. The nonholonomic robot dynamics is approximated with holonomic double integrator dynamics. Each robot agent executes the same coordination control law as described in section 3.4. Four robot agents start at four corners of a rectangle and move along the diagonal line to the opposite side. The nominal coordination controller would lead to collision at the center.
In order to compare the performance of the centralized (13) and decentralized (18) swarm barrier certificates, both of them were implemented as low level safety controllers on the robot agents. Fig. 6 illustrates the trajectory of four robots with centralized barrier certificates. All agents started heading toward the center as specified by the coordination control law (Fig. 6a). When collision was about to happen, the barrier certificates became active and dominated the nominal controller (Fig. 6b). The optimization-based controller forced the robots to move around the center to maintain safety distance, while moving closer to the goal (Fig. 6c). Ultimately, the robot agents navigated out of the "crowded" region safely.
Decentralized control barrier certificates yielded similar results as the centralized case as shown in Fig. 7. Except that robots are more conservative compared with the centralized case, as they prefer lower velocity manurers when they are close to the boundary of the safe set and take more time (6.5s vs. 5.5s) to finish the same task. Note that the robot agents would very likely enter a *deadlock* situation in a perfectly symmetric case. However, disturbances during the experiment perturbed the robots out of the deadlock region. As shown in Fig. 8, all agents kept a safe distance (red dotted line is the safety distance, $D_s = 0.2m$) throughout the experiment.

6. CONCLUSIONS AND FUTURE WORK

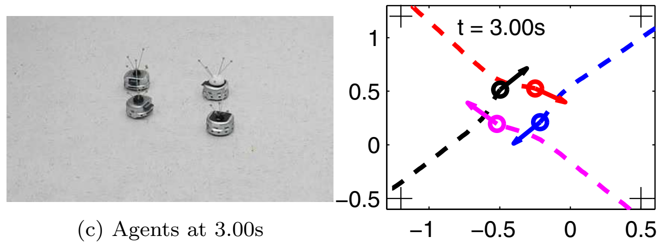
The safety barrier certificates proposed in this paper, as realized by optimization-based controllers, provides a mathematically elegant solution to multi-agent collision avoidance, validated by both simulation and experiment. These results raise several interesting problems for future research. When the agents form some symmetric geometric formations in the simulation, we found constellations in which the interaction between CBFs leads the agents into deadlock. The possibility of deadlock compromises task completion, and therefore the ability to characterize the deadlock regions and use them to synthesize CBFs.
Current version of CBFs requires the dynamics of robots to be affine in control. We have pursued the double integrator dynamics, which is representative yet limited. Most actual robots can be modelled more accurately with nonholonomic dynamics. When approximating nonholonomic dynamics with double integrator dynamics, the safety guarantee brought by barrier certificates might be compromised. As a consequence, it would be important to design a CBF and QP controller directly for actual robot models.



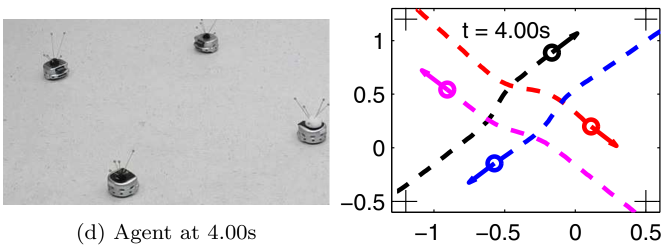
(a) Agents at 1.50s



(b) Agents at 2.50s

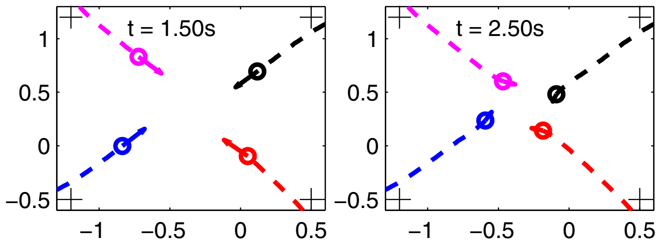


(c) Agents at 3.00s

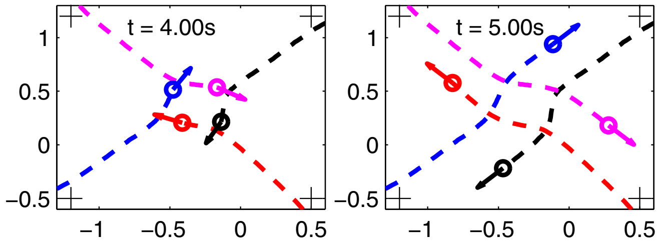


(d) Agent at 4.00s

Fig. 6. 4 robots with centralized swarm barrier certificates. The arrow, circle and dashed line represent current velocity, current position and trajectory of the agent, $D_s = 0.2m, \gamma = 0.1$. Units for X and Y coordinates are in meters. A video can be found Online (2015).

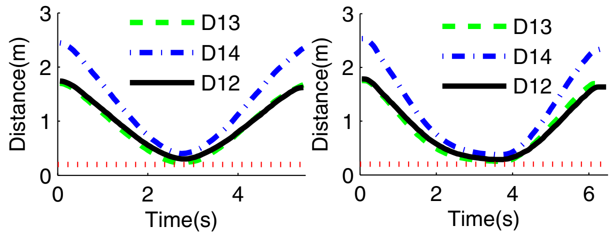


(a) Agents at 1.50s (b) Agents at 2.50s



(c) Agents at 4.00s (d) Agent at 5.00s

Fig. 7. Trajectories of 4 robot agents with decentralized swarm barrier certificate. The arrow, circle and dashed line represent current velocity, current position and trajectory the agent. $D_s = 0.2m, \gamma = 0.1$



(a) Centralized case (b) Decentralized case

Fig. 8. Distance between agent 1 and other agents

REFERENCES

Ames, A.D., Grizzle, J.W., and Tabuada, P. (2014). Control Barrier Function Based Quadratic Programs with Application to Adaptive Cruise Control. In *Decision and Control (CDC), 2014 IEEE 53rd Annual Conference on*, 6271–6278. doi:10.1109/CDC.2014.7040372.

Arkin, R.C. (1998). *Behavior-based Robotics*. MIT press.

Bullo, F., Cortés, J., and Martinez, S. (2009). *Distributed Control of Robotic Networks: a Mathematical Approach to Motion Coordination Algorithms*. Princeton University Press.

Fox, D., Burgard, W., and Thrun, S. (1997). The Dynamic Window Approach to Collision Avoidance. *IEEE Robotics & Automation Magazine*, 4(1), 23–33.

Hsu, S.C., Xu, X., and Ames, A.D. (2015). Control Barrier Function based Quadratic Programs with Application to bipedal robotic walking. In *American Control Conference, 2015. Proceedings of the 2015*. To appear.

Mesbahi, M. and Egerstedt, M. (2010). *Graph Theoretic Methods in Multiagent Networks*. Princeton University Press.

Ogren, P. and Leonard, N.E. (2005). A Convergent Dynamic Window Approach to Obstacle Avoidance. *Robotics, IEEE Transactions on*, 21(2), 188–195.

Online (2015). Barrier certificate for safe swarm behavior. `https://youtu.be/xNvsZ8sQp70`.

Panagou, D., Stipanovic, D.M., and Voulgaris, P.G. (2013). Multi-Objective Control for Multi-Agent Systems Using Lyapunov-Like Barrier Functions. In *Decision and Control (CDC), 2013 IEEE 52nd Annual Conference on*, 1478–1483. IEEE.

Prajna, S., Jadbabaie, A., and Pappas, G.J. (2007). A Framework for Worst-case and Stochastic Safety Verification Using Barrier Certificates. *Automatic Control, IEEE Transactions on*, 52(8), 1415–1428.

Ren, W. and Beard, R.W. (2008). *Distributed Consensus in Multi-vehicle Cooperative Control*. Springer.

Roumeliotis, S.I. and Mataric, M.J. (2000). "Small-World" Networks of Mobile Robots. In *AAAI/IAAI*, 1093.

Tee, K.P., Ge, S.S., and Tay, E.H. (2009). Barrier Lyapunov Functions for the Control of Output-Constrained Nonlinear Systems. *Automatica*, 45(4), 918–927.

Tomlin, C., Pappas, G.J., and Sastry, S. (1998). Conflict Resolution for Air Traffic Management: A Study in Multiagent Hybrid Systems. *Automatic Control, IEEE Transactions on*, 43(4), 509–521.