

# Formally Correct Composition of Coordinated Behaviors Using Control Barrier Certificates \*

Anqi Li<sup>‡</sup>, Li Wang<sup>†</sup>, Pietro Pierpaoli<sup>†</sup>, Magnus Egerstedt<sup>†</sup>

**Abstract**—In multi-robot systems, although the idea of behaviors allows for an efficient solution to low-level tasks, high-level missions can rarely be achieved by the execution of a single behavior. In contrast to this, a sequence of behaviors would provide the requisite expressiveness, but there are no a priori guarantees that the sequence is composable in the sense that the robots can actually execute it. In order to guarantee a provably correct composition of behaviors, Finite-Time Convergence Control Barrier Functions are introduced in this paper to guarantee the terminal configuration of one behavior is a valid initial configuration for the following one. Nominal control inputs prescribed by the behaviors are modified in a minimally invasive fashion, in order to establish the information-exchange network required by the following behavior. The effectiveness of the proposed composition strategy is validated on a team of mobile robots.

## I. INTRODUCTION

Teams of autonomous robots have a wide range of real-world applications including search and rescue [3], environmental surveillance [8] and space exploration [12]. Such high-level missions often require the team to accomplish a set of tasks through collective decision making and coordinated control. For example, the MAGIC 2010 competition [18] addressed a search and rescue scenario, where the robots were expected to complete a series of tasks including exploration, sensor fusion, target localization and human-robot interaction.

Given the complexity of these high-level missions, it is natural to decompose them into a sequence of dedicated coordinated tasks and design *behaviors* which achieve each of the tasks independently. In the context of multi-robot systems, this decomposition is particularly advantageous because (i) there has been a variety of coordinated behaviors developed for elementary tasks such as flocking [16], rendezvous [17], formation control [7], etc, and (ii) the computational burden would quickly become intractable if the complete mission is addressed as a single, highly complex search problem. Hence, it is favorable to leverage well-established and provably correct coordinated behaviors to achieve complicated tasks by composing these behaviors together.

The main challenge associated with such composition is that most behaviors rely on their underlying assumptions to

\*This work was sponsored by Grant No. 1544332 from the U.S. National Science Foundation and Grant No. N66001-17-2-4059 from DARPA.

<sup>‡</sup>Anqi Li is with the School of Interactive Computing, Georgia Institute of Technology, Atlanta, GA 30332, USA. Email: anqi.li@gatech.edu

<sup>†</sup>Li Wang, Pietro Pierpaoli and Magnus Egerstedt are with the School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA 30332, USA. Email: {liwang, pietro.pierpaoli, magnus}@gatech.edu

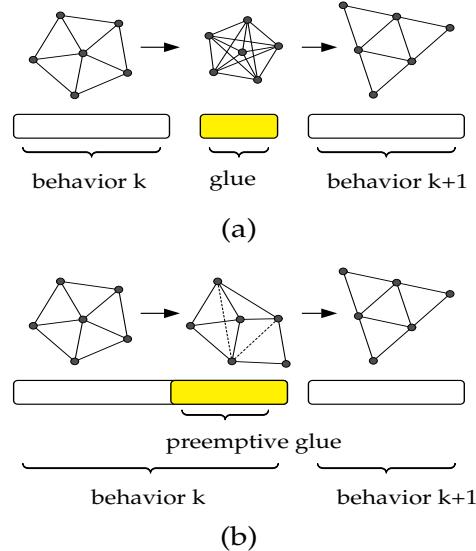


Fig. 1: Examples of different approaches in behavior composition. In (a), a hand-crafted transition behavior is used as a “glue” to ensure the communication graph structure is established for the following behavior. In (b), the approach presented in this paper is shown, where each of the individual behaviors are modified in order to act as a “preemptive glue” in the composition.

function as desired. One of the common assumptions for coordinated behaviors is the presence of sufficiently rich proximity-based information-exchange networks. For example, in order to have the robots rendezvous, the information-exchange network should remain connected [16]. Therefore, in order to complete the high-level mission through a sequence of behaviors, not only should each of the single behaviors be correct, but their assumptions should also be satisfied both at the starting time of the behavior and throughout its execution.

The decomposition of high-level missions into simple tasks is done routinely for single robots, e.g., [2], even with formal guarantees on the overall performance [6]. The idea to decompose complex missions has also been explored in the context of multi-robot systems using models such as Petri Nets [10], finite state automata [13], stochastic processes [15], temporal logic specifications [11], [20], embedded graph grammars [9], [21], and graph process specifications [22]. However, that body of work is typically task spe-

cific. But more importantly, it focuses on conditions and formalisms for characterizing when a sequence of behaviors can be composed together. Although a subset of the work proposes transition behaviors to be inserted between behavior pairs that are not composable [22], such hand-crafted transition behaviors are not always desirable because they are often elaborate and time-consuming. In this paper, we take a different view of this problem by *actively modifying the behaviors* so that they are provably composable, as illustrated in Fig. 1.

In this paper, we focus on providing formal guarantees on composition of coordinated behaviors. Namely, ensuring that once a behavior terminates, the underlying assumptions for the upcoming behavior are satisfied. We develop a variant of control barrier functions (CBFs) [1], [5] called *Finite-Time Convergence Control Barrier Functions*. By encoding the assumptions of each behaviors into Finite-Time Convergence Control Barrier Functions, we can formally guarantee that (i) if the assumptions required by a behavior are satisfied at some point, they will remain satisfied thereafter. On the other side, (ii) if the assumptions are not satisfied initially, they will be satisfied within finite time.

The main contribution of this paper is two-fold. Firstly, a variant control barrier functions is introduced to ensure that the states evolve into the desired subset of the state space within finite time. Secondly, a framework based on control barrier functions is presented to actively modify the behaviors so that the resulting composition of behaviors proves formally correct.

The rest of this paper is organized as follows. Section II formulates the behavior composition problem. Section III introduces finite-time convergence control barrier functions and discusses its application in the context of the behavior composition problem. Section IV leverages control barrier functions to propose a framework for composition of behaviors with theoretical guarantees. At last, experimental results and conclusions are in Section V and Section VI, respectively.

## II. COMPOSITION OF COORDINATED BEHAVIORS

Consider a robotic team consisting of  $N$  mobile robots with integrator dynamics, i.e.,  $\dot{x}_i = u_i$ , where  $x_i \in \mathbb{R}^2$ ,  $u_i \in \mathbb{R}^2$  are the states and control input for robot  $i$ . Let  $x = [x_1^T, \dots, x_N^T]^T$  and  $u = [u_1^T, \dots, u_N^T]^T$  denote the joint state vector and control input of the robotic team, respectively.

The high-level mission of the robotic team is specified through a motion sequence  $\sigma$ , consisting of  $M$  coordinated behaviors. We assume that each coordinated behavior requires a sufficiently rich communication graph to function as designed, which is common for decentralized coordinated behaviors including formation controls, flocking, consensus, etc. The communication graph structure of the robotic team is defined as  $\mathcal{G}(t) = (V, E(t))$ , where  $V = \{1, 2, \dots, N\}$  is the vertex set consisting of  $N$  mobile robots and  $E(t)$  is the edge set at time  $t$ . We assume a proximity-based communication graph, namely, the presence of an edge  $(i, j)$  indicates that robot  $i$  and robot  $j$  are within a communication distance of

$D_c$ , i.e.,

$$(i, j) \in E(t) \iff \|x_i(t) - x_j(t)\| \leq D_c. \quad (1)$$

The communication graph is time-varying due to the motion of robots.

Under the aforementioned assumptions, we use notations similar to the motion description language [14] to specify the sequence of behaviors, namely,

$$\sigma = (\mathcal{U}_1, \mathcal{G}_1, \tau_1), \dots, (\mathcal{U}_M, \mathcal{G}_M, \tau_M), \quad (2)$$

where each coordinated behavior is represented by a tuple  $(\mathcal{U}_k, \mathcal{G}_k, \tau_k)$ . Here,  $\mathcal{U}_k$ ,  $\mathcal{G}_k$ , and  $\tau_k$  are the coordinated controller, required communication graph structure, and starting time of behavior  $k$ , respectively. The required graph structure  $\mathcal{G}_k$  is defined through  $(V, E_k)$ . An edge  $(i, j) \in E_k$  indicates that the edge should be included in the actual communication graph throughout the execution of the behavior, i.e.,  $(i, j) \in E(t)$ ,  $\forall t \in [\tau_k, \tau_{k+1})$  in order to make the behavior function properly. A segment of an example motion sequence is shown in Fig. 2.

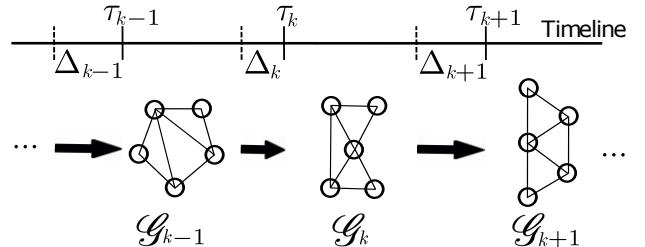


Fig. 2: A segment of a sequence of formation control behaviors for a team of five robots. Each formation control behavior  $k$  starts at  $\tau_k$  and requires a communication graph structure  $\mathcal{G}_k$  in order to function properly. The proposed framework assigns a preparation time  $\Delta_k$ , during which the preceding behavior is modified so as to establish the required communication graph structure by  $\tau_k$ .

Note that although we consider the communication graph structure as the only type of constraints required for the correct functioning of behaviors, one must acknowledge that different types of constraints could be considered as well, e.g. the position of a robot in the global frame of reference, or the relative bearings. In fact, the technique developed in this paper is rather general and can be extended to other constraints as long as it is possible to encode them through finite-time convergence control barrier functions introduced in Section III.

To ensure that the composition of behaviors is valid, we need to make sure that each behavior  $k$  starts with the required graph structure  $\mathcal{G}_k$  assembled, i.e.,  $E_k \subset E(\tau_k)$ . To this end, we develop *finite-time convergence control barrier functions* to provide theoretical guarantees on composition of behaviors.

## III. FINITE-TIME CONVERGENCE CONTROL BARRIER FUNCTIONS

In this section, we present *finite-time convergence control barrier functions* that are essential for valid behavior

composition. We will start with the general formulation that is applicable to any control affine system. Then we will introduce the application of finite-time convergence control barrier functions on establishing a communication graph structure within finite time.

Consider a nonlinear control system in control affine form,

$$\dot{x} = f(x) + g(x)u, \quad (3)$$

where  $x \in \mathcal{D} \subseteq \mathbb{R}^n$  and  $u \in U \subset \mathbb{R}^m$ ,  $f$  and  $g$  are locally Lipschitz continuous. For the sake of simplicity, we assume that (3) is forward complete, i.e. solutions  $x(t)$  are defined for all  $t \geq 0$ .

Without loss of generality, let the desired set for the state of the system to be defined as a super-level set of a continuously differentiable function  $h(x)$ ,

$$\mathcal{C} = \{x \in \mathbb{R}^n : h(x) \geq 0\}. \quad (4)$$

In order to guarantee the correct functioning of each behavior, the corresponding constraints must be satisfied at the behavior starting time, and throughout its entire duration. This, in the context of controller design, can be translated as follows. (i) For solutions that start outside  $\mathcal{C}$ , they should enter  $\mathcal{C}$  within a specific finite interval of time. (ii) The controller should render the desired set forward invariant, i.e., if a solution of (3) starts in the set  $\mathcal{C}$ , it will stay in  $\mathcal{C}$  for all future time.

Control barrier functions [1], [5] are Lyapunov-like functions that are used to provably ensure the forward-invariance of a desired set. Therefore, by definition, the use of control barrier functions automatically guarantees the satisfaction of the forward-invariance requirement. In addition to the forward-invariance property, *Zeroing control barrier functions* [23], a variant of control barrier functions, can be used to guarantee that the state of the system will converge to the boundary of the desired set asymptotically when the initial condition lays outside of the desired set. However, in the context of this paper, since the behavior composition problem is constrained by a sequence of predefined switching times, asymptotic convergence guarantees are not sufficient.

In order to provide theoretical guarantees for both forward invariance and finite-time convergence, we propose a new variant of control barrier functions called *Finite-Time Convergence Control Barrier Functions*.

**Definition III.1.** Given a dynamical system (3) and the set  $\mathcal{C}$  defined by (4) with a continuously differentiable function  $h : \mathbb{R}^n \rightarrow \mathbb{R}$ , if there exist real parameters  $\rho \in [0, 1], \gamma > 0$ , and a set  $\mathcal{C} \subseteq \mathcal{D} \subset \mathbb{R}^n$  such that, for all  $x \in \mathcal{D}$ ,

$$\sup_{u \in U} [L_f h(x) + L_g h(x)u + \gamma \cdot \text{sign}(h(x)) \cdot |h(x)|^\rho] \geq 0, \quad (5)$$

then the function  $h$  is called a *Finite-time Convergence Control Barrier Function (FCBF)* defined on the set  $\mathcal{D}$ .

Now, given a FCBF  $h$ , the set of feasible control inputs is,

$$K(x) = \{u \in U : L_f h(x) + L_g h(x)u + \gamma \cdot \text{sign}(h(x)) \cdot |h(x)|^\rho \geq 0\}. \quad (6)$$

**Proposition III.1.** Given a set  $\mathcal{C} \subset \mathbb{R}^n$  defined by a FCBF  $h(x)$  defined on  $\mathcal{D}$  with  $\mathcal{C} \subseteq \mathcal{D} \subset \mathbb{R}^n$ , and parameters  $\rho \in [0, 1], \gamma > 0$ , any continuous controller  $u : \mathcal{D} \rightarrow \mathbb{R}$  such that  $u \in K(x)$  for the system (3) renders the set  $\mathcal{C}$  forward invariant. Moreover, given the initial state  $x_0 \in \mathcal{D} \setminus \mathcal{C}$ , any continuous controller  $u : \mathcal{D} \rightarrow \mathbb{R}$  such that  $u \in K(x)$  for the system (3) drives the state  $x(t)$  back to  $\mathcal{C}$  within finite time  $T = \frac{1}{\gamma(1-\rho)} |h(x_0)|^{1-\rho}$ .

*Proof.* Consider the Lyapunov candidate function  $V(x) = \max\{-h(x), 0\}$ . The Lyapunov function satisfies,  $V(x) = 0$  for  $x \in \mathcal{C}$ ,  $V(x) > 0$  for  $x \in \mathcal{D} \setminus \mathcal{C}$ , and  $\dot{V}(x(t)) \leq -\gamma V^\rho(x(t))$  for  $t > 0$ .

Assume that  $x_0 \in \mathcal{C}$ . By the comparison lemma [4],  $V(x(t)) = 0$  for all  $t$ . Therefore,  $u \in K(x)$  renders the set  $\mathcal{C}$  forward invariant. Assume that  $x_0 \in \mathcal{D} \setminus \mathcal{C}$ . According to [4], the state  $x$  will converge to the set  $\mathcal{C}$  in finite time within  $T = \frac{1}{\gamma(1-\rho)} |h(x_0)|^{1-\rho}$ .  $\square$

To sum up, we can use finite-time convergence control barrier functions to ensure that the system will enter the desired set within finite time and stay in the set thereafter.

#### A. Communication Graph Barrier Certificates

Here we discuss how to establish a communication graph structure within finite time using finite-time control barrier functions. Assume that the robotic team is required to establish a communication graph  $\mathcal{G}_k = (V, E_k)$  by time  $\tau_k$ . The edge requirement  $(i, j) \in E_k$  can be encoded into a set  $\bar{\mathcal{C}}_{ij}$  defined by a function  $\bar{h}_{ij}$ , where,

$$\begin{aligned} \bar{\mathcal{C}}_{ij} &= \{x \in \mathbb{R}^{2N} : \bar{h}_{ij}(x) \geq 0\}, \\ \bar{h}_{ij}(x) &= D_c^2 - (x_i - x_j)^T (x_i - x_j). \end{aligned} \quad (7)$$

We can use the finite-time convergence barrier function to ensure that the states will enter into  $\bar{\mathcal{C}}_{ij}$  before time  $\tau_k$ . In this case, the time derivative of  $\bar{h}_{ij}(x)$  should satisfy,

$$\dot{\bar{h}}_{ij}(x) + \gamma \cdot \text{sign}(\bar{h}_{ij}(x)) \cdot |\bar{h}_{ij}(x)|^\rho \geq 0. \quad (8)$$

Considering all the edges in  $E_k$ , the communication graph requirement can be encoded into a single set  $\bar{\mathcal{C}}_k$ ,

$$\bar{\mathcal{C}}_k = \bigcap_{(i,j) \in E_k} \bar{\mathcal{C}}_{ij}. \quad (9)$$

By assembling all the communication graph barrier constraints in (8) together, we get the communication graph barrier certificates  $\bar{\mathcal{B}}_k$  for behavior  $k$ ,

$$\begin{aligned} \bar{\mathcal{B}}_k &= \{u \in \mathbb{R}^{2N} : \dot{\bar{h}}_{ij}(x) + \gamma \cdot \text{sign}(\bar{h}_{ij}(x)) \cdot |\bar{h}_{ij}(x)|^\rho \geq 0, \\ &\quad \forall (i, j) \in E_k\}. \end{aligned} \quad (10)$$

By restricting the control input  $u$  to the certificates  $\bar{\mathcal{B}}_k$ , the robotic team is guaranteed to assemble the required communication graph structure  $\mathcal{G}_k$  within finite time. This result is summarized in the following theorem.

**Theorem III.2.** Given a required communication graph structure  $\mathcal{G}_k = (V, E_k)$  and a robotic team with initial state  $x_0$ , any controller  $u(x) \in \bar{\mathcal{B}}_k(x)$  can form

the required graph  $\mathcal{G}_k$  within finite time duration  $T_k = \max_{\{(i,j) \in E_k : x_0 \notin \bar{\mathcal{C}}_{ij}\}} \left\{ \frac{1}{\gamma(1-\rho)} |\bar{h}_{ij}(x_0)|^{1-\rho} \right\}$ .

*Proof.* Consider any pair of robots  $i$  and  $j$  with  $(i, j) \in E_k$ , the controller  $u(x)$  satisfies the pairwise safety barrier constraint in (8). According to Proposition III.1, if  $x_0 \notin \bar{\mathcal{C}}_{ij}$ , the system state  $x$  will be driven into  $\bar{\mathcal{C}}_{ij}$  within time duration  $T_{ij} = \frac{1}{\gamma(1-\rho)} |\bar{h}_{ij}(x_0)|^{1-\rho}$ . Also according to Proposition III.1, if  $x_0 \in \bar{\mathcal{C}}_{ij}$ , the system will stay in  $\bar{\mathcal{C}}_{ij}$ . Note that in this case  $\bar{h}_{ij}(x_0) \geq 0$  and  $T_{ij} = 0$ .

Since every set  $\bar{\mathcal{C}}_{ij}$  will be reached within time duration  $T_{ij} = \frac{1}{\gamma(1-\rho)} |\bar{h}_{ij}(x_0)|^{1-\rho}$  if  $x \notin \bar{\mathcal{C}}_{ij}$ , the overall communication graph structure requirement encoded by  $\bar{\mathcal{C}}_k$  will be satisfied within finite time duration  $T_k = \max_{\{(i,j) \in E_k : x_0 \notin \bar{\mathcal{C}}_{ij}\}} \left\{ \frac{1}{\gamma(1-\rho)} |\bar{h}_{ij}(x_0)|^{1-\rho} \right\}$ .  $\square$

Once the communication graph barrier certificates  $\bar{\mathcal{B}}_k(x)$  are defined, we just need to activate the barrier certificates in an appropriate time  $\Delta_k \geq T_k$  before  $\tau_k$ . The required communication graph structure  $\mathcal{G}_k$  is guaranteed to be constructed before the behavior starting time  $\tau_k$ . The design strategies for the preparation time  $\Delta_k$  are discussed in Section V.

#### IV. THE BEHAVIOR COMPOSITION FRAMEWORK

In this section, we introduce a behavior composition framework anchored by finite-time convergence control barrier functions discussed in the previous section. The main feature of our framework is that there is no explicit transition mode inserted between behaviors. This means that when executing behavior  $k$ , the robots also need to get prepared for behavior  $k+1$ , i.e., all the edges in  $\mathcal{G}_{k+1}$  for behavior  $k+1$  need to be established in the communication graph before time  $\tau_{k+1}$ . We present a framework consisting of a sequence of optimization-based controllers constraint on communication graph barrier certificates. Each of the optimization-based controllers is a minimally invasive controller, i.e. each behavior will be modified only when required by its composition with the following behavior.

##### A. Optimization-based Controller

We propose an optimization-based controller that both fulfills the functionality of each behavior to its greatest extend and respects the communication graph barrier certificates  $\bar{\mathcal{B}}_k(x)$ . We assume that the nominal controller  $\hat{u}_{i,k}$  for each robot  $i$  and behavior  $k$  is designed with existing control techniques and given while executing the behaviors.

For each behavior  $k$ , we divide the time interval during which behavior  $k$  is executed into two phases: before activating  $\bar{\mathcal{B}}_{k+1}$  and after activating  $\bar{\mathcal{B}}_{k+1}$  i.e., after the robots start to get prepared for behavior  $k+1$ . Let  $\Delta_{k+1}$  be the preparation time for behavior  $k+1$ . Then the first phase is formally defined as  $t \in [\tau_k, \tau_{k+1} - \Delta_{k+1}]$ , and the second phase is defined as  $t \in [\tau_{k+1} - \Delta_{k+1}, \tau_{k+1}]$ .

First, consider the phase when the robot are not preparing for the next behavior. In this case, the robots only need to maintain the communication graph structured required by the

current behavior  $k$ . Therefore, the goal is to make the actual control input  $u_i$  as close to the nominal control input  $\hat{u}_{i,k}$  as possible while staying inside the certified space  $\bar{\mathcal{B}}_k$ ,

$$\begin{aligned} u^* = & \arg \min_u \sum_{i=1}^N \|u_i - \hat{u}_{i,k}\|^2 \\ \text{s.t. } & \dot{\bar{h}}_{ij}(x) + \gamma \cdot \text{sign}(\bar{h}_{ij}(x)) \cdot |\bar{h}_{ij}(x)|^\rho \geq 0, \forall (i, j) \in E_k. \end{aligned} \quad (11)$$

Note that the forward invariance property of finite-time convergence control barrier functions is used to ensure the required communication graph structure is preserved.

In the second phase, the robots not only need to maintain the communication graph structure required by behavior  $k$ , they also need to actively assemble the communication graph required by the next behavior. Therefore, the optimization-based behavior has additional constraint provided by  $\bar{\mathcal{B}}_{k+1}$ ,

$$\begin{aligned} u^* = & \arg \min_u \sum_{i=1}^N \|u_i - \hat{u}_{i,k}\|^2 \\ \text{s.t. } & \dot{\bar{h}}_{ij}(x) + \gamma \cdot \text{sign}(\bar{h}_{ij}(x)) \cdot |\bar{h}_{ij}(x)|^\rho \geq 0, \forall (i, j) \in E_k, \\ & \dot{\bar{h}}_{ij}(x) + \gamma \cdot \text{sign}(\bar{h}_{ij}(x)) \cdot |\bar{h}_{ij}(x)|^\rho \geq 0, \forall (i, j) \in E_{k+1}. \end{aligned} \quad (12)$$

In this case, the forward invariance property is used to ensure the required communication graph structure  $\mathcal{G}_k$  is maintained while the finite-time convergence property is used to ensure the required communication graph structure  $\mathcal{G}_{k+1}$  is formed before behavior  $k+1$  starts.

In summary, during the execution of behavior  $k$ , the optimization-based behavior is given by,

$$u = \begin{cases} \arg \min_{u \in \bar{\mathcal{B}}_k} \sum_i \|u_i - \hat{u}_{i,k}\|^2, & \text{if } t \in [\tau_k, \tau_{k+1} - \Delta_{k+1}], \\ \arg \min_{u \in \bar{\mathcal{B}}_k \cap \bar{\mathcal{B}}_{k+1}} \sum_i \|u_i - \hat{u}_{i,k}\|^2, & \text{if } t \in [\tau_{k+1} - \Delta_{k+1}, \tau_{k+1}]. \end{cases} \quad (13)$$

The minimally invasive control input is obtained as a solution to the above Quadratic Programming (QP) problem, where the objective is quadratic in  $u$  and the constraints are linear in  $u$ . Note that QPs can be solved very efficiently, which enables real-time implementation of this framework.

##### B. Overarching Constraints

In addition to the communication graph requirements given by the composability of behaviors, overarching constraints that need to be satisfied throughout the execution of the sequence might be defined as well. For example, it is important to ensure that the trajectories are collision-free and the speeds of the robots are within their limits. In this section, we will use collision-free constraint and speed limits as examples to show how to incorporate overarching constraints into the proposed behavior composition framework.

To produce collision-free motions, we encode collision avoidance as an additional constraint in the optimization-based controller. In this case, we need the collision-free constraint to be a linear constraint with respect to the control input to preserve the efficiency of quadratic programming. Therefore, it is a natural choice to use the safety barrier

certificates developed in [5]. We refer the readers to [5] for detailed derivations and proofs of the safety barrier certificates.

Consider any pair of agents  $i$  and  $j$ , a pairwise safe set can be defined as,

$$\begin{aligned}\mathcal{C}_{ij} &= \{x \in \mathbb{R}^{2N} : h_{ij}(x) \geq 0\}, \\ h_{ij}(x) &= (x_i - x_j)^T (x_i - x_j) - D_s^2, \quad \forall i > j,\end{aligned}\quad (14)$$

The forward invariance of this pairwise safe set can be ensured by the control barrier function  $h_{ij}(x)$ ,

$$\dot{h}_{ij}(x) + \gamma h_{ij}^3(x) \geq 0, \quad (15)$$

where  $\gamma > 0$  is a design parameter. To ensure collision-free trajectories, all pair-wise possible conflict must be addressed. Thus, the team-level collision-free set can be written as,

$$\mathcal{C}_s = \bigcap_{\{i,j \in V : i > j\}} \mathcal{C}_{ij}. \quad (16)$$

Assembling both the communication graph barrier constraints and safety barrier constraints together, we can form the certified admissible control space  $\mathcal{B}_k(x)$ ,

$$\begin{aligned}\mathcal{B}_k(x) &= \bar{\mathcal{B}}_k(x) \cap \\ \{u \in \mathbb{R}^{2N} : h_{ij}(x) + \gamma h_{ij}^3(x) \geq 0, \quad \forall i > j\}.\end{aligned}\quad (17)$$

By constraining the control input  $u$  within the certified space  $\mathcal{B}_k(x)$ , the robots both form the required graph  $\mathcal{G}_k$  within finite time and stay collision-free.

**Theorem IV.1.** *Given a required communication graph  $\mathcal{G}_k = (V, E_k)$  and a robotic team with initial state  $x_0 \in \mathcal{C}_s$ , any continuous controller  $u(x)$  that satisfies the communication graph and safety certificates  $\mathcal{B}_k$  will form the required graph  $\mathcal{G}_k$  within finite time duration  $T_k = \max_{\{(i,j) \in E_k : x_0 \notin \mathcal{C}_{ij}\}} \left\{ \frac{1}{\gamma(1-\rho)} \bar{h}_{ij}(x_0)^{1-\rho} \right\}$  and render the team collision free.*

*Proof.* Since  $u(x) \in \mathcal{B}_k(x) \subseteq \bar{\mathcal{B}}_k(x)$ , the robots will form the required communication graph structure  $\mathcal{G}_k$  within finite time duration  $\Delta_k$  according to Theorem III.2. We can also show that the team is guaranteed collision-free [5].  $\square$

Additionally, we assume that each robot  $i$  can not reach a speed that is beyond its speed limit  $v_{i,\max}$  in each dimension. Hence, we need  $u \in U$ , where,

$$U = \{u \in \mathbb{R}^{2N} : \|u_i\|_\infty \leq v_{i,\max}, \quad \forall i\}. \quad (18)$$

Consequently, the optimization-based controller respecting safety constraints and speed limits is given by,

$$u = \begin{cases} \arg \min_{u \in \mathcal{B}_k \cap U} \sum_i \|u_i - \hat{u}_{i,k}\|^2, & \text{if } t \in [\tau_k, \tau_{k+1} - \Delta_{k+1}), \\ \arg \min_{u \in \mathcal{B}_k \cap \mathcal{B}_{k+1} \cap U} \sum_i \|u_i - \hat{u}_{i,k}\|^2, & \text{if } t \in [\tau_{k+1} - \Delta_{k+1}, \tau_{k+1}]. \end{cases} \quad (19)$$

Note that since all the constraints in (19) are linear, the resulting controller can still be solved efficiently through quadratic programming in real-time.

## V. EXPERIMENTAL RESULTS

In this section, we validate the behavior composition framework through both simulation tasks as well as robotic implementation. Towards the end of this section, we discuss some practical issues including choices of parameters and extensions for more complicated behavior compositions.

### A. Simulation Results

We evaluate the framework presented in the previous sections by simulating a team of 6 agents performing an environmental exploration task. In particular, robots were tasked with reaching a region of interest, denoted by the blue ellipsis, and enter it. As discussed in Section I, the overall mission is divided in a sequence of behaviors to be performed sequentially by the robots. With reference to Fig. 3, the sequence of behaviors was composed by an initial lattice formation control, where robots gathered in the proximity of their initial positions forming a regular lattice. After that, a leader-follower protocol was employed, and agent number 1 moved the team to a pre-assigned point close to the region of interest. Once the region of interest had been approached, a formation control behavior moved the agents into a cross-shaped formation. In the final stage, through leader-follower protocol, agent number 6 moved the team inside the region. The colored bar at the bottom of each figure represents the time progress of the whole mission; each behavior is represented by a light color, while darker colors denote the time during which the robots prepared for the following behavior.

### B. Robotic Implementations

In this section we present results from experiments conducted on the Robotarium, a remotely accessible robotic test-bed [19]. In the experiment, the robotic team executed a sequence of behaviors consisting of a cyclic pursuit behavior, a lattice formation, a line formation and a square formation. The correctness of behavior composition was formally guaranteed by the framework introduced in this paper whereas safety of the robots was encoded by the safety barrier certificates [5] as an overarching constraint. Fig. 4 shows the snapshots taken during the experiment. The robots were initialized with a cycle formation and executed the cyclic pursuit behavior. Next, the robots modified the cyclic pursuit behavior to establish the required edges for the lattice formation (Fig. 4a), and then formed the lattice formation (Fig. 4b). After that the robot formed a line formation and a square formation successively in the similar manner (Fig. 4c–4f). Fig. 4 shows that the edges required by each behavior (red) remained to be connected throughout the execution, and the additional edges required by the following behavior (blue) were always established during the preparation period.

In the supplementary materials, we present the video of an additional experiment with a more complex sequence of behaviors where the robots were designed to form two sub-teams, accomplish different tasks in parallel, merge into a single team, and execute behaviors as a single team.

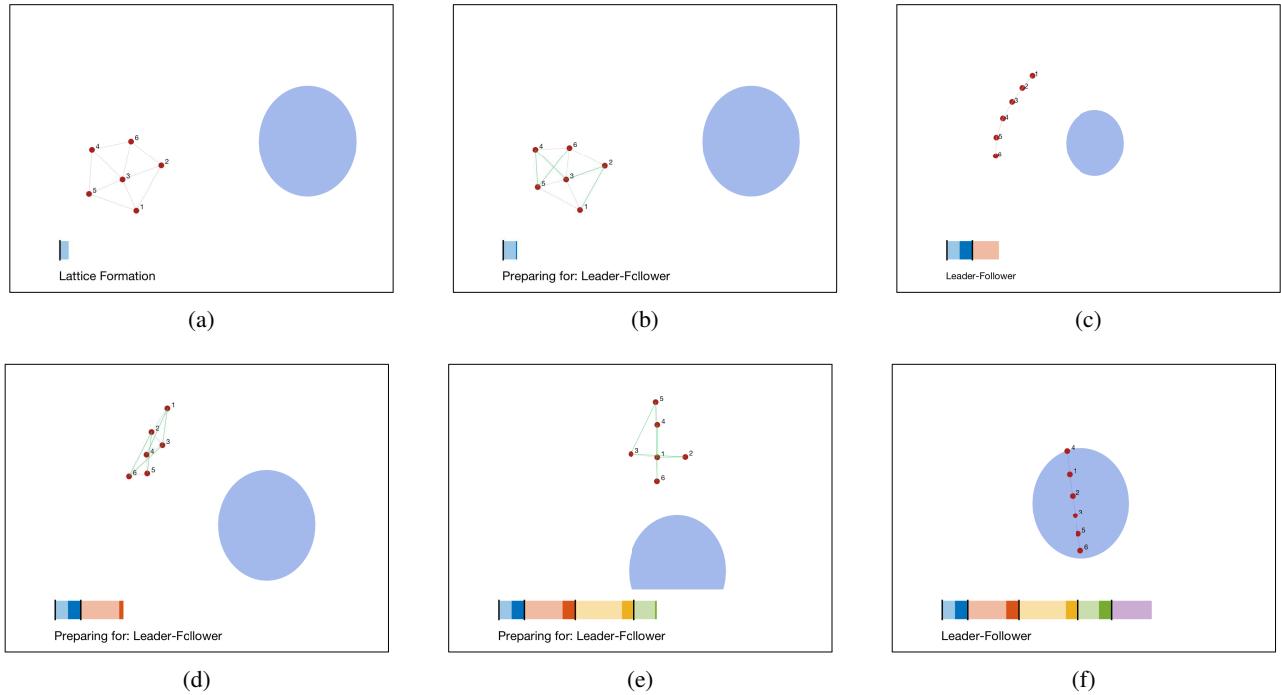


Fig. 3: A sequence of behaviors representing an environmental exploration task. A team composed by six robots was tasked with approaching a region of interest, denoted by the blue ellipse. Robots initially gathered under a lattice formation control; after that a sequence of leader-follower behavior and formation control moved the agents close to the region of interest and prepared the team for entering in it. The mission was concluded with a final leader-follower behavior, where agent number 6 led the team inside the region. In figure, gray lines denote the edges of the graph corresponding to the current behavior, while green lines represent the additional edges required by the next behavior.

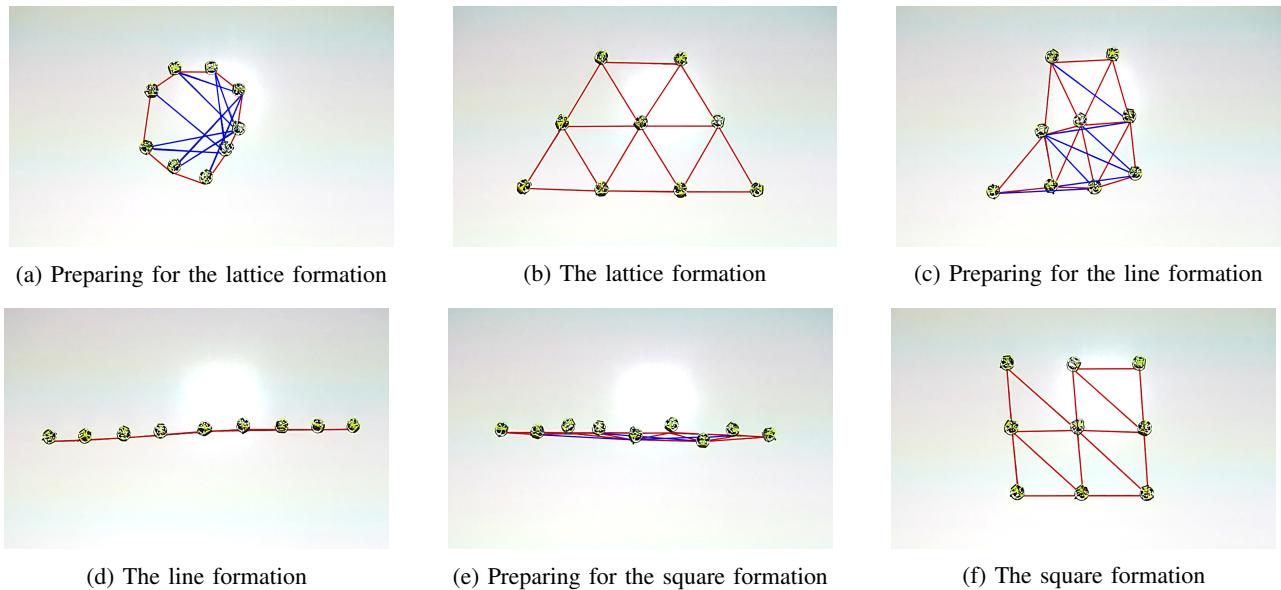


Fig. 4: Experiment results for a team of nine mobile robots executing a sequence of behaviors on the Robotarium testbed. The graphics were superimposed on the robot arena by an overhead projector whereas the colors of the graphics are augmented in the figures for the purpose of visualization. The edges required by each behavior remained connected (red) throughout its execution, and the additional edges (blue) required by the following behavior were established during the preparation time.

### C. Discussions

Here, we discuss some practical considerations when implementing the framework on the robotic teams, including how to determine preparation time  $\Delta_k$  for each behavior, and how to apply this framework to more complicated behavior composition beyond sequential composition.

There are two fundamental ways to design preparation time  $\Delta_k$  for each coordinated behavior  $k$ . The first one is called *event-triggered* where a fixed preparation time  $\Delta_k$  is specified ahead of time. The finite-time convergence control barrier certificates are automatically activated at  $t = \tau_k - \Delta_k$ , where the parameter  $\gamma$  in the control barrier certificates is computed such that,

$$T_k = \max_{\{(i,j) \in E_k : x(t) \notin \bar{\mathcal{C}}_{ij}\}} \left\{ \frac{1}{\gamma(1-\rho)} |\bar{h}_{ij}(x(t))|^{1-\rho} \right\} \leq \Delta_k. \quad (20)$$

The second approach is called *self-triggered*. In this case, we specify a fixed  $\gamma$ , and at each time, we check whether the finite-time convergence control barrier certificates need to be activated by the worst case scenario in the domain  $\mathcal{D}$ , i.e., we activate the finite-time convergence control barrier certificates at time  $t$  when,

$$\tau_k - t \leq \max_{x \in \mathcal{D}} \max_{\{(i,j) \in E_k : x \notin \bar{\mathcal{C}}_{ij}\}} \left\{ \frac{1}{\gamma(1-\rho)} |\bar{h}_{ij}(x)|^{1-\rho} \right\} + \varepsilon, \quad (21)$$

where  $\varepsilon > 0$  is a small positive scalar. Note that this way of designing the preparation time requires us to have additional knowledge about the domain of the joint state  $\mathcal{D}$ , for example, the boundary of the field.

An additional remark is that the framework is applicable beyond simple sequential composition of behaviors. One key observation of that is that the required communication graph structure need not be connected. Therefore, we can divide the robotic team into a number of sub-teams, and design nominal controllers and control barrier certificates for each of the sub-teams. In this case, sequential and parallel composition of behaviors can interleave with each other so that sub-teams of robots are formed and merged during the execution to achieve more complicated missions.

## VI. CONCLUSIONS

The paper presented a framework that provides theoretical guarantees on composition of coordinated behaviors for teams of autonomous robots. To achieve provably correct composition, finite-time convergence control barrier functions were proposed so the the robots can assemble the required communication graph structure within a specified finite interval of time. The control barrier functions-based framework actively modifies the behaviors in the minimally invasive way so that the initial configuration of each single behavior is valid when executed in sequence. Simulation results and experiments on a team of mobile robots validated the effectiveness of the proposed behavior composition framework.

## REFERENCES

- [1] A. D. Ames, J. W. Grizzles, and P. Tabuada. Control barrier function based quadratic programs with application to adaptive cruise control. In *53rd IEEE Conference on Decision and Control*, pages 6271–6278. IEEE, 2014.
- [2] R. C. Arkin. *Behavior-based Robotics*. MIT press, 1998.
- [3] J. L. Baxter, E. Burke, J. M. Garibaldi, and M. Norman. Multi-robot search and rescue: A potential field based approach. In *Autonomous robots and agents*, pages 9–16. Springer, 2007.
- [4] S. P. Bhat and D. S. Bernstein. Finite-time stability of continuous autonomous systems. *SIAM Journal on Control and Optimization*, 38(3):751–766, 2000.
- [5] U. Borrman, L. Wang, A. D. Ames, and M. Egerstedt. Control barrier certificates for safe swarm behavior. In *Analysis and Design of Hybrid Systems, 2015 IFAC Conference on*. IEEE, Oct 2015.
- [6] R. R. Burridge, A. A. Rizzi, and D. E. Koditschek. Sequential composition of dynamically dexterous robot behaviors. *The International Journal of Robotics Research*, 18(6):534–555, 1999.
- [7] T. Eren, W. Whiteley, B. D. Anderson, A. S. Morse, and P. N. Belhumeur. Information structures to secure control of rigid formations with leader-follower architecture. In *American Control Conference, 2005. Proceedings of the 2005*, pages 2966–2971. IEEE, 2005.
- [8] G. A. Hollinger, S. Choudhary, P. Qarabaqi, C. Murphy, U. Mitra, G. S. Sukhatme, M. Stojanovic, H. Singh, and F. Hover. Underwater data collection using robotic sensor networks. *IEEE Journal on Selected Areas in Communications*, 30(5):899–911, 2012.
- [9] E. Klavins, R. Ghrist, and D. Lipsky. Graph grammars for self assembling robotic systems. In *Robotics and Automation, 2004. Proceedings. ICRA'04. 2004 IEEE International Conference on*, volume 5, pages 5293–5300. IEEE, 2004.
- [10] E. Klavins and D. E. Koditschek. A formalism for the composition of concurrent robot behaviors. In *Robotics and Automation, 2000. Proceedings. ICRA'00. IEEE International Conference on*, volume 4, pages 3395–3402. IEEE, 2000.
- [11] M. Kloetzer and C. Belta. Temporal logic planning and control of robotic swarms by hierarchical abstractions. *IEEE Transactions on Robotics*, 23(2):320–330, 2007.
- [12] J. Leitner. Multi-robot cooperation in space: A survey. In *Advanced Technologies for Enhanced Quality of Life, 2009. AT-EQUAL'09*, pages 144–151. IEEE, 2009.
- [13] A. Marino, L. Parker, G. Antonelli, and F. Caccavale. Behavioral control for multi-robot perimeter patrol: A finite state automata approach. In *Robotics and Automation, 2009. ICRA'09. IEEE International Conference on*, pages 831–836. IEEE, 2009.
- [14] P. Martin and M. B. Egerstedt. Hybrid systems tools for compiling controllers for cyber-physical systems. *Discrete Event Dynamic Systems*, 22(1):101–119, 2012.
- [15] N. Napp and E. Klavins. A compositional framework for programming stochastically interacting robots. *The International Journal of Robotics Research*, 30(6):713–729, 2011.
- [16] R. Olfati-Saber, J. A. Fax, and R. M. Murray. Consensus and cooperation in networked multi-agent systems. *Proceedings of the IEEE*, 95(1):215–233, 2007.
- [17] R. Olfati-Saber and R. M. Murray. Consensus problems in networks of agents with switching topology and time-delays. *IEEE Transactions on automatic control*, 49(9):1520–1533, 2004.
- [18] E. Olson, J. Strom, R. Morton, A. Richardson, P. Ranganathan, R. Goeddel, M. Bulic, J. Crossman, and B. Marinier. Progress toward multi-robot reconnaissance and the magic 2010 competition. *Journal of Field Robotics*, 29(5):762–792, 2012.
- [19] D. Pickem, P. Glotfelter, L. Wang, M. Mote, A. Ames, E. Feron, and M. Egerstedt. The robotarium: A remotely accessible swarm robotics research testbed. *arXiv preprint arXiv:1609.04730*, 2016.
- [20] I. Saha, R. Ramaithitima, V. Kumar, G. J. Pappas, and S. A. Seshia. Automated composition of motion primitives for multi-robot systems from safe ltl specifications. In *Intelligent Robots and Systems (IROS 2014), 2014 IEEE/RSJ International Conference on*, pages 1525–1532. IEEE, 2014.
- [21] B. Smith, A. Howard, J.-M. McNew, J. Wang, and M. Egerstedt. Multi-robot deployment and coordination with embedded graph grammars. *Autonomous Robots*, 26(1):79–98, 2009.
- [22] P. Twu, P. Martin, and M. Egerstedt. Graph process specifications for hybrid networked systems. *IFAC Proceedings Volumes*, 43(12):65–70, 2010.

CONFIDENTIAL. Limited circulation. For review only.

- [23] X. Xu, P. Tabuada, J. W. Grizzle, and A. D. Ames. Robustness of control barrier functions for safety critical contro. In *Analysis and Design of Hybrid Systems, 2015 IFAC Conference on*. IEEE, Oct 2015.