# Distributed Time-Varying Optimization With State-Dependent Gains: Algorithms and Experiments

Shan Sun, *Graduate Student Member, IEEE*, Yifan Zhang, Peng Lin, Wei Ren, *Fellow, IEEE*, and Jay A. Farrell, *Fellow, IEEE*

***Abstract*—This brief addresses distributed continuous-time optimization problems with time-varying objective functions. The goal is for multiple agents to cooperatively minimize the sum of local time-varying objective functions with only local interaction and information while explicitly taking into account distributed adaptive gain design. Here, the optimal point is time varying and creates an optimal trajectory. First, for the unconstrained case, a distributed nonsmooth algorithm coupled with a state-dependent gain is proposed. It is shown that the interaction gain for each agent can be computed according to the variation of the Hessian and gradient information of the convex local objective functions so that the algorithm can solve the time-varying optimization problem without imposing a bound on any information about the local objective functions. Second, for the case where there exist common time-varying linear equality constraints, an extended algorithm is presented, where local Lagrangian functions are introduced to address the equality constraints. The asymptotic convergence of both algorithms to the optimal solution is proved. Numerical simulations are presented to illustrate the theoretical results. In addition, the one proposed algorithm is experimentally implemented and validated on a multi-Crazyflie platform.**

***Index Terms*—Distributed continuous-time optimization, state-dependent control gains, time-varying objective functions.**

## I. Introduction

RECENTLY, the solution of distributed optimization problems by multiagent systems has received increasing attention due to its broad applications in sensor networks, big-data analysis, smart grids, multirobot teams, and resource allocation. The goal is for multiple agents to solve an optimization problem cooperatively in a distributed manner where the team objective function is the sum of local objective functions, each of which is known to only one agent. There are many existing results on distributed optimization in the discrete-time settings (see [1] and references therein).

*Related Works:* Recently, significant results on distributed continuous-time optimization have been published [2]–[12]. The continuous-time algorithms have application in motion coordination of multiagent systems. For example, multiple agents modeled by continuous-time dynamics might need to rendezvous or dock at a location that is optimal for the team (with respect to certain team performance functions). The aforementioned works [2]–[12] focus on time-invariant objective functions, whereas the time-varying counterparts are encountered in many applications (see [13]–[16] and their references). Time-varying objective functions make the design and analysis for the distributed optimization problem much more complex. In this brief, we are interested in solving distributed continuous-time optimization problems with time-varying objective functions for agents that have single-integrator dynamics.

In the literature on distributed continuous-time optimization with time-varying objective functions, some researchers combined the consensus method and optimization algorithms. The resulting algorithms require the convex objective function to satisfy restrictive assumptions [17]–[20], in addition to the general differentiable assumption. For example, Sun *et al.* [17] consider a class of distributed time-varying quadratic optimization problems, where not only the quadratic coefficients (Hessians) but also their first and second-order derivatives are required to be bounded. Hosseinzadeh *et al.* [18] propose a distributed solution for linear programming problems with possibly time-varying inequality constraints, and prove that the tracking error is proportional to the rate of change of the parameters. Distributed time-varying convex optimization problems are studied for multiagent systems with, respectively, linear and nonlinear dynamics in [19] and [20], where a bound is placed on the Hessians and the time rate of change of the gradients of the local objective functions.

Furthermore, the articles on the distributed time-varying optimization mentioned above all focus on unconstrained cases [19], [20], or constraint set cases [17], [18], while in some applications it is desirable to consider equality constraints. While equality constraints are considered in [8]–[11], the focus there is on distributed optimization with time-invariant objective functions and time-invariant equality constraints. Fazlyab *et al.* [21] focuses on the optimization problems with time-varying objective functions and time-varying inequality and equality constraints. However, the algorithms there can only be employed in a centralized manner. Although optimizing certain team performance functions in a distributed manner while considering the time-varying equality constraints is a highly motivating task in many multiagent applications, it has not been addressed in the literature.

*Contributions:* In this brief, the distributed continuous-time optimization problem is studied for more general time-varying

Manuscript received October 15, 2020; revised January 18, 2021; accepted February 8, 2021. Date of publication March 5, 2021; date of current version December 15, 2021. Manuscript received in final form February 9, 2021. This work was supported by the National Science Foundation under Grant ECCS-1920798. Recommended by Associate Editor M. Abbaszadeh. *(Corresponding author: Wei Ren.)*

Shan Sun, Yifan Zhang, Wei Ren, and Jay A. Farrell are with the Department of Electrical and Computer Engineering, University of California, Riverside, CA 92521 USA (e-mail: ssun029@ucr.edu; yzhan407@ucr.edu; ren@ece.ucr.edu; jay.farrell@ucr.edu).

Peng Lin is with the School of Information Science and Engineering, Central South University, Changsha 410083, China (e-mail: lin_peng0103@sohu.com).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TCST.2021.3058845.

Digital Object Identifier 10.1109/TCST.2021.3058845

objective functions. The main contributions are given as follows. First, we propose a distributed nonsmooth algorithm with state-dependent gains for the unconstrained case. Here, the interaction gain of each agent is adjusted according to the variation of the Hessian and gradient information of the convex local objective functions, so that the algorithm can solve the time-varying optimization problem without imposing a bound on any information about the local objective functions (see Remark 4 for details). Therefore, the proposed algorithm can deal with more general objective functions. To the best of our knowledge, this is the first work in the literature of distributed continuous-time -varying optimization that the optimization problem can be solved without imposing a bound on any information about the local objective functions. Second, for the case where there exist common time-varying linear equality constraints, an extended algorithm is proposed. Local Lagrangian functions are introduced to address the equality constraints. Similarly, the time-varying constrained optimization problem can be solved without imposing a bound on any information about the local objective functions and constraint functions. Note that the distributed time-varying constrained optimization problem has its unique difficulties and is more challenging than the unconstrained counterpart since the local constraints are also time-varying. Both algorithms are distributed in the sense that each agent uses only information from itself and its neighbors and there is no need to know any global parameters. For both algorithms, it is shown that all agents achieve consensus in finite time and the consensus solution converges to the optimal solution asymptotically. Numerical simulations are presented to illustrate the theoretical results. One proposed algorithm is also experimentally validated on a multi-Crazyflie platform.

*Comparison With the Literature*[1]: To address the distributed optimization problem, [2]–[12] propose distributed continuous-time algorithms for the time-invariant objective functions. In contrast, the current paper proposes distributed continuous-time algorithms for the time-varying objective functions with zero tracking errors. In the literature, only a few works consider the distributed continuous-time time-varying optimization problem [17]–[20]. However, with our proposed algorithms, there is no need to impose a bound on certain information about the local objective functions as in all of the existing works that solve the distributed continuous-time time-varying optimization problem [17]–[20] (see related works in Section I for details). The proposed algorithms are partly inspired by Rahili and Ren [19]. Specifically, this brief and the work by Rahili and Ren [19] both introduce the idea of distributed average tracking to facilitate the distributed algorithm design. However, Rahili and Ren [19] place a bound on the Hessians and the rate of change of the gradients of the local objective functions. This brief presents a state-dependent gain design such that the above assumption can be removed. Therefore, this brief can deal with more general objective functions that cannot be handled in [19]. In addition, this brief takes into account the time-varying equality constraints which are not considered in [19]. Moreover, the approach in [19] increases the adaptive control gain until a consensus is achieved (see Remark 4). Therefore, the control gain there might become unnecessarily large. In contrast, the state-dependent gain in this brief is computed in such a way that the above situation is eliminated. The novel algorithm design herein introduces new challenges in the theoretical analysis.

This brief extends our prior adaptive algorithm with time-invariant objective functions presented in [12, Sec. III-B] to distributed time-varying unconstrained and constrained optimization cases. Besides more rigorous and detailed proof, additional numerical examples and experimental results are also presented.

[1]Since we focus on the continuous-time algorithms in this work, we do not present the comparison to the discrete-time algorithms here.

## II. Preliminaries

### A. Notation

Let $\mathbb{R}, \mathbb{R}^n$ and $\mathbb{R}^{n\times m}$ denote the sets of real numbers, real vectors of dimension $n$, and real matrices of size $n \times m$, respectively. Let $\mathbb{R}_{>0}$ and $\mathbb{R}_{\geq 0}$ represent, respectively, the set of positive and nonnegative real numbers. Let $\mathbf{1}_n$ (resp. $\mathbf{0}_n$) denote the vector of $n$ ones (resp. $n$ zeros), and $I_n$ denote the $n \times n$ identity matrix. For a matrix $A \in \mathbb{R}^{m\times n}$, $A^T$ (resp. $A^{-1}$) is the transpose (resp. inverse) of $A$. For a vector $x = [x_1, \ldots, x_n]^T \in \mathbb{R}^{n\times 1}$, $\mathrm{diag}(x) \in \mathbb{R}^{n\times n}$ represents the diagonal matrix with the elements in the main diagonal being the elements of $x$, $\|x\|_p$ represents the $p$-norm of the vector $x$, and $\mathrm{sgn}(x) = [\mathrm{sgn}(x_1), \ldots, \mathrm{sgn}(x_n)]^T$, where $\mathrm{sgn}(x_i) = -1$ if $x_i < 0$, $\mathrm{sgn}(x_i) = 1$ if $x_i > 0$, and otherwise $\mathrm{sgn}(x_i) = 0$. Let $\nabla f(x,t)$ and $\nabla^2 f(x,t)$ denote, respectively, the gradient and Hessian of the function $f(x,t) : \mathbb{R}^n \times \mathbb{R}_{\geq 0} \mapsto \mathbb{R}$ with respect to the vector $x$.

### B. Graph Theory

An undirected graph, is denoted by $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{A})$, where $\mathcal{V} = \{1, \ldots, n\}$ is the node set, $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ is the edge set, and $\mathcal{A} = [a_{ij}] \in \mathbb{R}^{n\times n}$ is the weighted adjacency matrix with entries $a_{ij},\ i, j \in \mathcal{V}$. For an undirected graph, an edge $(j,i)$ implies that nodes $i$ and $j$ are able to share information with each other, and $a_{ij} = 1$ if $(j,i) \in \mathcal{E}$ and $a_{ij} = 0$ otherwise. Here $a_{ij} = a_{ji}$. Let $\mathcal{N}_i = \{j \in \mathcal{V} : (j,i) \in \mathcal{E}\}$ denote the set of neighbors of node $i$. A path is a sequence of nodes connected by edges. An undirected graph is connected if for every pair of nodes there is a path connecting them.

## III. Distributed Time-Varying Unconstrained Optimization

Consider a multiagent system consisting of $n$ agents with an interaction topology described by the undirected graph $\mathcal{G}(t)$. Each agent can interact only with its local neighbors. Suppose that the agents satisfy the following single-integrator dynamics:

$$\dot{p}_i(t) = u_i(t) \tag{1}$$

where $p_i(t) \in \mathbb{R}^m$ and $u_i(t) \in \mathbb{R}^m$ are the state and control input of agent $i$. Our goal here is to design $u_i(t)$ using only local information and interactions with neighbors, such that all the agents cooperatively find the optimal solution $r^*(t) \in \mathbb{R}^m$ (assuming it exists for all $t \geq 0$) which is defined as

$$r^*(t) = \arg\min_{r(t)} \left\{ \sum_{i=1}^{n} f_i[r(t), t] \right\} \tag{2}$$

where $f_i[r(t), t] : \mathbb{R}^m \times \mathbb{R}_{\geq 0} \mapsto \mathbb{R}$ are the local objective functions. It is assumed that $f_i[r(t), t]$ is only known to agent $i$ and is twice continuously differentiable with respect to $r(t)$ and continuously differentiable with respect to $t$. Note that $\sum_{i=1}^{n} f_i[p_i(t), t] = \sum_{i=1}^{n} f_i[r(t), t]$, if $p_i(t) = p_j(t) = r(t)$ for all $i, j \in \mathcal{V}$, the above problem (2) is equivalent to finding the optimal solution $p^*(t) \in \mathbb{R}^{m*n}$ which is defined as

$$\begin{aligned} p^*(t) \;&=\; \arg\min_{p(t)} \left\{ \sum_{i=1}^{n} f_i[p_i(t), t] \right\} \\ &\text{Subject to } p_i(t) = p_j(t) \quad \forall i, j \in \mathcal{V} \end{aligned} \tag{3}$$

where $p(t)$ is the vector that concatenates the state vectors $p_i(t) \in \mathbb{R}^m$ of all the agents. Note that problem (2) will be solved as a consensus minimization problem with the time-varying team objective function $\sum_{i=1}^{n} f_i[p_i(t), t]$. Here, the goal is that each state $p_i(t)$ converges to the optimal solution $r^*(t)$, that is

$$\lim_{t \to \infty} \left[ p_i(t) - r^*(t) \right] = \mathbf{0}_m. \tag{4}$$

We introduce the following assumptions and the following lemma which are all standard in the recent literature [19].

*Assumption 1:* The graph $\mathcal{G}(t)$ is undirected and connected for all $t \geq 0$.

*Assumption 2:* The length of the time interval between any two contiguous switching topologies is greater than or equal to a given positive constant.

Arbitrary switching of the graph $\mathcal{G}(t)$ might lead to the Zeno behavior. Hence Assumption 2 is imposed to prevent the system from exhibiting the Zeno behavior.

*Assumption 3:* Each objective function $f_i[p_i(t), t]$ is uniformly strongly convex in $p_i(t)$ and its Hessian matrix $\nabla^2 f_i[p_i(t), t]$ is identical under identical local states $p_i(t)$ for all $t \geq 0$, i.e., $\nabla^2 f_i[p_i(t), t] \geq \alpha I_m$, for some $\alpha > 0$ and $\nabla^2 f_i[p_i(t), t] = \nabla^2 f_j[p_j(t), t]$ if $p_i(t) = p_j(t)$ for all $i, j \in \mathcal{V}$.

The uniform strong convexity of each objective function $f_i[r(t), t]$ implies the uniform strong convexity of $\sum_{i=1}^{n} f_i[r(t), t]$, such that the optimal trajectory $r^*(t)$ (assuming it exists for all $t \geq 0$) is unique for all $t \geq 0$. Moreover, due to the equivalence between (2) and (3), the optimal solution $p^*(t)$ defined in (3) is unique for all $t \geq 0$.

*Remark 1:* Assumption 3 requires that the Hessian matrix $\nabla^2 f_i[p_i(t), t]$ be identical under identical local states $p_i(t)$ for all $t \geq 0$, which might be restrictive. However, in the literature of distributed continuous-time time-varying optimization, it is common to assume that all the Hessian matrices $\nabla^2 f_i[p_i(t), t]$ are identical, i.e., $\nabla^2 f_i[p_i(t), t] = \nabla^2 f_j[p_j(t), t]$ for all $i, j \in \mathcal{V}$ and all $t \geq 0$ (see [19], [20]). In this brief, we reconsider the identical Hessian assumption, and relax it further. The algorithms herein do not need $\nabla^2 f_i[p_i(t), t] = \nabla^2 f_j[p_j(t), t]$ for all $p_i(t)$ and $p_j(t)$. Instead, they only need $\nabla^2 f_i[p_i(t), t] = \nabla^2 f_j[p_j(t), t]$ when $p_i(t) = p_j(t)$. Note that the identical Hessian condition can be satisfied in many situations, e.g., $f_i[p_i(t), t] = [\alpha\, p_i(t) + b_i(t)]^2$ with a positive constant $\alpha$ and a time-varying function $b_i(t)$, which is commonly used for robot control and energy minimization.

*Lemma 1 [22]: Let $f(r) : \mathbb{R}^m \to \mathbb{R}$ be a continuously differentiable convex function with respect to $r$. The function $f(r)$ is minimized at $r^*$ if and only if $\nabla f(r^*) = 0$.*

### *A. Distributed Algorithm Design*

This section presents and analyzes a distributed adaptive control algorithm for the time-varying optimization problem in (3). The controller for agent $i$ is

$$\begin{aligned} u_i(t) &= \phi_i(t) - \sum_{j \in \mathcal{N}_i(t)} \Big\{ \big[ \|\phi_i(t)\|_\infty + \|\phi_j(t)\|_\infty + \gamma_i + \gamma_j \big] \\ &\qquad \times \operatorname{sgn}\big[p_i(t) - p_j(t)\big] \Big\} \\ \phi_i(t) &= -\big\{\nabla^2 f_i[p_i(t), t]\big\}^{-1} \left\{ \nabla f_i[p_i(t), t] + \frac{\partial}{\partial t} \nabla f_i[p_i(t), t] \right\} \end{aligned} \tag{5}$$

where $\gamma_i \in \mathbb{R}_{>0}$ is a constant control gain. The auxiliary variables $\phi_i(t)$ and $\phi_j(t) \in \mathbb{R}^m$ automatically adjust the gain of the interaction term $\operatorname{sgn}[p_i(t) - p_j(t)]$ for $j \in \mathcal{N}_i(t)$.

*Remark 2:* Algorithm (5) solves the time-varying optimization problem of (2) as a consensus minimization problem with the time-varying team objective function $\sum_{i=1}^{n} f_i[p_i(t), t]$. The term $-\sum_{j \in \mathcal{N}_i(t)} \{[\|\phi_i(t)\|_\infty + \|\phi_j(t)\|_\infty + \gamma_i + \gamma_j] \operatorname{sgn}[p_i(t) - p_j(t)]\}$ is introduced to achieve consensus among the agents, that is $p_i(t) \to p_j(t),\ \forall i, j \in \mathcal{V}$. The auxiliary variable $\phi_i(t)$ is employed to force the consensus state to track the optimal solution $r^*(t)$. Note that the three terms in $\phi_i(t)$, namely, $\{\nabla^2 f_i[p_i(t), t]\}^{-1}$, $\nabla f_i[p_i(t), t]$ and $\partial \nabla f_i[p_i(t), t]/\partial t$, could become unbounded due to the involvement of $p_i(t)$ and $t$. Here, the state-dependent gain $\|\phi_i(t)\|_\infty + \|\phi_j(t)\|_\infty + \gamma_i + \gamma_j$ is used to overcome the possible unboundedness of $\phi_i(t)$.

*Remark 3:* Algorithm (5) is distributed, because each agent only uses information about its own objective function and information communicated by its neighbors. Take agent $i$ as an example. Agent $i$ uses its own information: $p_i(t)$ and the Hessian and gradient information of its objective function $f_i[p_i(t), t]$; as well as information received from its neighbors: $p_j(t) \in \mathbb{R}^m$, $\gamma_j \in \mathbb{R}_{>0}$ and $\|\phi_j(t)\|_\infty \in \mathbb{R}$ for $j \in \mathcal{N}_i(t)$. Moreover, in order to implement Algorithm (5), all agents need to share a common coordinate system. Note that many global coordinate systems exist worldwide, such as GPS.

*Remark 4:* The design of Algorithm (5) is partly motivated by the algorithm given in [19]

$$\begin{aligned} u_i(t) &= -\sum_{j \in \mathcal{N}_i} \beta_{ij}(t)\, \operatorname{sgn}\big[p_i(t) - p_j(t)\big] + \phi_i(t) \\ \dot{\beta}_{ij}(t) &= \|p_i(t) - p_j(t)\|_1, \;\; j \in \mathcal{N}_i \end{aligned} \tag{6}$$

where $\phi_i(t)$ is the same as that in (5). Compared with Algorithm (6), Algorithm (5) has two advantages. First, Algorithm in [19] places a bound on the Hessians and the rate of the change of the gradients of the local objective functions. These requirements can limit the applicable class of objective functions. For example, as stated in Remark 1, a commonly used objective function for robot control and energy minimization is $f_i[p_i(t), t] = [\alpha p_i(t)+b_i(t)]^2$. If $\dot{b}_i(t)$ is unbounded, then it is obvious that it does not satisfy the above requirement. In this brief, we introduce a novel state-dependent control gain design to remove the above requirement. Algorithm (5) can deal with more general objective functions that cannot be handled in [19]. Second, the adaptive control gain $\beta_{ij}(t)$ designed in (6) keeps increasing until the consensus is achieved. Therefore, the control gain might become unnecessarily large. In contrast, the state-dependent gain in this brief is computed such that the above situation is eliminated. The state-dependent control gain approach introduces new theoretical challenges that are the focus of this brief. However, Algorithm (5) has a disadvantage as well. That is, each agent is required to be able to get the information of the variable $\phi_j(t)$ and $\gamma_j$ from its neighbors, which requires the existence of the communication capabilities, while Algorithm (6) can be implemented using only local sensing without the need for the existence of the communication capabilities as long as the relative position $(x_i - x_j)$ between each agent and its neighbors can be measured.

## *B. Convergence Analysis*

This section establishes the asymptotic convergence of system (1) under controller (5) to the optimal solution in (2).

*Lemma 2:* Given (5), using (5) for (1), all the states $p_i(t)$ will achieve consensus in finite time, i.e., there exists a time $T$ such that $\|p_i(t) - p_j(t)\|_2 = 0$ for all $i, j \in \mathcal{V}$ and for all $t > T$.

*Proof:* The main idea of our proof is to show that each corresponding component of the agents' state vectors reaches a consensus separately in finite time. Let $\dot{p}_{ik}(t)$, $p_{ik}(t)$ and $\phi_{ik}(t)$ denote, respectively, the $k$th components of $\dot{p}_i(t)$, $p_i(t)$ and $\phi_i(t)$. Define $A_{1k}(t) \triangleq \{i \mid p_{ik}(t) = \max_{i\in\mathcal{V}}[p_{ik}(t)]\}$, $A_{2k}(t) \triangleq \{i \mid p_{ik}(t) = \min_{i\in\mathcal{V}}[p_{ik}(t)]\}$, $\bar{p}_k(t) \triangleq (1/|A_{1k}(t)|)\sum_{i\in A_{1k}(t)} p_{ik}(t)$ and $\underline{p}_k(t) \triangleq (1/|A_{2k}(t)|)\sum_{i\in A_{2k}(t)} p_{ik}(t)$, where $|A_{1k}(t)| \geq 1$ and $|A_{2k}(t)| \geq 1$ denote, respectively, the cardinality of $A_{1k}(t)$ and $A_{2k}(t)$. Clearly, using (5) for (1), the $k$th component of each $\dot{p}_i(t)$ can be written as

$$\dot{p}_{ik}(t) = \phi_{ik}(t) - \sum_{j\in\mathcal{N}_i(t)} \Big\{\big[\|\phi_i(t)\|_\infty + \|\phi_j(t)\|_\infty + \gamma_i + \gamma_j\big] \times \mathrm{sgn}\big[p_{ik}(t) - p_{jk}(t)\big]\Big\}. \quad (7)$$

We first show that, when $\bar{p}_k(t) \neq \underline{p}_k(t)$, $p_{ik}(t)$ for all $i \in A_{1k}(t)$ are nonincreasing and $p_{ik}(t)$ for all $i \in A_{2k}(t)$ are nondecreasing. Note that even though all elements $p_{ik}(t)$, $i \in A_{1k}(t)$ have the same value, they need not have the same derivative. The proof will proceed by induction to establish a contradiction.

Assume that for agent $l \in A_{1k}(t)$ there exist two time instants $t_1 < t_2$ such that $\dot{p}_{lk}(t) > 0$ for all $t \in [t_1, t_2]$ almost everywhere (i.e., except for some isolated time instants of measure zero).[2] Note that $\|\phi_l(t)\|_\infty + \|\phi_j(t)\|_\infty + \gamma_l + \gamma_j > \phi_{lk}(t)$, and $\mathrm{sgn}[p_{lk}(t) - p_{jk}(t)] = 1$ when $j \in \mathcal{N}_l(t)$ and $p_{jk}(t) \neq p_{lk}(t)$. Therefore, it follows from (7) where $i$ is replaced with $l$ and the fact that $\dot{p}_{lk}(t) > 0$ for all $t \in [t_1, t_2]$ almost everywhere that $\phi_{lk}(t) > 0$ and $p_{jk}(t) = p_{lk}(t)$ for all $j \in \mathcal{N}_l(t)$ and all $t \in [t_1, t_2]$. Further recall that $l \in A_{1k}(t)$ and $\dot{p}_{lk}(t) > 0$ for all $t \in [t_1, t_2]$ almost everywhere. Given these facts, there must exist two time instants $t_3 < t_4$ satisfying $[t_3, t_4] \subseteq [t_1, t_2]$ such that $\dot{p}_{jk}(t) > 0$ for all $j \in \mathcal{N}_l(t)$ and all $t \in [t_3, t_4]$ almost everywhere. Similarly, it can be obtained that $p_{qk}(t) = p_{jk}(t)$ (and hence $p_{lk}(t)$) for all $t \in [t_3, t_4]$ and all $q \in \mathcal{N}_j(t)$ with $j \in \mathcal{N}_l(t)$. Since the graph $\mathcal{G}(t)$ is connected, by induction, we have $p_{ik}(t) = p_{lk}(t)$ for all $i$ and all $t$ within a certain time interval, which contradicts with the assumption that $\bar{p}_k(t) \neq \underline{p}_k(t)$. Thus, $p_{ik}(t)$ for all $i \in A_{1k}(t)$ are nonincreasing when $\bar{p}_k(t) \neq \underline{p}_k(t)$. Similarly, $p_{ik}(t)$ are nondecreasing for all $i \in A_{2k}(t)$ when $\bar{p}_k(t) \neq \underline{p}_k(t)$.

To show that consensus is reached in finite time consider $V(t) = \bar{p}_k(t) - \underline{p}_k(t)$ as a Lyapunov function candidate for all $\bar{p}_k(t) \neq \underline{p}_k(t)$. Note that $V(t) > 0$ when $\bar{p}_k(t) \neq \underline{p}_k(t)$. Based on the above analysis, when $\bar{p}_k(t) \neq \underline{p}_k(t)$, $\dot{p}_{ik}(t) \leq 0$ for all $i \in A_{1k}(t)$ and $\dot{p}_{ik}(t) \geq 0$ for all $i \in A_{2k}(t)$ almost everywhere. Because the graph $\mathcal{G}(t)$ is connected, when $\bar{p}_k(t) \neq \underline{p}_k(t)$, there exists at least a node $\ell \in A_{1k}(t)$ having an edge to a node $j \notin A_{1k}(t)$, implying that $p_{\ell k}(t) > p_{jk}(t)$. Note that here the indices $\ell$ and $j$ might change over time. It follows from (7) that when $\bar{p}_k(t) \neq \underline{p}_k(t)$:

$$\begin{aligned}\dot{p}_{\ell k}(t) &\leq \phi_{\ell k}(t) - \big[\|\phi_\ell(t)\|_\infty + \|\phi_j(t)\|_\infty + \gamma_\ell + \gamma_j\big] \\ &\quad \times \mathrm{sgn}\big[p_{\ell k}(t) - p_{jk}(t)\big] \\ &\leq -\big(\gamma_\ell + \gamma_j\big).\end{aligned}$$

Note that when $\bar{p}_k(t) \neq \underline{p}_k(t)$, $\dot{\bar{p}}_k(t) = (1/|A_{1k}(t)|)\sum_{i\in A_{1k}(t)} \dot{p}_{ik}(t) = (1/|A_{1k}(t)|)[\dot{p}_{\ell k}(t) + \sum_{i\in A_{1k}(t)\setminus\{\ell\}} \dot{p}_{ik}(t)]$. Recall that when $\bar{p}_k(t) \neq \underline{p}_k(t)$, $\dot{p}_{ik}(t) \leq 0$ for all $i \in A_{1k}(t) \setminus \{\ell\}$ almost everywhere. We have $\dot{\bar{p}}_k(t) \leq -(((\gamma_\ell + \gamma_j))/|A_{1k}(t)|) \leq -((2\min_{i\in\mathcal{V}}(\gamma_i))/n - 1)$ almost everywhere. Note that when $\bar{p}_k(t) \neq \underline{p}_k(t)$, $\dot{\underline{p}}_k(t) = (1/|A_{2k}(t)|)\sum_{i\in A_{2k}(t)} \dot{p}_{ik}(t) \geq 0$ almost everywhere. We hence have

$$\dot{V}(t) = \dot{\bar{p}}_k(t) - \dot{\underline{p}}_k(t) \leq -2\min_{i\in\mathcal{V}}(\gamma_i)\Big/(n-1)$$

almost everywhere when $\bar{p}_k(t) \neq \underline{p}_k(t)$. Based on Lebesgue's theory for the Riemann integrability, a function on a compact interval is Riemann integrable if and only if it is bounded and the set of its discontinuous points has measure zero [23]. Therefore, although the time-derivative $\dot{V}(t)$ here is discontinuous at some time points, it is Riemann integrable. Then, we have

$$V(t) - V(0) = \int_0^t \dot{V}(\tau)d\tau \leq -\frac{2t\min_{i\in\mathcal{V}}(\gamma_i)}{n-1}$$

where $t > 0$. It follows that:

$$V(t) \leq V(0) - \Big[2\min_{i\in\mathcal{V}}(\gamma_i)\Big/(n-1)\Big]t. \quad (8)$$

It then can be concluded that $V(t)$ converges to zero in finite time and the convergence time $T$ satisfies $T \leq (((n-1)V(0))/(2\min_{i\in\mathcal{V}}(\gamma_i)))$. That is, consensus is reached

[2]Here, we do not consider the sets of measure zero in $[t_1, t_2]$ on which the derivatives at certain isolated time instants are nonpositive as these sets have no effect on the state value $p_{ik}(t)$.

in finite time and there exists a positive number $T$ such that $p_i(t) = p_j(t)$ for all $t \geq T$ and all $i, j \in \mathcal{V}$. □

*Remark 5:* It follows from (8) that the convergence time $T$ of the consensus process can be made smaller by selecting larger $\gamma_i$. However, if $\gamma_i$ is too large, the chattering phenomenon would become worse due to the discontinuous signum function in (5).

Following is the main result of this section.

*Theorem 1:* If Assumptions 1–3 hold, for system (1) under the controller (5), all the states $p_i(t)$ will converge asymptotically to the optimal solution $r^*(t)$ in (2).

*Proof:* Under Assumptions 1 and 2, it follows from Lemma 2 that the states of all the agents achieve consensus in finite time, i.e., there exists a time $T$ such that $p_i(t) = p_j(t)$ for all $i, j \in \mathcal{V}$ and all $t \geq T$. For $t \geq T$, consider the Lyapunov function candidate

$$V_2(t) = \frac{1}{2}\left\{\sum_{i=1}^{n} \nabla f_i[p_i(t), t]\right\}^T \left\{\sum_{i=1}^{n} \nabla f_i[p_i(t), t]\right\}. \quad (9)$$

It follows from [19, Th. 3.9] that the derivative of $V_2(t)$ is:

$$\begin{aligned}
&\dot{V}_2(t) \\
&= \left\{\sum_{i=1}^{n} \nabla f_i[p_i(t), t]\right\}^T \left\{\sum_{i=1}^{n} \frac{d\nabla f_i[p_i(t), t]}{dt}\right\} \\
&= -\left\{\sum_{i=1}^{n} \nabla f_i[p_i(t), t]\right\}^T \Bigg[\sum_{i=1}^{n} (\nabla f_i[p_i(t), t] \\
&\quad + \nabla^2 f_i[p_i(t), t] \sum_{j \in \mathcal{N}_i(t)} \{[\|\phi_i(t)\|_\infty + \|\phi_j(t)\|_\infty + \gamma_i + \gamma_j] \\
&\quad \times \operatorname{sgn}[p_i(t) - p_j(t)]\})\Bigg]. \qquad (10)
\end{aligned}$$

Note from Assumptions 1 and 3 that the graph $\mathcal{G}(t)$ is undirected and $\nabla^2 f_i[p_i(t), t] = \nabla^2 f_j[p_j(t), t]$ if $p_i(t) = p_j(t)$ for all $i, j \in \mathcal{V}$, it follows that for all $t \geq T$:

$$\begin{aligned}
&\sum_{i=1}^{n} \nabla^2 f_i[p_i(t), t] \sum_{j \in \mathcal{N}_i(t)} \{[\|\phi_i(t)\|_\infty + \|\phi_j(t)\|_\infty + \gamma_i + \gamma_j] \\
&\quad \times \operatorname{sgn}[p_i(t) - p_j(t)]\} \\
&= \nabla^2 f_i[p_i(t), t] \sum_{i=1}^{n} \sum_{j \in \mathcal{N}_i(t)} \{[\|\phi_i(t)\|_\infty + \|\phi_j(t)\|_\infty + \gamma_i + \gamma_j] \\
&\quad \times \operatorname{sgn}[p_i(t) - p_j(t)]\} = \mathbf{0}_m.
\end{aligned}$$

Then we have for all $t \geq T$

$$\begin{aligned}
\dot{V}_2(t) &= -\left\{\sum_{i=1}^{n} \nabla f_i[p_i(t), t]\right\}^T \left\{\sum_{i=1}^{n} \nabla f_i[p_i(t), t]\right\} \\
&= -2V_2(t) \qquad (11)
\end{aligned}$$

which indicates that $V_2(t) = e^{-2t} V_2(T)$ for all $t \geq T$. It can be concluded that $\lim_{t\to\infty} V_2(t) = 0$, and thus $\lim_{t\to\infty} \sum_{i=1}^{n} \nabla f_i[p_i(t), t] = \mathbf{0}_m$. Due to Assumption 3, the Lyapunov function $V_2(t)$ defined in (9) has a unique time-varying global minimum $r^*(t)$ such that $\{\sum_{i=1}^{n} \nabla f_i[r^*(t), t]\}^T \{\sum_{i=1}^{n} \nabla f_i[r^*(t), t]\} = 0$. Recall that $p_i(t) = p_j(t)$ for all $i, j \in \mathcal{V}$ and all $t \geq T$, which in turn implies that all $p_i(t)$ will converge to the optimal solution $r^*(t)$ in (2) based on Lemma 1. □

*Remark 6:* In some robotic applications, it is desirable for the agents to come into a formation, while the center of the formation moves along the optimal trajectory. To achieve this goal, we introduce a deviation vector $\delta_i(t)$ for each agent $i$ and replace $p_i(t)$ in (5) with $p_i(t) - \delta_i(t)$. It follows that Algorithm (5) will guarantee that $p_i(t) - \delta_i(t)$ converges to the optimal trajectory, which in turn implies that $p_i(t) - p_j(t)$ converges to $\delta_i(t) - \delta_j(t)$. Here, $\delta_i(t) - \delta_j(t)$ defines the desired relative position from agent $j$ to agent $i$ in the formation. That is, the agents will be able to converge to the optimal trajectory with the deviation vector $\delta_i(t)$. The analysis follows directly by letting $p_i(t) - \delta_i(t)$ play the role of $p_i(t)$ in the previous proof.

## IV. Distributed Time-Varying Optimization With Time-Varying Linear Equality Constraints

In this section, we extend the results in Section III-A to take into account common time-varying linear equality constraints. The goal is to design $u_i(t)$ such that all the agents cooperatively find the optimal solution $\tilde{r}^*(t) \in \mathbb{R}^m$ defined as

$$\begin{aligned}
\tilde{r}^*(t) \;=\; &\arg\min_{\tilde{r}(t)} \left\{\sum_{i=1}^{n} f_i[\tilde{r}(t), t]\right\} \\
&\text{Subject to } A(t)\tilde{r}(t) = b(t) \qquad (12)
\end{aligned}$$

where $A(t) \in \mathbb{R}^{q\times m}$ and $b(t) \in \mathbb{R}^q$ are the equality constraint functions. Note that $A(t)p_i(t) = A(t)\tilde{r}(t)$ for all $i \in \mathcal{V}$ and $\sum_{i=1}^{n} f_i[p_i(t), t] = \sum_{i=1}^{n} f_i[\tilde{r}(t), t]$, if $p_i(t) = p_j(t) = \tilde{r}(t)$ for all $i, j \in \mathcal{V}$. Therefore, the above problem (12) is equivalent to finding the optimal solution $p^*(t) \in \mathbb{R}^{m*n}$ which is defined as

$$\begin{aligned}
p^*(t) = &\arg\min_{p(t)} \left\{\sum_{i=1}^{n} f_i[p_i(t), t]\right\} \\
&\text{Subject to } A(t)p_i(t) = b(t) \quad \forall i \in \mathcal{V} \\
&\text{and} \quad p_i(t) = p_j(t) \quad \forall i, j \in \mathcal{V}. \quad (13)
\end{aligned}$$

Here, the goal is that each state $p_i(t)$ converges to the optimal solution $\tilde{r}^*(t)$, that is

$$\lim_{t\to\infty} [p_i(t) - \tilde{r}^*(t)] = \mathbf{0}_m. \quad (14)$$

Here, agent $i$ only has access to its own objective function $f_i[p_i(t), t]$, the constraint function $A(t)$ and $b(t)$, its own state $p_i(t)$, and information received from its neighbors $j \in \mathcal{N}_i(t)$.

We need an additional assumption.

*Assumption 4:* The number of equality constraints is less than the dimension of the state variable $p_i$, i.e., $q < m$. Moreover, the rows of $A(t)$ are linearly independent for all $t \geq 0$, i.e., $\operatorname{rank}[A(t)] = q$.

Assumption 4 ensures that the constraint function has infinitely many solutions at each $t \geq 0$.

### A. Distributed Algorithm Design

In this section, we derive a distributed control algorithm such that (14) holds. The Lagrangian function of problem (12) is

$$L[\tilde{r}(t), t] = \sum_{i=1}^{n} f_i[\tilde{r}(t), t] + \nu^T(t)[A(t)\tilde{r}(t) - b(t)] \quad (15)$$

where $\nu(t) \in \mathbb{R}^q$ is the Lagrangian multiplier. Note that the function $L[\tilde{r}(t), t]$ is strongly convex in $\tilde{r}(t)$ and concave in $\nu(t)$. Based on the KKT conditions, we know that the optimal solution of problem (12) must satisfy

$$\begin{aligned} \sum_{i=1}^{n} \nabla f_i\big[\tilde{r}^*(t), t\big] + A^T(t)\nu^*(t) &= \mathbf{0}_m \\ A(t)\tilde{r}^*(t) - b(t) &= \mathbf{0}_q. \end{aligned} \quad (16)$$

Let $\lambda_i(t) \in \mathbb{R}^q$ be local internal states playing the role of the local counterparts of the global Lagrangian multiplier $\nu(t)$. Then the optimal solution in (16) is equivalent to

$$\lim_{t\to\infty} \|p_i(t) - p_j(t)\|_2 = 0 \quad \forall i, j \in \mathcal{V} \quad (17a)$$

$$\lim_{t\to\infty} \|\lambda_i(t) - \lambda_j(t)\|_2 = 0 \quad \forall i, j \in \mathcal{V} \quad (17b)$$

$$\sum_{i=1}^{n} \nabla f_i[p_i(t), t] + A^T(t)\lambda_i(t) = \mathbf{0}_m \quad (17c)$$

$$A(t)p_i(t) - b(t) = \mathbf{0}_q \quad \forall i \in \mathcal{V}. \quad (17d)$$

The controller for agent $i$ is defined as

$$\begin{aligned} u_i(t) &= \psi_i^F(t) - \sum_{j\in\mathcal{N}_i(t)} \big\{\big[\|\psi_i(t)\|_\infty + \|\psi_j(t)\|_\infty + \gamma_i + \gamma_j\big] \\ &\quad \times \mathrm{sgn}\big[p_i(t) - p_j(t)\big]\big\} \\ \dot{\lambda}_i(t) &= \psi_i^L(t) - \sum_{j\in\mathcal{N}_i(t)} \big\{\big[\|\psi_i(t)\|_\infty + \|\psi_j(t)\|_\infty + \gamma_i + \gamma_j\big] \\ &\quad \times \mathrm{sgn}\big[\lambda_i(t) - \lambda_j(t)\big]\big\} \\ \psi_i(t) &= -\big\{\nabla^2 \tilde{L}_i[s_i(t), t]\big\}^{-1}\left\{\nabla \tilde{L}_i[s_i(t), t] + \frac{\partial}{\partial t}\nabla \tilde{L}_i[s_i(t), t]\right\} \end{aligned} \quad (18)$$

where $\tilde{L}_i[s_i(t), t] = f_i[p_i(t), t] + \lambda_i^T(t)[A(t)p_i(t) - b(t)]$ with $s_i(t) \in \mathbb{R}^{m+q} = [p_i^T(t), \lambda_i^T(t)]^T$, and $\psi_i^F(t)$ and $\psi_i^L(t)$ denote, respectively, the first $m$ components and the last $q$ components of the vector $\psi_i \in \mathbb{R}^{m+q}$. It follows from Assumptions 3 and 4 that $\nabla^2 \tilde{L}_i[s_i(t), t]$ is invertible [22]. There are four conditions in (17). In Algorithm (18), the term $-\sum_{j\in\mathcal{N}_i(t)}\{[\|\psi_i(t)\|_\infty + \|\psi_j(t)\|_\infty + \gamma_i + \gamma_j] \ \mathrm{sgn}[p_i(t) - p_j(t)]\}$ is introduced to ensure that all the agents achieve consensus on states $p_i(t)$, i.e., condition (17a). The term $-\sum_{j\in\mathcal{N}_i(t)}\{[\|\psi_i(t)\|_\infty + \|\psi_j(t)\|_\infty + \gamma_i + \gamma_j] \ \mathrm{sgn}[\lambda_i(t) - \lambda_j(t)]\}$ is employed to ensure that all the agents achieve consensus on $\lambda_i(t)$, i.e., condition (17b). The term $\psi_i$ is introduced to achieve the optimal condition given by (17c) and (17d).

*Remark 7:* It is worth mentioning that the discontinuous signum function in (5) and (18) might cause chattering behavior. In practice, a simple and useful way to solve this oscillating problem is to approximate the signum function using a continuous function in a region called the boundary layer around the sliding surface [25]. For example, we can replace the signum function with the function

$$h(z) = \frac{z}{||z||_2 + \epsilon}$$

where $z \in \mathbb{R}^m$ and $\epsilon$ is a positive constant. Despite the drawback of the chattering effect, sliding-mode control has its own merits such as fast convergence and robustness against system uncertainties and disturbances.

### *B. Convergence Analysis*

This section establishes the asymptotic convergence of system (1) under controller (18) to the optimal solution in (12).

*Theorem 2:* If Assumptions 1–4 hold, for system (1) under controller (18), then all the states $p_i(t)$ will converge asymptotically to the optimal solution $\tilde{r}^*(t)$ in (12).

*Proof:* First, we show that the conditions given by (17a) and (17b) can be achieved. Applying controller (18) to system (1) leads to

$$\begin{aligned} \dot{s}_i(t) = \psi_i(t) - \sum_{j\in\mathcal{N}_i(t)} &\big\{\big[\|\psi_i(t)\|_\infty + \|\psi_j(t)\|_\infty + \gamma_i + \gamma_j\big] \\ &\times \mathrm{sgn}\big[s_i(t) - s_j(t)\big]\big\}. \end{aligned} \quad (19)$$

The desired result follows under Assumptions 1 and 2 by letting $\dot{s}_i(t)$, $s_i(t)$, and $\psi_i(t)$, respectively, play the role of $\dot{p}_i(t)$, $p_i(t)$, and $\phi_i(t)$ in the proof of Lemma 2. That is, consensus on $s_i(t)$ will be achieved in finite time. Then there exists a time $T$ such that $s_i(t) = s_j(t)$ for all $t > T$ and all $i, j \in \mathcal{V}$ and thus $p_i(t) = p_j(t)$ and $\lambda_i(t) = \lambda_j(t)$ for all $t > T$ and all $i, j \in \mathcal{V}$.

Next we show that conditions (17c) and (17d) will be achieved. The gradient and Hessian of the function $\tilde{L}_i[s_i(t), t]$ with respect to $s_i(t)$ are

$$\begin{aligned} \nabla \tilde{L}_i[s_i(t), t] &= \begin{bmatrix} \nabla f_i[p_i(t), t] + A^T(t)\lambda_i(t) \\ A(t)p_i(t) - b(t) \end{bmatrix} \\ \nabla^2 \tilde{L}_i[s_i(t), t] &= \begin{bmatrix} \nabla^2 f_i[p_i(t), t] & A^T(t) \\ A(t) & \mathbf{0}_q \end{bmatrix} \end{aligned} \quad (20)$$

where $\nabla^2 \tilde{L}_i[s_i(t), t]$ is invertible due to Assumptions 3 and 4. It is obvious that if $\nabla^2 f_i[p_i(t), t] = \nabla^2 f_j[p_j(t), t]$ under $p_i(t) = p_j(t)$ for all $i, j \in \mathcal{V}$, then it holds that $\nabla^2 \tilde{L}_i[s_i(t), t] = \nabla^2 \tilde{L}_j[s_j(t), t]$ under $p_i(t) = p_j(t)$ for all $i, j \in \mathcal{V}$. Consider the Lyapunov function candidate $V_3(t) = (1/2)\{\sum_{i=1}^n \nabla \tilde{L}_i[s_i(t), t]\}^T\{\sum_{i=1}^n \nabla \tilde{L}_i[s_i(t), t]\}$. Similar to the analysis in Theorem 1, it can be concluded that as $t \to \infty$, $V_3(t) \to 0$, we have $\lim_{t\to\infty} \sum_{i=1}^n \nabla \tilde{L}_i[s_i(t), t] = \mathbf{0}_{m+q}$ and thus $\lim_{t\to\infty} \sum_{i=1}^n \nabla f_i[p_i(t), t] + A^T(t)\lambda_i(t) = \mathbf{0}_m$ and $\lim_{t\to\infty} \sum_{i=1}^n A(t)p_i(t) - b(t) = \mathbf{0}_q$ based on the definition in (20). The conclusion of the theorem then follows by combining the above statements. □

## V. Simulation Results

The simulation results in this section illustrate the effectiveness of the theoretical results obtained in Sections III and IV. Assume that there are six agents ($n = 6$) in 2-D ($m = 2$). The network topology shown in Fig. 1 is undirected and connected. Let $p_i(t) = [x_i(t), y_i(t)]^T \in \mathbb{R}^2$ denote the state (position) of agent $i$, where $x_i(t) \in \mathbb{R}$ (respectively, $y_i(t) \in \mathbb{R}$) denotes the position of agent $i$ in the $x$-coordinate (respectively, $y$-coordinate).

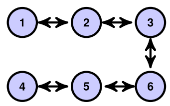


Fig. 1. Undirected graph representing the communication topology between agents.

First, we show the simulation result using Algorithm (5). Let $r(t) = [r_x(t), r_y(t)]^T$ and consider the following unconstrained optimization problem:

$$r^*(t) \in \mathbb{R}^2 = \operatorname{argmin} \sum_{i=1}^{n} \Big\{ [r_x(t) - 0.1(0.25 + 0.5i)t]^2 + \big[r_y(t) - 0.1(0.25 + 0.5i)t\big]^2 \Big\}. \quad (21)$$

This problem is an instance of (2). The goal is that each state $p_i(t)$ converges to the optimal solution $r^*(t)$ defined in (21). The intuition of problem (21) is from the multirobot target tracking problem, where $[0.1(0.25+0.5i)t, 0.1(0.25+0.5i)t]^T$ encodes the tracking target of agent $i$. Here, multiple robots aim to cooperatively find the optimal position that is close to all the targets. Choose $\gamma_i = 1,\ \forall i \in \mathcal{V}$. The proof of Lemma 2 proves that the maximum time for consensus to be achieved satisfies $T \le (((n-1)V(0))/(2\min_{i\in\mathcal{V}}(\gamma_i)))$. Therefore, with $\gamma_i = \gamma_j = \bar{\gamma}$, the time to achieve consensus, for any given set of initial conditions, is inversely proportional to $\bar{\gamma}$.

The initial states of the agents are chosen as $p_1(0) = [0, 1]^T$, $p_2(0) = [0.5, 1]^T$, $p_3(0) = [0.5, 0.5]^T$, $p_4(0) = [0, 0.5]^T$, $p_5(0) = [-0.5, 0]^T$, $p_6(0) = [0, 0]^T$. The agents' states and the optimal trajectory in the $(x, t)$ [respectively, $(y, t)$] coordinates are shown in Fig. 2(a) [respectively, Fig. 2(b)]. The red dashed line is the optimal solution and the other solid lines are the trajectories of all agents' states. It is clear that all the agents track the optimal trajectory asymptotically (i.e., $\lim_{t\to\infty} \|p_i(t) - r^*(t)\|_2 = 0$ for all $i \in \mathcal{V}$) which is consistent with Theorem 1. We introduce a deviation vector to (5) by replacing $p_i$ with $p_i - \delta_i$ (see Remark 3.4). Here, $\delta_1 = [0.5, 0.5]^T$, $\delta_2 = [0.5, 0]^T$, $\delta_3 = [0.5, -0.5]^T$, $\delta_4 = [-0.5, 0.5]^T$, $\delta_5 = [-0.5, 0]^T$, $\delta_6 = [-0.5, -0.5]^T$. In Fig. 3, the blue circles present a snapshot of all the agents' initial positions and the blue crosses present two snapshots of all the agents at 4.5 and 9 s, respectively. Fig. 3 shows each agent's trajectories with the deviation vectors introduced (blue dashed lines), the center position of all the agents (black solid line), and the optimal trajectory (red dashed line) in the $(x, y, t)$ coordinates. Note that the agents asymptotically form a rectangle formation with its center tracking the optimal trajectory, implying that $\lim_{t\to\infty} \|p_i(t) - \delta_i - r^*(t)\|_2 = 0$ for all $i \in \mathcal{V}$.

Second, we show a simulation result using Algorithm (18). Let $\tilde{r} = [\tilde{r}_x, \tilde{r}_y]^T$ and consider the following constrained optimization problem:

$$\tilde{r}^*(t) \in \mathbb{R}^2 = \operatorname{argmin} \sum_{i=1}^{n} \Big\{ [\tilde{r}_x(t) - it]^2 + \big[\tilde{r}_y(t) - it\big]^2 \Big\}$$
$$\text{Subject to } \cos(t)\tilde{r}_x(t) + \sin(t)\tilde{r}_y(t) = 3. \quad (22)$$

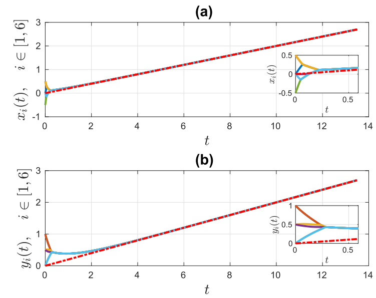


Fig. 2. Simulation results showing state convergence using controller (5).

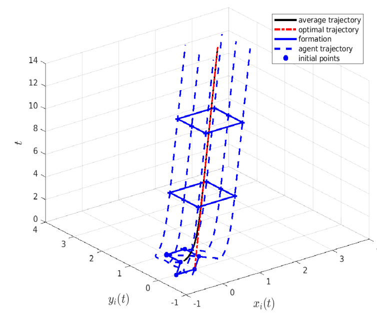


Fig. 3. Simulation results using controller (5) with the deviation vectors introduced.

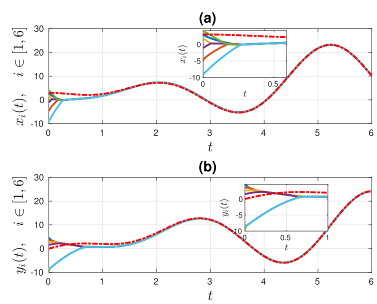


Fig. 4. Simulation results showing state convergence to the optimal solution using controller (18).

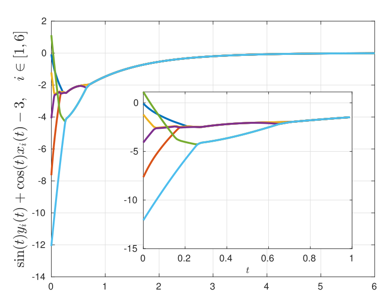

Fig. 5. Simulation results showing convergence of the constraint using controller (18).

Problem (22) is an instance of Problem (12). The goal here is that each state $p_i(t)$ converges to the optimal solution $\tilde{r}^*(t)$ defined in (22). The intuition of Problem (22) is also from the multirobot target tracking problem, where $[i * t, i * t]^T$ encodes the tracking signal of agent $i$ and the function $\cos(t)\tilde{r}_x(t) + \sin(t)\tilde{r}_y(t) = 3$ represents some

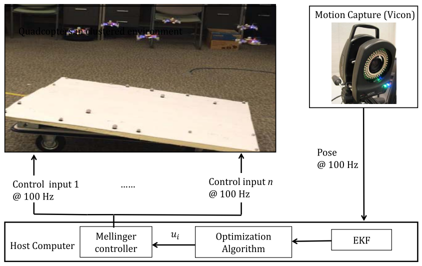


Fig. 6. Experimental setup and information flow.

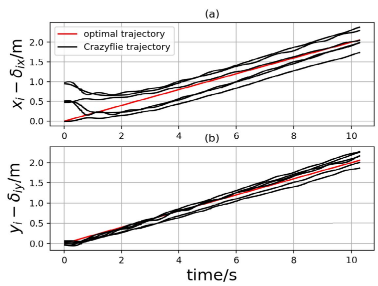


Fig. 7. Experimental results using controller (5).

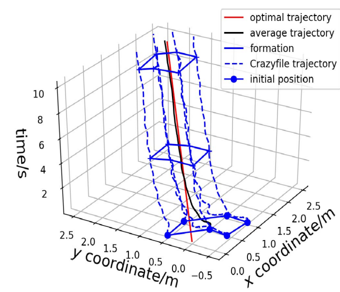


Fig. 8. Experimental results using controller (5).

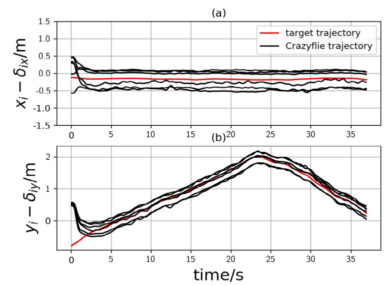


Fig. 9. Experimental results using controller (5).

physical constraints for the robots. For this simulation, $\forall i \in \mathcal{V}$, we select $\gamma_i = 5$ and choose the initial states $x_i(0)$ and $y_i(0)$ randomly from the range $[-10, 10]$. The state trajectories of the agents (solid lines) and the optimal trajectory $\tilde{r}^*(t)$ defined in (22) (red dashed line) are shown in Fig. 4. It is clear that all the agents track the optimal trajectory asymptotically, i.e., $\lim_{t\to\infty} \|p_i(t) - \tilde{r}^*(t)\|_2 = 0$ for all $i \in \mathcal{V}$. Fig. 5 shows convergence of the constraint for each agent. We can see that $\cos(t)x_i(t) + \sin(t)y_i(t) - 3$ converge to zero asymptotically for all the agents, which is consistent with Theorem 2.

## VI. Experimental Results

In this section, the algorithm designed in Section III is applied to the multiagent formation control problem and the multiagent moving target tracking problem and is tested in experiments. The experiments are conducted in the Cooperative Vehicle Networks (COVEN) Laboratory at the University of California, Riverside with six Crazyflie 2.0 quadrotors [26] in an $5 \times 5$ m$^2$ indoor environment covered by a VICON motion capture system [28]. The Crazyflies are controlled by the velocity commands (i.e., the control signals $u_i(t)$ are the velocity commands that are sent to the Crazyflies) such that their dynamics follow the single-integrator system given by (1). The experimental setup is illustrated in Fig. 6. In this experiment, the control system is divided into two parts, namely, high level and low level. The high-level control involves the setup of the network topology, implementation of the distributed optimization algorithm, and generation of the velocity commands $u_i(t)$. The host computer is used to run the high-level controller due to the fact that the Crazyflies used in the experiments do not have sufficient computation capability to run the controller in real-time. A VICON motion capture system coupled with the extended Kalman filter is used to estimate the positions of each agent. The host computer requests the information packet from the Vicon system every 0.01 s. The low-level control is responsible for achieving the velocity commands (using the Mellinger controller [27]). The host computer sends control commands to the Crazyflies every 0.01 s. The restrictions of a distributed environment are fully considered and the distributed network topology defined in Fig. 1 is emulated. We establish six nodes under the robotics operating system (ROS) to control the six Crazyflies in parallel.

### A. Multiagent Formation Control

First, the distributed time-varying optimization algorithm given by (5) is implemented experimentally to solve problem (21). The desired deviations from the optimal trajectory and the Crazyflies' initial positions have the same values as in Section V. Fig. 7(a) [respectively, Fig. 7(b)] shows the six Crazyflies' normalized positions (i.e., $p_i(t) - \delta_i$) and the optimal trajectory [i.e., $r^*(t)$ defined in (21)] in the $(x, t)$ coordinate (respectively, $(y, t)$ coordinate). The black solid lines are the normalized positions of each Crazyflie. The red solid line is the optimal trajectory. Here $\delta_i = [\delta_{ix}, \delta_{iy}]^T$. Based on Theorem 1, all $p_i(t) - \delta_i$ should converge to the optimal trajectory asymptotically, i.e., $\lim_{t\to\infty} \|p_i(t) - \delta_i - \hat{r}^*(t)\|_2 = 0$. This is achieved within a tracking accuracy of 0–2 cm. Various factors from the experiment might explain the tracking error: communication time-delay within the VICON system, failure to perfectly achieve the velocity commands, or interaction forces among the Crazyflies. Tracking errors of 2 cm are similar to those experiences in other multi-Crazyflie experiments [26]. The trajectories of all the Crazyflies (blue dashed lines), the center position of all the Crazyflies (black solid line) and the optimal trajectory (red solid line) in the $(x, y, t)$ coordinates are shown in Fig. 8, where the blue circles present a snapshot of all the Crazyflies' initial positions. The blue crosses present two snapshots of all the Crazyflies at 4.5 and 9 s, respectively. As can be seen, the center of the Crazyflies' positions tracks the optimal trajectory with small tracking errors (about 0.05 cm) while the Crazyflies converge to the desired formation. When considering the center of all the Crazyflies' positions, the tracking gaps caused by the interaction forces among them should cancel.

### B. Multiagent Moving Target Tracking

In this section, we solve the moving target tracking problem using Algorithm (5). More precisely, the moving target tracking problem can be formulated as the following convex optimization problem:

$$\text{minimize} \quad \frac{1}{2}\sum_{i=1}^{n} \| p_i(t) - T_i(t)\|_2^2 \tag{23}$$

where $p_i(t)$ is the position of robot $i$, and $T_i(t)$ is the position of the moving target sensed by agent $i$. Due to the sensing capability limitation of each agent, the position of the moving target sensed by different robots can be different. It is obvious that the optimal trajectory of problem (23) is $(1/n)\sum_{i=1}^{n} T_i(t)$.

In our experiment, we let six Crazyflies track a moving whiteboard (see Fig. 6). The whiteboard is placed on a cart that is dragged by a person to move it around. There are six marked areas located in the four corners and the middle of the two long edges on the whiteboard. Each area is identified by three markers. The center position of each marked area (i.e., the center of three markers in the area) is sent to one assigned Crazyflie, representing the position of the moving target (whiteboard) sensed by that Crazyflie [i.e., $T_i(t)$ in (23)]. Essentially each Crazyflie senses a different biased position of the whiteboard. The Crazyflies obtain their target's positions from the VICON system and calculate their targets' velocities [i.e., $\dot{T}_i(t)$ in (23)] based on the position data received between consecutive camera frames. We apply controller (5) with the same deviation vectors as those in Section V introduced to the multirobot moving target tracking problem given by (23). In the experiment, we move the cart around and let the Crazyflies track the whiteboard while maintaining the desired formation shape. Fig. 9(a) [respectively, Fig. 9(b)] shows the six Crazyflies' normalized positions represented by $p_i(t) - \delta_i$ and the moving whiteboard's center position represented by $(1/6)\sum_{i=1}^{6} T_i(t)$ in the $(x, t)$ [respectively, $(y, t)$] coordinate. The black lines are the normalized positions of each Crazyflie. The red line is the center position of the whiteboard. Fig. 10 shows the trajectories of all the Crazyflies (blue dashed lines), the center position of all the six Crazyflies (black solid line), and the center position of the moving whiteboard (red solid line) in the $(x, y, t)$ coordinate, where the blue circles present a snapshot of all the Crazyflies' initial positions. The blue crosses present two snapshots of all the Crazyflies at 15 and 28 s, respectively. It can be seen that the six Crazyflies work together to estimate and track the center position of the moving whiteboard with small tracking errors successfully. The tracking error between each Crazyflie's actual position $p_i(t)$

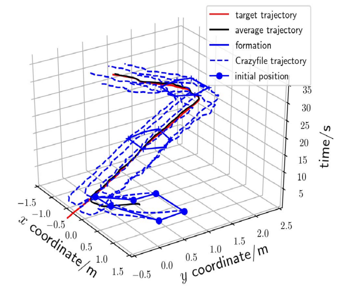


Fig. 10. Experimental results using controller (18).

and its desired position $(1/6)\sum_{i=1}^{6} T_i(t) + \delta_i$ is up to 2 cm, and the tracking error between the average trajectory of all the Crazyflies and the target's trajectory is up to 0.05 cm. The tracking errors are acceptable based on the error analysis in Section VI-A.

## VII. Conclusion

In this brief, we have studied the distributed continuous-time optimization problem with time-varying objective functions. The goal is for a set of networked agents to cooperatively track the time-varying optimal solution that minimizes the summation of all the local time-varying objective functions, where each agent has only local information and local interactions. We have considered the unconstrained optimization case and the case with a common time-varying equality constraint. This brief proposes and analyzes two distributed algorithms coupled with state-dependent control gains for the considered problems. This brief proves that each algorithm yields asymptotic convergence to the optimal solution under reasonable assumptions. Both numerical simulation results and experimental results have been given to illustrate the theoretical algorithms.

## References

[1] T. Yang *et al.*, "A survey of distributed optimization," *Annu. Rev. Control*, vol. 47, pp. 278–305, 2019.
[2] P. Lin, W. Ren, and J. A. Farrell, "Distributed continuous-time optimization: Nonuniform gradient gains, finite-time convergence, and convex constraint set," *IEEE Trans. Autom. Control*, vol. 62, no. 5, pp. 2239–2253, May 2017.
[3] Z. Li, Z. Wu, Z. Li, and Z. Ding, "Distributed optimal coordination for heterogeneous linear multiagent systems with event-triggered mechanisms," *IEEE Trans. Autom. Control*, vol. 65, no. 4, pp. 1763–1770, Apr. 2020.
[4] Z. Qiu, S. Liu, and L. Xie, "Distributed constrained optimal consensus of multi-agent systems," *Automatica*, vol. 68, pp. 209–215, Jun. 2016.
[5] J. Lu and C. Y. Tang, "Zero-gradient-sum algorithms for distributed convex optimization: The continuous-time case," *IEEE Trans. Autom. Control*, vol. 57, no. 9, pp. 2348–2354, Sep. 2012.
[6] Z. Feng and G. Hu, "Finite-time distributed optimization with quadratic objective functions under uncertain information," in *Proc. IEEE Conf. Decis. Control*, Melbourne, VIC, Australia, Dec. 2017, pp. 208–213.
[7] S. Liang, X. Zeng, and Y. Hong, "Distributed nonsmooth optimization with coupled inequality constraints via modified lagrangian function," *IEEE Trans. Autom. Control*, vol. 63, no. 6, pp. 1753–1759, Jun. 2018.
[8] Y. Zhu, W. Yu, G. Wen, G. Chen, and W. Ren, "Continuous-time distributed subgradient algorithm for convex optimization with general constraints," *IEEE Trans. Autom. Control*, vol. 64, no. 4, pp. 1694–1701, Apr. 2019.

[9] S. Yang, Q. Liu, and J. Wang, "A multi-agent system with a proportional-integral protocol for distributed constrained optimization," *IEEE Trans. Autom. Control*, vol. 62, no. 7, pp. 3461–3467, Jul. 2017.

[10] M. Zhu and S. Martinez, "On distributed optimization under inequality and equality constraints via penalty primal-dual methods," in *Proc. Amer. Control Conf.*, Baltimore, MD, USA, Jun. 2010, pp. 2434–2439.

[11] J. Cortés and S. K. Niederländer, "Distributed coordination for non-smooth convex optimization via saddle-point dynamics," *J. Nonlinear Sci.*, vol. 29, no. 4, pp. 1247–1272, Aug. 2019.

[12] P. Lin, W. Ren, Y. Song, and J. A. Farrell, "Distributed optimization with the consideration of adaptivity and finite-time convergence," in *Proc. Amer. Control Conf.*, Portland, OR, USA, Jun. 2014, pp. 3177–3182.

[13] S. G. Lee, Y. Diaz-Mercado, and M. Egerstedt, "Multirobot control using time-varying density functions," *IEEE Trans. Robot.*, vol. 31, no. 2, pp. 489–493, Apr. 2015.

[14] D. Verscheure, B. Demeulenaere, J. Swevers, J. De Schutter, and M. Diehl, "Time-optimal path tracking for robots: A convex optimization approach," *IEEE Trans. Autom. Control*, vol. 54, no. 10, pp. 2318–2327, Oct. 2009.

[15] W. Su, "Traffic engineering and time-varying convex optimization," Ph.D. dissertation, Dept. Elect. Eng., Pennsylvania State Univ., Philadelphia, PA, USA, 2009.

[16] Y. Tang, "Time-varying optimization and its application to power system operation," Ph.D. dissertation, Dept. Elect. Eng., California Inst. Technol., Pasadena, CA, USA, 2019.

[17] C. Sun, M. Ye, and G. Hu, "Distributed time-varying quadratic optimization for multiple agents under undirected graphs," *IEEE Trans. Autom. Control*, vol. 62, no. 7, pp. 3687–3694, Jul. 2017.

[18] M. Hosseinzadeh, E. Garone, and L. Schenato, "A distributed method for linear programming problems with box constraints and time-varying inequalities," *IEEE Control Syst. Lett.*, vol. 3, no. 2, pp. 404–409, Apr. 2019.

[19] S. Rahili and W. Ren, "Distributed continuous-time convex optimization with time-varying cost functions," *IEEE Trans. Autom. Control*, vol. 62, no. 4, pp. 1590–1605, Apr. 2017.

[20] B. Huang, Y. Zou, Z. Meng, and W. Ren, "Distributed time-varying convex optimization for a class of nonlinear multiagent systems," *IEEE Trans. Autom. Control*, vol. 65, no. 2, pp. 801–808, Feb. 2020.

[21] M. Fazlyab, S. Paternain, V. M. Preciado, and A. Ribeiro, "Prediction-correction interior-point method for time-varying convex optimization," *IEEE Trans. Autom. Control*, vol. 63, no. 7, pp. 1973–1986, Jul. 2018.

[22] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, 2004.

[23] T. M. Apostol, *Mathematical Analysis*. Boston, MA, USA: Addison-Wesley, 1974.

[24] H. K. Khalil and W. G. Jessy, *Nonlinear Systems*. Upper Saddle River, NJ, USA: Prentice-Hall, 2002.

[25] C. Edwards and S. Sarah, *Sliding Mode Control: Theory and Applications*. Boca Raton, FL, USA: CRC Press, 1998.

[26] J. A. Preiss, W. Honig, G. S. Sukhatme, and N. Ayanian, "Crazyswarm: A large nano-quadcopter swarm," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, Singapore, May 2017, pp. 3299–3304.

[27] D. Mellinger and V. Kumar, "Minimum snap trajectory generation and control for quadrotors," in *Proc. IEEE Int. Conf. Robot. Autom.*, Shanghai, China, May 2011, pp. 2520–2525.

[28] *Vicon Motion Capture System*. Accessed: Jun. 2017. [Online]. Available: https://www.vicon.com