Appeared in NecSys 2019

Distributed dynamic containment control over a strongly connected and weight-balanced digraph

Yi-Fan Chung and Solmaz S. Kia

Department of Mechanical and Aerospace Engineering, University of California, Irvine, Irvine, CA 92697 USA (e-mail: {yfchung,solmaz}@uci.edu).

Abstract: This paper proposes a distributed containment control solution for a group of communicating mobile agents that aim to track the convex hull spanned by a group of moving leaders with unknown dynamics. The communication topology of the mobile agents is described by a strongly connected and weight-balanced directed graph. In our problem setting the agents can communicate in discrete-time and also detect the leaders in specific sampling times. Our next contribution in this paper is to show how a group of unicycle robots can use our proposed containment control algorithm to track the convex hull of their jointly monitoring mobile leaders. In our proposed framework, the unicycle robots have continuous-time dynamics but communicate with each other in discrete-time fashion. We demonstrate our results through a numerical example.

Keywords: Distributed containment control, Dynamic average consensus, Unicycle robots.

1. INTRODUCTION

We consider the problem of the distributed containment control for a group of communicating mobile agents. The objective is to drive a group of mobile agents into a region that is enclosed by another set of agents, which we refer to as leaders. A prime application example for containment problem is when a group of communicating robots follows a group of leader robots that have the ability to avoid obstacles when they transport through a hazardous area (Liu et al. (2012b)). Other potential applications include formation control for UAVs (Wang et al. (2007)) and underwater vehicles (Hou and Cheah (2011)), hazardous material delivery (Engelberger (2012)) and mobile sensor networks (Iyengar and Brooks (2016)).

Containment control has been of interest in the literature in recent years. Dimarogonas et al. (2006) proposed a control for a group of unicycle agents that drove the leaders to a formation and the followers to the convex hull. Ji et al. (2008) designed a containment control for single integrator agents based on the theory of partial differential equations and a stop-go policy for the moving leaders. For double integrator agents, the containment control problem was considered in Li et al. (2012) and Wang et al. (2014). The algorithms propose in Dimarogonas et al. (2006); Ji et al. (2008); Li et al. (2012); Wang et al. (2014) require the communication topology of the followers to be an undirected graph. In some situations, the interaction topology among agents may be a directed graph due to realistic communication restrictions. Accordingly, Cao et al. (2012) extended the work of Ji et al. (2008) and focused on switching directed interaction topology among agents. The agents with more complex dynamics under directed interaction graphs was investigated in Liu et al. (2012a) for general linear dynamics and Mei et al. (2012) for nonlinear Lagrangian dynamics, respective. Nevertheless, the work mentioned so far needs to continuously exchange information among the network, which may not be realistic in practice. In Galbusera et al. (2013), a group of agents with discrete-time dynamics was considered and the containment control problem was solved by a discrete-time scheme of hybrid model predictive control. However, it is preferable to having a controller for continuous-time dynamics agents with discrete-time communications with their neighbors. Liu et al. (2012b) proposed a containment control based on periodic sampled-data for agents with continuous-time single and double integrator dynamics over a directed graph. Furthermore, aperiodic sampled-based containment controls for double integrator and continuous-time linear agents were studied by Liu et al. (2014) and Liu et al. (2015), respectively. It is worth to mention that all of the work mentioned above is considering the homogeneous network systems, that is all agents are with an identical dynamics. Zheng and Wang (2014) considered a heterogeneous multi-agent system that the leader group and the follower group could be with either single or double integrator dynamics, respectively. Haghshenas et al. (2015) studied a more general case that the follower agents were with different linear dynamics and the leader group has the same passive linear dynamics (linear dynamics without input). However, for the work of Zheng and Wang (2014) and Haghshenas et al. (2015), the leaders are still homogeneous and their dynamics are assumed to be known to the followers.

In this paper, we propose a distributed containment control algorithm for a group of agents, which communicate over a strongly connected and weight-balanced directed graph. The agents jointly detect a group of moving leaders in a periodic sampling time. Each agent detects a subset (could be empty) of the leaders and computes the average center of the subset. Then, the discrete-time containment control algorithm based on the dynamic average consensus algorithm is developed to track a point in the convex hull of the observed leaders. Therefore, even though some of the agents do not observe any leaders, they can still track the convex hull. Note that, there is no assumption about the leaders' dynamics (they can be heterogeneous) and the agents only measure the positions of the leaders. Next, we show that our proposed containment control algorithm can

⋆ This work is supported by NSF award IIS-SAS-1724331.

be further applied to a group of unicycle robots to track the convex hull of the mobile leaders. The control scheme combines the discrete-time containment control algorithm as an observer to estimate the position of the point in the convex hull and a local finite-time control to track their estimate, such that the control scheme can drive the robots to the convex hull in time. In our proposed framework, the unicycle robots have continuous-time dynamics but communicate with each other in discrete-time fashion. A numerical example demonstrates the efficiency of the proposed solution.

The rest of this paper is organized as follows. Section 2 introduces our basic notation, graph-theoretic definitions and notions and reviews the dynamic average consensus algorithm, which we use in our developments. Section 3 gives our problem definition and objective statement. Section 4 presents our main result on design of a distributed containment control algorithm. Section 5 applies the containment control scheme for unicycle robots. Section 6 gives a numerical example to demonstrate our results. Section 7 concludes the results of this paper.

2. NOTATIONS AND PRELIMINARIES

Notation: We let $\mathbb{R}$, $\mathbb{R}_{>0}$, $\mathbb{R}_{\geq 0}$, $\mathbb{Z}$, and $\mathbb{Z}_{\geq a}$ denote the set of real, positive real, non-negative real, integer, and integer numbers greater than $a \in \mathbb{Z}$, respectively. The transpose of a matrix $\mathbf{A} \in \mathbb{R}^{n\times m}$ is $\mathbf{A}^\top$. For $\mathbf{s} \in \mathbb{R}^d$, $\|\mathbf{s}\| = \sqrt{\mathbf{s}^\top \mathbf{s}}$ denotes the standard Euclidean norm. For a given set of points $\mathcal{X} = \{\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_M\}$ in a Euclidean space, their convex hull is $\mathrm{Co}(\mathcal{X}) = \{\mathbf{q} \in \mathbb{R}^n | \mathbf{q} = \sum_{j=1}^{M} \alpha_j \mathbf{x}_j, \alpha \geq 0, \sum_{j=1}^{M} \alpha_j = 1\}$, which is the smallest convex set containing all the points in $\mathcal{X}$. In a network of N agents, to distinguish and emphasis that a variable is local to an agent $i \in \{1, \dots, N\}$, we use superscripts, e.g., $\mathbf{x}^i$ is the local variable of agent i. Moreover, if $\mathbf{r}^i \in \mathbb{R}^{n^i}$ is a variable of agent $i \in \mathcal{V} = \{1, \cdots, N\}$, the aggregated $\mathbf{r}^i$'s of the network is the vector $\mathbf{r} = [\{\mathbf{r}^i\}_{i\in\mathcal{V}}] = [\mathbf{r}^{1\top}, \cdots, \mathbf{r}^{N\top}]^\top \in \mathbb{R}^m$, $m = \sum_{i=1}^{N} n^i$.

Graph theoretic notations and definitions: Here we review our graph related notations and relevant definitions and concepts from graph theory following Bullo et al. (2009). A *digraph*, is a triplet $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathbf{A})$, where $\mathcal{V} = \{1, \dots, N\}$ is the *node set* and $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ is the *edge set*, and $\mathbf{A} = [\mathsf{a}_{ij}] \in \mathbb{R}^{N\times N}$ is the *adjacency* matrix of the graph defined according to $\mathsf{a}_{ij} = 1$ if $(i, j) \in \mathcal{E}$ and $\mathsf{a}_{ij} = 0$, otherwise. An edge (i, j) from i to j means that agent j can send information to agent i. Here, i is called an *in-neighbor* of j and j is called an *out-neighbor* of i. A *directed path* is a sequence of nodes connected by edges. The *out-degree* of a node i is $\mathsf{d}^i_{\text{out}} = \Sigma_{j=1}^{N} \mathsf{a}_{ij}$. We let $\mathsf{d}^{\max} = \max\{\mathsf{d}^i_{\text{out}}\}_{i=1}^{N}$. The out-degree matrix of a graph is $\mathbf{D}_{\text{out}} = \mathrm{Diag}(\mathsf{d}^1_{\text{out}}, \mathsf{d}^2_{\text{out}}, \cdots, \mathsf{d}^N_{\text{out}})$. The *(out-) Laplacian* matrix is $\mathbf{L} = \mathbf{D}_{\text{out}} - \mathbf{A}$. Note that $\mathbf{L}\mathbf{1}_N = 0$. A weighted digraph $\mathcal{G}$ is weight-balanced if and only if $\mathbf{1}_N^\top \mathbf{L} = 0$. Based on the structure of $\mathbf{L}$, at least one of the eigenvalues of $\mathbf{L}$ is zero and the rest of them have nonnegative real parts.

Average dynamic consensus algorithm: In our developments we will use a dynamic average consensus algorithm (see Kia et al. (2019)) as described in the lemma below .

Lemma 1. (Dynamic average consensus algorithm). Let $\mathcal{G}(\mathcal{V}, \mathcal{E})$ be a strongly connected and weight-balanced digraph of N agents. Assume each agent $i \in \mathcal{V}$ has access to a dynamic input $\mathbf{r}^i(k) = \mathbf{r}^i(t_k)$ at time $t_k = k\delta$, $\delta \in \mathbb{R}_{>0}$, $k \in \mathbb{Z}_{\geq 0}$. For $\delta \in (0, \beta^{-1}(\mathsf{d}^{\max})^{-1})$, where $\beta \in \mathbb{R}_{>0}$, if each agent $i \in \mathcal{V}$ implements

$$\mathbf{p}^i(k+1) = \mathbf{p}^i(k) + \delta\beta \sum_{j=1}^{N} \mathsf{a}_{ij}(\mathbf{x}^i(k) - \mathbf{x}^j(k)), \tag{1a}$$

$$\mathbf{x}^i(k) = \mathbf{r}^i(k) - \mathbf{p}^i(k), \tag{1b}$$

starting at $\mathbf{p}^i(0) = \mathbf{0}$, then the trajectory $k \mapsto \mathbf{p}^i(k)$ of each agent $i \in \mathcal{V}$ is bounded and satisfies

$$\lim_{k\to\infty} \Big\| \mathbf{x}^i(k) - \frac{1}{N}\sum_{j=1}^{N} \mathbf{r}^j(k) \Big\| \leq \frac{\gamma(\infty)\,\delta}{\beta\hat{\lambda}_2}, \tag{2}$$

where $\sup_{\bar{k}\in\mathbb{Z}_{\geq k}} \|(\mathbf{I}_N - \frac{1}{N}\mathbf{1}_N\mathbf{1}_N^\top)(\mathbf{r}(\bar{k}+1) - \mathbf{r}(\bar{k}))\| = \gamma(k) < \infty$, and $\hat{\lambda}_2$ is second smallest eigenvalue of $\mathbf{L}$.

3. PROBLEM DEFINITION

In this section, we formalize our distributed containment control problem of interest. We assume that a group of M (M can change with time) mobile leaders are moving with a bounded velocity on a $\mathbb{R}^2$ or $\mathbb{R}^3$ space. We let $\mathbf{x}_{L,j}(t)$ be the position vector of leader $j \in \{1, \dots, M\}$ at time $t \in \mathbb{R}_{\geq 0}$. In our setting, a set of networked mobile agents $\mathcal{V} = \{1, \cdots, N\}$ monitors the leaders. The communication topology $\mathcal{G}$ of the agents is a strongly connected and weight-balanced digraph and the agents can communicate at discrete-times $t_k = k\delta_c$, $k \in \mathbb{Z}_{\geq 0}$, $\delta_c \in \mathbb{R}_{>0}$. The agents sample the leaders at sampling times $t^s_k = k\delta_s$, $k \in \mathbb{Z}_{\geq 0}$, $\delta_s \in \mathbb{R}_{>0}$. We let $\mathcal{V}^i_L(t^s_k)$ be the set of leaders observed by agent $i \in \mathcal{V}$ at sampling time t^s_k. Between each sampling time, agent $i \in \mathcal{V}$ uses $\mathbf{x}_{L,j}(t) = \mathbf{x}_{L,j}(t^s_k)$ and $\mathcal{V}^i_L(t) = \mathcal{V}^i_L(t^s_k)$, $t \in [t^s_k, t^s_{k+1})$, $k \in \mathbb{Z}_{\geq 0}$, $j \in \mathcal{V}^i_L(t^s_k)$. At every sampling time $t^s_k \in \mathbb{R}_{\geq 0}$, we let $\mathcal{V}_L(t^s_k)$ be the set of the mobile leaders that are observed jointly by the agents $\mathcal{V}$, i.e., $\mathcal{V}_L(t^s_k) = \cup_{i=1}^{N} \mathcal{V}^i_L(t^s_k)$ (see Fig. 1). We let $\mathcal{V}_o(t^s_k) \subset \mathcal{V}$ be the set of the agents that observe at least one leader at t^s_k, $k \in \mathbb{Z}_{\geq 0}$; we assume that $\mathcal{V}_o(t^s_k) \neq \emptyset$. Our objective in this paper is to design a distributed control algorithm that enables each agent $i \in \mathcal{V}$ to derive its local state $\boldsymbol{\chi}^i$ to asymptotically track $\mathrm{Co}(\mathcal{V}_L(t^s_k))$, the convex hull of the set of the location of the leaders $\mathcal{V}_L(t^s_k)$ with a bounded error $e \geq 0$ (to simplify notation, we wrote $\mathrm{Co}(\{\mathbf{x}_{L,j}(t)\}_{j\in\mathcal{V}_L(t)})$ as $\mathrm{Co}(\mathcal{V}_L(t))$). We state our objective as

$$\|\boldsymbol{\chi}^i(t_k) - \bar{\mathbf{x}}_L(t_k)\| \leq e, \quad i \in \mathcal{V}. \tag{3}$$

where $\bar{\mathbf{x}}_L(t_k) \in \mathrm{Co}(\mathcal{V}_L(t_k))$. We assume that the agents have no knowledge about the motion model of the leaders. As the information of each agent takes some time to propagate through the network, tracking the convex hull of an arbitrarily fast moving leader set with zero error is not feasible unless agents have some a priori information about the dynamics generating the signals. Also, how fast the information travels across the network $\mathcal{G}$ depends on the connectivity of $\mathcal{G}$. Interestingly, as expected, we will show that the size of the error e is going to be a function of the bound on the velocity of the leaders and $\hat{\lambda}_2$, which is a measure of connectivity of the network. We will also show that when the leader set is stationary and observed by the same set of agents, the agents converge to a point in the convex hull of the leader set, i.e., $e = 0$.

4. CONTAINMENT CONTROL ALGORITHM

In this section, we present our solution for the distributed containment control problem stated in Section 3. If the

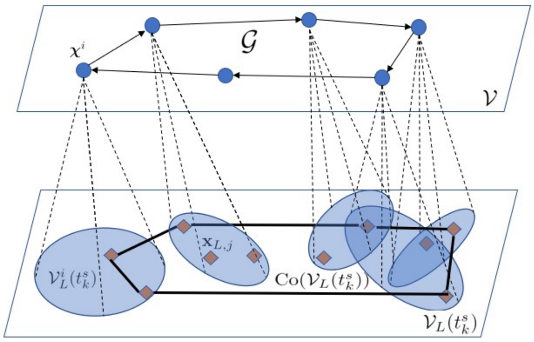

Fig. 1. The agent set and the leader set. The ellipsoids show the observation zone of the observing agents.

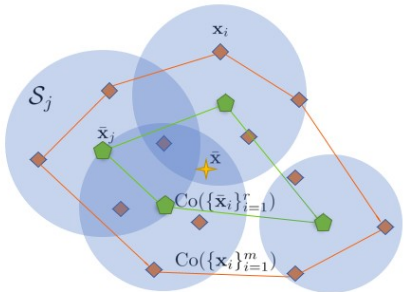

Fig. 2. Graphical demonstration of Lemma 2 for an example case.

number of the leaders is equal to the number of the agents, i.e., $|\mathcal{V}_L(t_k^s)| = N$, and $\mathcal{V}_L^i(t_k^s) \cap \mathcal{V}_L^j(t_k^s) = \emptyset$ for any $i, j \in \mathcal{V}$, $i \neq j$, a simple solution to our containment problem of interest is to implement the dynamic average consensus algorithm (1) with $\mathbf{r}^i(k) = \sum_{j \in \mathcal{V}_L^i(t_k)} \mathbf{x}_{L,j}(t_k)$ for $i \in \mathcal{V}$ (when $\mathcal{V}_L^i(t_k) = \emptyset$, we use $\mathbf{r}^i(k) = \mathbf{0}$). This is because, in this scenario, as certified by Lemma 1, the agents track the geometric center of the leader set, which is a point in the convex hull of the leader set, with a bounded error. However, in general, as the agents monitor the leader set, the observed leader sets $\mathcal{V}_L^i(t_k^s)$, $i \in \mathcal{V}$ of the agents likely have overlap and also the number of the leaders are different than the number of the agents, see Fig 1. In what follows, we present a simple solution for the containment problem that works for the general case.

To present our solution, we first make the following key observation about the convex hull of a set of points $\{\mathbf{x}_i\}_{i=1}^m$ in an Euclidean space. The proof of this result is omitted due to space limitation and will appear elsewhere.

Lemma 2. (Auxiliary result on convex hull of a set of points). Consider a set of points $\{\mathbf{x}_i\}_{i=1}^m$ in $\mathbb{R}^2$ or $\mathbb{R}^3$. Let $\mathcal{S}_j \neq \emptyset$, $j \in \{1, \cdots, r\}$, be a subset of $\{1, \cdots, m\}$. Let $\bar{\mathbf{x}}_j = \frac{\sum_{k \in \mathcal{S}_j} \mathbf{x}_k}{|\mathcal{S}_j|}$, $j \in \{1, \cdots, r\}$. Then, the point

$$\bar{\mathbf{x}} = \frac{\sum_{i=1}^r \bar{\mathbf{x}}_i}{r} \tag{4}$$

is a point in $\mathrm{Co}(\{\mathbf{x}_j\}_{j=1}^m)$. □

We note here that in Lemma 2, $\mathcal{S}_i \cap \mathcal{S}_j = \emptyset$, $i, j \in \{1, \cdots, r\}$, is not required. An example case that demonstrates the result of Lemma 2 is shown in Fig. 2.

Now for the containment problem, consider the general case when $0 < |\mathcal{V}_o(t_k^s)| \leq N$, and for any two distinct agents $i, j \in \mathcal{V}_o(t_k^s)$, $\mathcal{V}_L^i(t_k^s) \cap \mathcal{V}_L^j(t_k^s)$ is not necessarily empty. Then, in light of Lemma 2, we know that

$$\bar{\mathbf{x}}_L(t_k^s) = \frac{\sum_{i=1}^N \mathbf{r}^i(t_k^s)}{|\mathcal{V}_o(t_k^s)|} \in \mathrm{Co}(\mathcal{V}_L(t_k^s)), \tag{5}$$

where

$$\mathbf{r}^i(t_k^s) = \begin{cases} \frac{\sum_{j \in \mathcal{V}_L^i(t_k^s)} \mathbf{x}_{L,j}(t_k^s)}{|\mathcal{V}_L^i(t_k^s)|}, & i \in \mathcal{V}_o(t_k^s), \\ \mathbf{0}, & i \in \mathcal{V} \backslash \mathcal{V}_o(t_k^s). \end{cases} \tag{6}$$

Based on this observation, we propose the following distributed solution for our containment problem of interest. Our solution uses two dynamic average consensus algorithms of the form (1), to obtain the numerator and the denominator of $\bar{\mathbf{x}}_L$ in (5).

Theorem 3. (Distributed containment control algorithm). Let the communication topology of the agents $\mathcal{V}$ be a strongly connected and weight-balanced digraph $\mathcal{G}(\mathcal{V}, \mathcal{E})$. Assume that $\mathcal{V}_o(t_k^s) \neq \emptyset$ at each sampling time t_k^s, $k \in \mathbb{Z}_{\geq 0}$. Let $\mathbf{r}^i(t) = \mathbf{r}^i(t_k^s)$, $i \in \mathcal{V}$, and $\bar{\mathbf{x}}_L(t) = \bar{\mathbf{x}}_L(t_k^s)$ for $t \in [t_k^s, t_{k+1}^s)$, where $\mathbf{r}^i(t_k^s)$ and $\bar{\mathbf{x}}_L(t_k^s)$ are given respectively, in (5) and (6). Moreover, let $\bar{r}^i(t_k) = 1$ if $i \in \mathcal{V}_o(t_k)$, and $\bar{r}^i(t_k) = 0$ if $i \in \mathcal{V} \backslash \mathcal{V}_o(t_k)$. Suppose that $\sup_{\bar{k} \in \mathbb{Z}_{\geq k}} \|\mathbf{r}(t_{\bar{k}+1}) - \mathbf{r}(t_{\bar{k}})\| = \gamma(t_k) < \infty$, and $\sup_{\bar{k} \in \mathbb{Z}_{\geq k}} \|\bar{\mathbf{r}}(t_{\bar{k}+1}) - \bar{\mathbf{r}}(t_{\bar{k}})\| = \bar{\gamma}(t_k) < \infty$. Assume that each agent $i \in \mathcal{V}$ implements the distributed algorithm

$$\mathbf{p}^i(t_{k+1}) = \mathbf{p}^i(t_k) + \delta_c \beta \sum_{j=1}^N \mathsf{a}_{ij}(\mathbf{w}^i(t_k) - \mathbf{w}^j(t_k)), \tag{7a}$$

$$\mathbf{w}^i(t_k) = \mathbf{r}^i(t_k) - \mathbf{p}^i(t_k), \tag{7b}$$

$$q^i(t_{k+1}) = q^i(t_k) + \delta_c \beta \sum_{j=1}^N \mathsf{a}_{ij}(z^i(t_k) - z^j(t_k)), \tag{7c}$$

$$z^i(t_k) = \bar{r}^i(t_k) - q^i(t_k), \tag{7d}$$

$$\boldsymbol{\chi}^i(t_k) = \frac{\mathbf{w}^i(t_k)}{\max\{\epsilon, |z^i(t_k)|\}}, \tag{7e}$$

$\beta \in \mathbb{R}_{>0}$, initialized at $\mathbf{p}^i(t_0) = \mathbf{0}$ and $q^i(t_0) = 0$, with a communication stepsize $\delta_c \in (0, \beta^{-1}(\mathsf{d}^{\max})^{-1})$ ($t_k = \delta_c k$, $k \in \mathbb{Z}_{\geq 0}$). Here, $0 < \epsilon < 1/N$ is a small positive real number. Then, there exists a bounded value $e \in \mathbb{R}_{>0}$ such that

$$\|\boldsymbol{\chi}^i(t_k) - \bar{\mathbf{x}}_L(t_k)\| \leq e, \quad t_k \in \mathbb{R}_{\geq 0}, \quad i \in \mathcal{V}, \tag{8}$$

Moreover, if for a finite $\bar{k} \in \mathbb{Z}_{\geq 0}$ we have $\bar{\gamma}(t_k) = 0$ for all $k \in \mathbb{Z}_{\geq \bar{k}}$, i.e., the set of observing agents is not changing with time for $t_k \geq t_{\bar{k}}$, we have

$$\lim_{k \to \infty} \|\boldsymbol{\chi}^i(t_k) - \bar{\mathbf{x}}_L(t_k)\| \leq \frac{N \gamma(\infty)\, \delta_c}{|\mathcal{V}_o(t_{\bar{k}})|\, \beta \hat{\lambda}_2}, \quad i \in \mathcal{V}. \tag{9}$$

Proof. Given the initial conditions and stated bounds on $\mathbf{r} = [\{\mathbf{r}^i\}_{i \in \mathcal{V}}]$ and $\bar{\mathbf{r}} = [\{\bar{r}^i\}_{i \in \mathcal{V}}]$, by invoking Lemma 1, we conclude that for $k \in \mathbb{Z}_{\geq 0}$ the trajectories $k \mapsto \mathbf{w}^i(t_k)$ and $k \mapsto z^i(t_k)$, $i \in \mathcal{V}$ are bounded and satisfy

$$\lim_{k \to \infty} \frac{|\mathcal{V}_o(t_k)|}{N} \Big\| \mathbf{w}^i(t_k) / \frac{N}{|\mathcal{V}_o(t_k)|} - \bar{\mathbf{x}}_L(t_k) \Big\| = \lim_{k \to \infty} \Big\| \mathbf{w}^i(t_k) - \frac{1}{N} \sum_{j \in \mathcal{V}_o(t_k)} \mathbf{r}^j(t_k) \Big\| \leq \frac{\gamma(\infty)\, \delta_c}{\beta \hat{\lambda}_2}, \tag{10a}$$

$$\lim_{k \to \infty} \Big| z^i(t_k) - \frac{|\mathcal{V}_o(t_k)|}{N} \Big| \leq \frac{\bar{\gamma}(\infty)\, \delta_c}{\beta \hat{\lambda}_2}. \tag{10b}$$

Therefore, we can conclude that $\|\boldsymbol{\chi}^i(t_k)\|$ is finite for all $k \in \mathbb{Z}_{\geq 0}$, which confirms also (8). Next, if for a finite

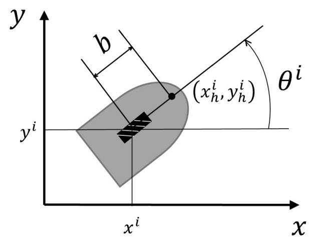

Fig. 3. A unicycle robot and its corresponding state variables.

$\bar{k} \in \mathbb{Z}_{\geq 0}$ we have $\bar{\gamma}(t_k) = 0$ for all $k \in \mathbb{Z}_{\geq \bar{k}}$, we note that from (10b) we obtain $\lim_{k\to\infty} z^i(t_k) = \frac{|V_o(t_{\bar{k}})|}{N}$, $i \in \mathcal{V}$. Consequently, $\lim_{k\to\infty} \boldsymbol{\chi}^i(t_k) = \frac{N}{|V_o(t_{\bar{k}})|} \lim_{k\to\infty} \mathbf{w}^i(t_k)$, which along with (10a) confirms (9).

It is worth nothing that if in addition to the set of observing agents not changing with time for $t_k \geq t_{\bar{k}}$ for all $k \geq \bar{k} \geq 0$, we also have $\gamma(t_k) = 0$ for $k \geq \hat{k} \geq \bar{k}$ (every $i \in \mathcal{V}_o(t_k)$ observes the same leaders for $t \geq t_{\hat{k}}$ and the leaders converge to a stationary configuration), it follows from (9) that $\lim_{k\to\infty} \boldsymbol{\chi}^i(t_k) = \bar{\mathbf{x}}_L(\infty)$, $i \in \mathcal{V}$. We also note here that use of a dynamic average consensus algorithm in constructing the distributed containment controller (7) results in a better tracking performance than use of a static average consensus (Olfati-Saber and Murray (2004)) that gets re-initialized at each sampling time with the new observed inputs. For more details see the example scenario discussed in Figure 2 of Kia et al. (2019), which compared performance of a dynamic average consensus algorithm and a static average consensus algorithm for sampled time-varying input signals.

5. AN APPLICATION EXAMPLE: CONTAINMENT PROBLEM FOR A GROUP OF NETWORKED UNICYCLE ROBOTS

In this section, we demonstrate how a group of N unicycle robots with strongly connected and weight-balanced topology can track the convex hull of a set of leaders that they observe. Let the dynamics of each robot be expressed by

$$\mathbf{x}^i = \begin{bmatrix} \dot{x}^i \\ \dot{y}^i \\ \dot{\theta}^i \end{bmatrix} = \begin{bmatrix} v^i \cos\theta^i \\ v^i \sin\theta^i \\ \omega^i \end{bmatrix}, \qquad i \in \mathcal{V}, \tag{11}$$

where x^i, $y^i \in \mathbb{R}$ are the coordinates in 2 dimensional space and $\theta^i \in \mathbb{R}$ is the heading angle of the robot. v^i, $\omega^i \in \mathbb{R}$ are, respectively, the linear velocity and angular velocity of robot $i \in \mathcal{V}$. We define a head point

$$\mathbf{x}_h^i = \begin{bmatrix} x_h^i \\ y_h^i \end{bmatrix} = \begin{bmatrix} x^i + b\cos\theta^i \\ y^i + b\sin\theta^i \end{bmatrix}, \qquad i \in \mathcal{V}. \tag{12}$$

as a position at a distance b along the main axis of the robot i as shown in Fig. 3. The head point can be the place where the robot observation sensor (e.g., camera) is located or a point that carry sensitive goods. There is another group of mobile robots called leaders, which have the ability to avoid obstacles. These unicycle robots want to track the leaders and keep their head points inside the convex hull formed by the leaders. The unicycle robots communicate to their neighbors at the instant t_k, $k \in \mathbb{Z}_{\geq 0}$ with the time period δ_c and jointly detect the positions of the leaders $\mathbf{x}_{L,j}$, $j \in \mathcal{V}_L(t_k^s)$ at the instant t_k^s with time period δ_s, but they do not have the information of the leaders' dynamics.

To control the unicycle robots to stay in the convex hull $\mathrm{Co}(\mathcal{V}_L(t_k^s))$ of the moving leaders, we propose a two-layer containment control scheme. The first layer is a distributed observer with the discrete-time process of (7) proposed in Theorem 3 which produces a estimated position $\boldsymbol{\chi}^i(t_k)$ of a pin $\bar{\mathbf{x}}_L \in \mathrm{Co}(\mathcal{V}_L(t_k^s))$ at every discrete communication time t_k. By this distributed observer, the robots can estimate the position of the pin in the convex hull even though some of them could not detect any leader. Then, in order to track the discrete estimate $\boldsymbol{\chi}^i(t_k)$ in time, we use a local finite-time converging controller which drives the head point of the continuous-time unicycle robot to track $\boldsymbol{\chi}^i(t_k)$ before the next communication time t_{k+1}. That is

$$\mathbf{x}_h^i(t_{k+1}) = \boldsymbol{\chi}^i(t_k), \qquad i \in \mathcal{V}. \tag{13}$$

The following result gives the local tracking control that realizes the objective (13).

Theorem 4. (Finite-time converging controller to track the convex hull of the leaders). Consider a group of N unicycle robots with dynamics described by (11) and the head point defined by (12). Let the communication topology of the robots be a strongly connected and weight-balanced digraph and suppose the robots are implementing the containment controller (7). Starting at an initial condition $\mathbf{x}^i(t_0) \in \mathbb{R}^3$, let for $t \in [t_k, t_{k+1})$

$$v^i(t) = u_1^i \cos\theta^i + u_2^i \sin\theta^i, \tag{14a}$$

$$\omega^i(t) = \frac{u_2^i \cos\theta^i - u_1^i \sin\theta^i}{b}, \tag{14b}$$

$$\mathbf{u}^i(t) = \begin{bmatrix} u_1^i \\ u_2^i \end{bmatrix} = \frac{1}{\delta_c}(\boldsymbol{\chi}^i(t_k) - \mathbf{x}_h^i(t_k)). \tag{14c}$$

Then, for every robot $i \in \mathcal{V}$, we have $\lim_{t\to t_{k+1}^-} \mathbf{x}_h^i(t) = \boldsymbol{\chi}^i(t_k)$ for all $k \in \mathbb{Z}_{\geq 0}$.

Proof. Note that

$$\dot{\mathbf{x}}_h^i = \begin{bmatrix} \dot{x}^i - b\dot{\theta}^i \sin\theta^i \\ \dot{y}^i + b\dot{\theta}^i \cos\theta^i \end{bmatrix}, \qquad i \in \mathcal{V}.$$

Substituting for robot dynamics from (11) and using the velocity inputs (14a) and (14b), we arrive at (see De Luca et al. (2001))

$$\dot{\mathbf{x}}_h^i = \begin{bmatrix} u_1^i \\ u_2^i \end{bmatrix}. \tag{15}$$

Then, using the control input (14c), we obtain

$$\mathbf{x}_h^i(t) = \mathbf{x}_h^i(t_k) + \frac{t - t_k}{\delta_c}(\boldsymbol{\chi}^i(t_k) - \mathbf{x}_h^i(t_k)), \quad t \in [t_k, t_{k+1}),$$

for $i \in \mathcal{V}$, which confirms that $\lim_{t\to t_{k+1}^-} \mathbf{x}_h^i(t) = \boldsymbol{\chi}^i(t_k)$ for all $k \in \mathbb{Z}_{\geq 0}$.

6. NUMERAL DEMONSTRATION

In this section, we use a numerical example to demonstrate the performance of the distributed containment control algorithm (7) and the local tracking controller (14). We consider a group of 6 unicycle robots with dynamics (11) whose communication topology is given in Fig. 4. A position defined by (12) is the head point for each robot. The robots aim to make their head points to track the convex hull that is formed by 10 mobile leaders with unknown dynamics, moving in a 2D flat space. The robots detect the leaders at the frequency 0.5 Hz, i.e., $\delta_s = 2$ seconds. The observed leaders by the robots are time varying as described below (time intervals are in second):

- $0 \leq t_k^s < 5$: $\mathcal{V}_L^1(t_k^s) = \{1,4,6,8\}$, $\mathcal{V}_L^2(t_k^s) = \{2,4,7,8,10\}$, $\mathcal{V}_L^3(t_k^s) = \{3,4,5,9\}$, $\mathcal{V}_L^4(t_k^s) = \emptyset$, $\mathcal{V}_L^5(t_k^s) = \{1,3,9\}$ and $\mathcal{V}_L^6(t_k^s) = \emptyset$,

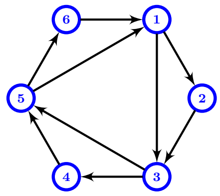

Fig. 4. A strongly connected and weight-balance topology with edge weights of 0 and 1.

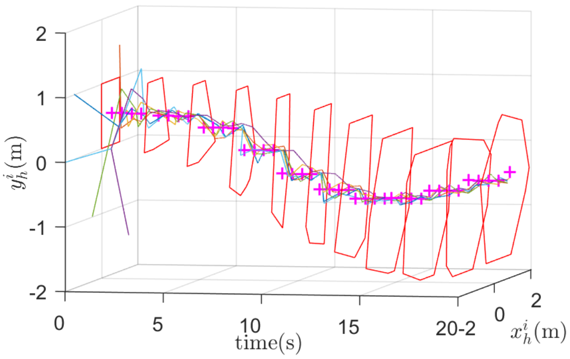

Fig. 5. The tracking performance of the robots while implementing the distributed containment observer (7) and the local control (14) with $\delta_c = 0.5$ seconds: the lines show the trajectory of (x_h^i, y_h^i) vs. time, while "+" show the location of $\bar{\mathbf{x}}_L(t_k)$ of the leaders. The red polygons indicate the convex hull formed by the moving leaders at each sampling time.

- $5 \leq t_k^s < 10$: $\mathcal{V}_L^1(t_k^s) = \{3,5,6,8\}$, $\mathcal{V}_L^2(t_k^s) = \{1,2,7,9,10\}$, $\mathcal{V}_L^3(t_k^s) = \{3,4,5,9\}$, $\mathcal{V}_L^4(t_k^s) = \emptyset$, $\mathcal{V}_L^5(t_k^s) = \{1,3,9\}$ and $\mathcal{V}_L^6(t_k^s) = \{2,5,7,9\}$,
- $10 \leq t_k^s \leq 20$: $\mathcal{V}_L^1(t_k^s) = \{1,2,5,8\}$, $\mathcal{V}_L^2(t_k^s) = \{2,3,6,7,10\}$, $\mathcal{V}_L^3(t_k^s) = \{3,4,5,9\}$, $\mathcal{V}_L^4(t_k^s) = \{3,10\}$, $\mathcal{V}_L^5(t_k^s) = \{1,3,9\}$ and $\mathcal{V}_L^6(t_k^s) = \{2,5,7,9\}$.

The robots implement their distributed containment observer (7) at the communication frequency of 2 Hz, i.e., $\delta_c = 0.5$ seconds, to estimate $\bar{\mathbf{x}}_L(t_k)$ which is in the convex hull formed by the leaders. The time varying convex hull of the observed leaders are shown by the red closed curves in Fig. 5 and Fig. 6. These figures along with Fig. 7 also show that for the given scenario, our proposed tracking control scheme achieves its tracking goal satisfactorily. Figure 8 and Fig. 9 show the performance of our containment controller when robots communicate in a higher frequency with $\delta_c = 0.1$ seconds. As we can see, the containment observer converges faster in every sampling interval and the tracking error start to diminish over time. It is very likely that the δ_s is much bigger than δ_c. For such cases, the containment observer results shown in Fig. 7 and Fig. 9 suggest that instead of deriving the local states to satisfy (13) we can use a lower tracking frequency to avid the transient perturbation at the beginning of each sampling time.

7. CONCLUSION

In this paper, we first proposed a distributed algorithm to solve a containment control problem for a group of follower agents that are communicating in discrete-time over a strongly connected and weight-balance digraph. In our problem of interest, the follower agents observed the location of a set of leader agents at a pre-specified sampling times and used our proposed algorithm to track the convex hull of the dynamic leaders. After developing the dis-

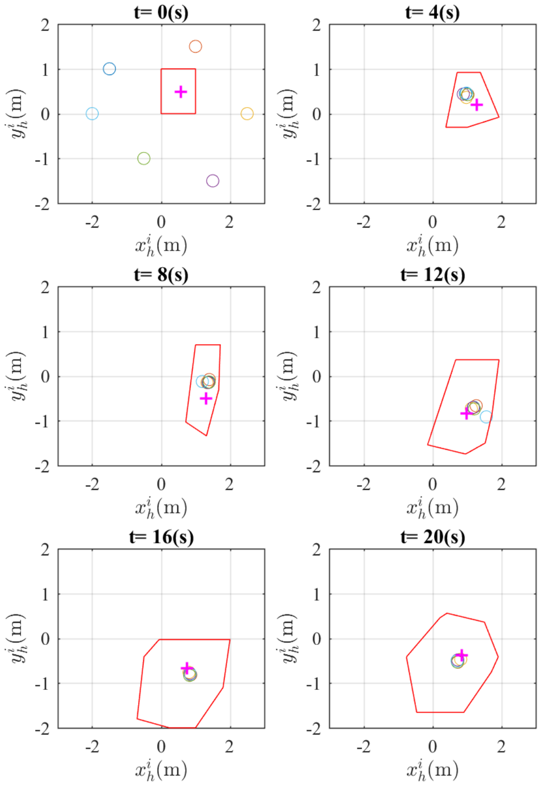

Fig. 6. The snapshots showing the leaders convex hull (red polygons), the location of the head point (x_h^i, y_h^i) of the robots ("o" markers) and the location of $\bar{\mathbf{x}}_L(t_k)$ ("+" marker), when robots implement the distributed containment observer (7) and the local control (14) with $\delta_c = 0.5$ seconds.

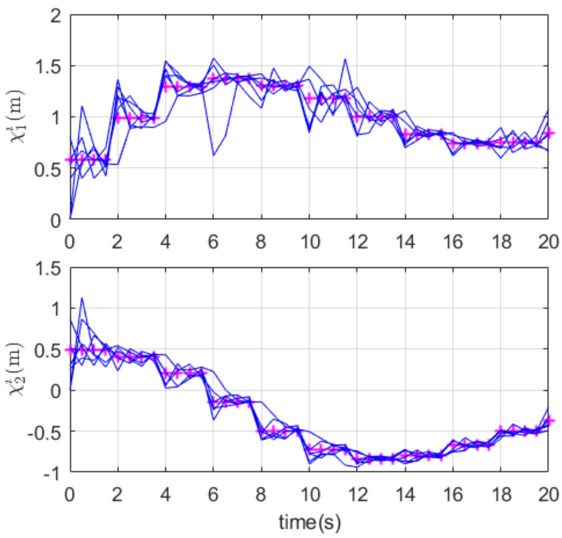

Fig. 7. Time history of the output of the containment observer (7) with $\delta_c = 0.5$ seconds: the blue curves show $\boldsymbol{\chi}^i$ and the markers "+" show the location of the coordinates of $\bar{\mathbf{x}}_L(t_k)$. The jumps in the location of $\bar{\mathbf{x}}_L(t_k)$ is due to the motion of the leaders and also the changes in the set of the observed leaders by each agent.

tributed containment algorithm, we proposed a two-layer control scheme to apply the containment control algorithm for a group of unicycle robots. In this control scheme, at the first layer we used the distributed containment control algorithm as the observer to estimate the location of a point in the convex hull of the moving leaders. In the second layer we used a local finite-time controller to drive

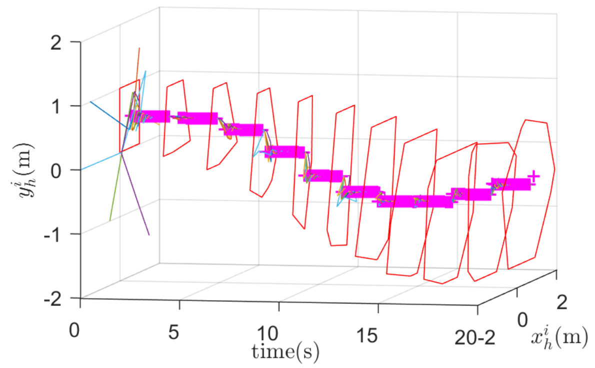

Fig. 8. The tracking performance of the robots while implementing the distributed containment observer (7) and the local control (14) with $\delta_c = 0.1$ seconds: the lines show the trajectory of (x_h^i, y_h^i) vs. time, while "+" show the location of $\bar{\mathbf{x}}_L(t_k)$ of the leaders. The red polygons indicate the convex hull formed by the moving leaders at each sampling time.

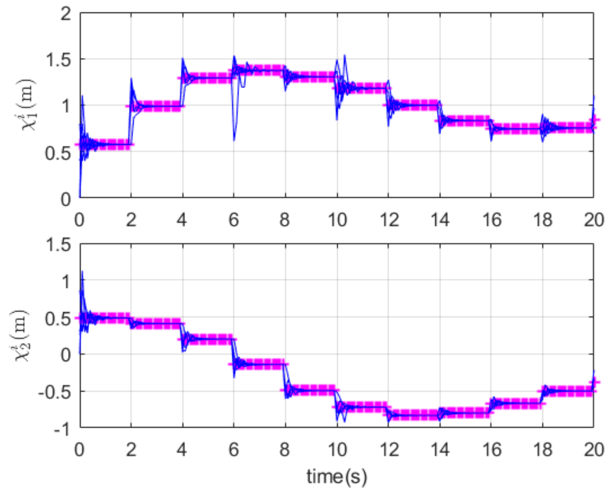

Fig. 9. Time history of the output of the containment observer (7) with $\delta_c = 0.1$ seconds: the blue curves show $\boldsymbol{\chi}^i$ and the markers "+" show the location of the coordinates of $\bar{\mathbf{x}}_L(t_k)$. The jumps in the location of $\bar{\mathbf{x}}_L(t_k)$ is due to the motion of the leaders and also the changes in the set of the observed leaders by each agent.

the unicycle robots to track the output of the containment observer in finite time. In this framework, dynamics of the unicycle robots were continuous-time but the robots communicated with each other in discrete-time fashion. Numerical simulations demonstrated the efficiency of our proposed algorithms.

REFERENCES

Bullo, F., Cortés, J., and Martínez, S. (2009). *Distributed Control of Robotic Networks*. Applied Mathematics Series. Princeton University Press.

Cao, Y., Ren, W., and Egerstedt, M. (2012). Distributed containment control with multiple stationary or dynamic leaders in fixed and switching directed networks. *Automatica*, 48(8), 1586–1597.

De Luca, A., Oriolo, G., and Vendittelli, M. (2001). Control of wheeled mobile robots: An experimental overview. In *Ramsete*, 181–226. Springer.

Dimarogonas, D.V., Egerstedt, M., and Kyriakopoulos, K.J. (2006). A Leader-based Containment Control Strategy for Multiple Unicycles. In *Proceedings of the 45th IEEE Conference on Decision and Control*, 5968–5973. IEEE.

Engelberger, J.F. (2012). *Robotics in practice: management and applications of industrial robots*. Springer Science & Business Media.

Galbusera, L., Ferrari-Trecate, G., and Scattolini, R. (2013). A hybrid model predictive control scheme for containment and distributed sensing in multi-agent systems. *Systems & Control Letters*, 62(5), 413 – 419.

Haghshenas, H., Badamchizadeh, M.A., and Baradarannia, M. (2015). Containment control of heterogeneous linear multi-agent systems. *Automatica*, 54, 210–216.

Hou, S.P. and Cheah, C.C. (2011). Can a simple control scheme work for a formation control of multiple autonomous underwater vehicles? *IEEE Transactions on Control Systems Technology*, 19(5), 1090–1101.

Iyengar, S.S. and Brooks, R.R. (2016). *Distributed Sensor Networks: Sensor Networking and Applications (Volume Two)*. CRC press.

Ji, M., Ferrari-Trecate, G., Egerstedt, M., and Buffa, A. (2008). Containment Control in Mobile Networks. *IEEE Transactions on Automatic Control*, 53(8), 1972–1975.

Kia, S.S., Scoy, B.V., Cortés, J., Freeman, R.A., Lynch, K.M., and Martínez, S. (2019). Tutorial on dynamic average consensus: The problem, its applications, and the algorithms. *IEEE Control Systems Magazine*, 39(3), 40–72.

Li, J., Ren, W., and Xu, S. (2012). Distributed containment control with multiple dynamic leaders for double-integrator dynamics using only position measurements. *IEEE Transactions on Automatic Control*, 57(6), 1553–1559.

Liu, H., Cheng, L., Tan, M., and Hou, Z. (2014). Containment control of double-integrator multi-agent systems with aperiodic sampling: A small-gain theorem based method. In *Proceedings of the 33rd Chinese Control Conference*, 1407–1412.

Liu, H., Cheng, L., Tan, M., and Hou, Z.G. (2015). Containment control of continuous-time linear multi-agent systems with aperiodic sampling. *Automatica*, 57, 78–84.

Liu, H., Xie, G., and Wang, L. (2012a). Containment of linear multi-agent systems under general interaction topologies. *Systems & Control Letters*, 61(4), 528 – 534.

Liu, H., Xie, G., and Wang, L. (2012b). Necessary and sufficient conditions for containment control of networked multi-agent systems. *Automatica*, 48(7), 1415 – 1422.

Mei, J., Ren, W., and Ma, G. (2012). Distributed containment control for Lagrangian networks with parametric uncertainties under a directed graph. *Automatica*, 48(4), 653–659.

Olfati-Saber, R. and Murray, R.M. (2004). Consensus problems in networks of agents with switching topology and time-delays. *IEEE Transactions on Automatic Control*, 49(9), 1520–1533.

Wang, X., Yadav, V., and Balakrishnan, S.N. (2007). Cooperative uav formation flying with obstacle/collision avoidance. *IEEE Transactions on Control Systems Technology*, 15(4), 672–679.

Wang, X., Li, S., and Shi, P. (2014). Distributed Finite-Time Containment Control for Double-Integrator Multiagent Systems. *IEEE Transactions on Cybernetics*, 44(9), 1518–1528.

Zheng, Y. and Wang, L. (2014). Containment control of heterogeneous multi-agent systems. *International Journal of Control*, 87(1), 1–8.