Abstract—We study the problem of enforcing spatial and temporal (spatiotemporal) constraints in control of multiagent systems with uncertainties. We present a new distributed control algorithm for achieving both ii) a user-defined system performance by limiting the worst-case difference between the agent state trajectories and their corresponding reference state trajectories (spatial constraint) and ii) a user-defined finite-time convergence (temporal constraint). Specifically, the presented algorithm effectively addresses these spatiotemporal constraints without the need for a strict knowledge of agent uncertainties upper bounds and without relying on agent initial conditions. The efficacy of the proposed distributed control algorithm is further illustrated in a numerical example.

I. INTRODUCTION

In many real-world applications, multiagent systems are subject to spatial and temporal (spatiotemporal) constraints. Specifically, spatial constraints usually result from critical structural limitations; hence, a control algorithm need to keep each agent trajectories close enough to their corresponding reference model trajectories, which denotes agent’s ideal responses. As it is well-known, (model reference) adaptive control methods are effective to handle system uncertainties; yet, their (worst-case) performance bounds are often very conservative and not user-defined (see [1] for details). To this end, the studies in, for example, [1–7] propose adaptive control approaches to assign user-defined bounds for a-priori guaranteeing desired performance. However, they make no attempt to address temporal constraints.

Temporal constraints commonly arise from time-critical real-world applications, where it is important to complete a given objective over a (desired) time interval. Yet, most of the existing research results on finite-time control provide a convergence time that depends on the initial conditions of considered physical system [8, 9]. For addressing this drawback, control approaches with fixed-time convergence properties (see, for example, [10, 11]) and with predefined-time convergence properties are studied (see, for example, [12, 13]). In these approaches, however, either the calculated convergence time upper bound do not globally hold for all initial conditions and they can be conservative [10] or they may need a knowledge of uncertainty upper bounds (also see introduction sections of [14, 15] for additional details). Recently, the studies in [14–17] present distributed control algorithms using a time transformation method for guaranteeing the completion of a given control objective at $T$ seconds. Note that this $T$ denoting a user-defined finite time convergence parameter does neither depend on agents’ initial conditions nor the knowledge of system uncertainty upper bounds. Yet, [14–17] do not address spatial constraints; therefore, the presence of system uncertainties may excessively deviate agents from their desired, ideal responses especially in transient time.

In this paper, we study the control of uncertain multiagent systems with spatiotemporal constraints. We present a new distributed control algorithm for achieving both i) a user-defined system performance by limiting the worst-case difference between the agent state trajectories and their corresponding reference state trajectories (spatial constraints) and ii) a user-defined finite-time convergence (temporal constraint). Specifically, the presented algorithm effectively addresses these spatiotemporal constraints without the need for a strict knowledge of agent uncertainties upper bounds and without relying on agent initial conditions. The efficacy of the proposed control algorithm is further illustrated in a numerical example.

II. NOTATION AND NECESSARY PRELIMINARIES

In this paper $\mathbb{R}$, $\mathbb{R}^n$, and $\mathbb{R}^{n \times m}$ respectively stand for the set of real numbers, the set of $n \times 1$ real column vectors, and the set of $n \times m$ real matrices; $\mathbb{R}_+$ and $\mathbb{R}_+^{n \times n}$ (resp., $\mathbb{R}_+^{n \times m}$) stand for the set of positive real numbers and the set of $n \times n$ positive-definite (resp., nonnegative-definite) real matrices; $\mathbb{Z}_+$ (resp., $\mathbb{Z}_+$) stands for the set of positive (resp., nonnegative) integers; $0_n$ and $1_n$ respectively stand for the $n \times 1$ zero vector and the $n \times 1$ ones vector; and “$\triangleq$” stands for equality by definition. Furthermore, $(\cdot)^T$ is used for the transpose operator, $(\cdot)^{-1}$ is used for the inverse operator, $\det(\cdot)$ is used for the determinant operator, $\|\cdot\|_2$ is used for the Euclidean norm, and $\|\cdot\|_F$ is used for the weighted Euclidean norm (i.e., $\|x\|_A = \sqrt{x^T A x}$ for $x \in \mathbb{R}^n$ and $A \in \mathbb{R}_+^{n \times n}$). We further use $\lambda_{\min}(A)$ (resp., $\lambda_{\max}(A)$) for $1$To achieve this objective, the proposed algorithm utilizes a (set-theoretic) error-dependent learning rate that is obtained from a generalized restricted potential function based on the studies documented in [1–4].

$2$To achieve this objective, we utilize the time transformation method documented in [14–17], which connects an original, prescribed time interval $t \in (0, T)$ with its infinite-time, stretched version $s \in (0, \infty)$. 

Enforcing Spatiotemporal Constraints in Control of Multiagent Systems with Uncertainties

Ehsan Arabi and Tansel Yucelen
Next, we overview several notions from graph theory (see [18, 19] for more details). Specifically, an undirected graph denoted by $\mathcal{G}$ is defined by a set $\mathcal{V}$ of nodes and a set $\mathcal{E}$ of edges. When nodes $i$ and $j$ are neighbors, then $(i, j) \in \mathcal{E}$ and $i \neq j$ indicates the neighboring relation. The degree of the node $i$, denoted by $d_i$, is defined by the number of its neighbors. The degree matrix of a graph $\mathcal{G}$, $D(\mathcal{G}) \in \mathbb{R}^{N \times N}$, is then given by $D(\mathcal{G}) \equiv \text{diag}(d) = [d_1, \ldots, d_N]^T$. A path $i_0, i_1, \cdots, i_L$ of a graph $\mathcal{G}$ is a finite sequence of nodes such that $i_{k-1} \sim i_k$, $k = 1, \ldots, L$. The graph $\mathcal{G}$ is said to be connected when every pair of distinct nodes has a path. We use $\mathcal{A}(\mathcal{G}) \in \mathbb{R}^{N \times N}$ for the adjacency matrix of a graph $\mathcal{G}$, which is defined by $[\mathcal{A}(\mathcal{G})]_{ij} \equiv 1$ when $(i, j) \in \mathcal{E}$ and $[\mathcal{A}(\mathcal{G})]_{ij} \equiv 0$ otherwise. We also use $\mathcal{B}(\mathcal{G}) \in \mathbb{R}^{N \times M}$ for the (node-edge) incidence matrix of a graph $\mathcal{G}$, which is defined by $[\mathcal{B}(\mathcal{G})]_{ij} \equiv 1$ when node $i$ is the head of edge $j$, $[\mathcal{B}(\mathcal{G})]_{ij} \equiv -1$ when node $i$ is the tail of edge $j$, and $[\mathcal{B}(\mathcal{G})]_{ij} \equiv 0$ otherwise. Above, $M$ is the number of edges, $i$ is an index for the node set, and $j$ is an index for the edge set. Finally, the graph Laplacian matrix, $L(\mathcal{G}) \in \mathbb{R}^{N \times N}$, is defined by $L(\mathcal{G}) \equiv D(\mathcal{G}) - \mathcal{A}(\mathcal{G})$ or, equivalently, $L(\mathcal{G}) = \mathcal{B}(\mathcal{G})\mathcal{B}(\mathcal{G})^T$. In this paper, a given multiagent system is modeled as a connected, undirected graph $\mathcal{G}$ with nodes and edges representing agents and interagent communication links, respectively.

**Remark 1:** As in [14, 16, 17], we utilize a notion from Section 1.1.1.4 of [20]. In particular, consider that $\xi(t)$ is the solution to the dynamics satisfying

$$\dot{x}(t) = f(t, x(t)), \quad x(0) = x_0, \quad (1)$$

$f : \mathbb{R}_+ \times \mathbb{R}^n \to \mathbb{R}^n$. In addition, consider that $t = \theta(s)$ is a time transformation, where $\theta(s)$ is a continuously differentiable and strictly increasing function. Here, define also $\psi(s) = \xi(\theta^{-1}(0))$. We can then write

$$\psi(s) = \theta'(s)f(\theta(s), \psi(s)), \quad \psi(\theta^{-1}(0)) = x_0, \quad (2)$$

$$\psi'(s) \equiv d\psi(s)/ds, \quad \theta'(s) \equiv d\theta(s)/ds.$$

**Remark 2:** Considering a given signal $\eta(t)$ and the discussion in Remark 1, we utilize the notation $\eta_i(s)$ to represent the corresponding transformed signal; that is, $\eta_i(s) \equiv \eta(\theta(s))$.

**Lemma 1 ([Lemma 3.3, 21]):** Consider the matrix given by $K = \text{diag}(k)$, where $k = [k_1, k_2, \ldots, k_N]^T$, $k_i \in \mathbb{Z}_+, i = 1, \ldots, N$. Furthermore, consider that (at least) one entry of $k$ is not zero. For a connected, undirected graph $\mathcal{G}$, then $\mathcal{F}(\mathcal{G}) \equiv \mathcal{L}(\mathcal{G}) + K \in \mathbb{R}^{N \times N}$ holds.

**Lemma 2:** Consider the dynamics satisfying

$$\psi'(s) = A_r \psi(s) + \eta(s), \quad \psi(0) = \psi_0, \quad (3)$$

where $\psi(s) \in \mathbb{R}^n$ is the system state. Let $A_r \in \mathbb{R}^{n \times n}$ be a Hurwitz matrix. Then, $\psi(s)$ is bounded for any bounded input $\eta(s)$. If, in addition, $\lim_{s \to \infty} \eta(s) = 0$, then $\lim_{s \to \infty} \psi(s) = 0$.

**Definition 1:** Consider $\Omega = \{\theta \in \mathbb{R}^n : (\theta_{\text{min}} \leq \theta_i \leq \theta_{\text{max}})_{i=1,2,\ldots,N}\}$ to be a convex hypercube in $\mathbb{R}^n$. Here, $(\theta_{\text{min}}, \theta_{\text{max}})$ stand for the minimum and maximum bounds for the $i^{\text{th}}$ component of the $n$-dimensional parameter vector $\theta$. In addition, consider $\Omega_2 = \{\theta \in \mathbb{R}^n : (\theta_{\text{min}} + v < \theta_i < \theta_{\text{max}} - v)_{i=1,2,\ldots,N}\} \subset \Omega$ to be the second hypercube, where $v$ is a small positive constant. The projection operator $\text{Proj} : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}^n$ is then given by $\text{Proj}(\theta, y) \equiv \max(\theta_{\text{min}} + v, y)$ when $\theta_i > \theta_{\text{max}} - v$ and $\theta_i > 0$, $\text{Proj}(\theta, y) \equiv (\theta_{\text{min}} + v)\theta_i$ when $\theta_i < \theta_{\text{max}} + v$ and $\theta_i < 0$, and $\text{Proj}(\theta, y) \equiv y_i$ otherwise, where $y \in \mathbb{R}^n$. Furthermore, one can write $(\theta - \theta)^T(\text{Proj}(\theta, y) - y) \leq 0$ [23, 24].

**Definition 2:** We define a generalized restricted potential function (generalized barrier Lyapunov function) $\phi(\|z\|_H)$, $\phi : \mathbb{R} \to \mathbb{R}$ on the set $\mathcal{H} \equiv \{z : \|z\|_H \leq c, 0 \leq c < \infty\}$. For $z \in \mathbb{R}^n$, $H \in \mathbb{R}^{n \times n}$, and $c \in \mathbb{R}_+$ being a user-defined constant, when the following statements hold [1]: i) $\|z\|_H = 0$, then $\phi(\|z\|_H) = 0$. ii) $\|z\|_H = 0$, then $\phi(\|z\|_H) = 0$. iii) $\|z\|_H > 0$, then $\phi(\|z\|_H) = 0$. iv) $\phi(\|z\|_H)$ is continuously differentiable on $\mathcal{H}$, $\nabla \phi(\|z\|_H)/\|z\|_H^2$. $\phi$ is defined as $\|z\|_H = \|z\|_H$. $\phi(\|z\|_H)$ is continuously differentiable on $\mathcal{H}$, $\nabla \phi(\|z\|_H)/\|z\|_H^2$. $\phi$ is defined as $\|z\|_H$.

**III. Problem Formulation**

Consider a leader-follower scenario in a multiagent system that consists of $N$ agents exchanging information based on a connected, undirected graph $\mathcal{G}$. In addition, consider that a subset of the agents has access to the position of a time-varying leader satisfying

$$p(t) = \int_0^t v(t) \, dt + p(0), \quad p(t) \in \mathbb{R}, \quad (4)$$

with $v(t) \in \mathbb{R}$ being the bounded (with unknown bound) and piecewise continuous velocity of the leader. Furthermore, we consider the dynamics of each agent satisfies

$$\dot{x}_i(t) = u_i(t) + \omega_i x_i(t), \quad x_i(0) = x_0, \quad i \in \{1,2,\ldots,N\}. \quad (5)$$

In (5), $x_i(t) \in \mathbb{R}^n$, $i \in \{1,2,\ldots,N\}$ and $u_i(t) \in \mathbb{R}^n$, $i \in \{1,2,\ldots,N\}$ respectively represent the position and the control signal of the agent $i$, and $\omega_i \in \mathbb{R}$, $i \in \{1,2,\ldots,N\}$ represents system uncertainty. One can write (5) in the compact form

$$\dot{x}(t) = u(t) + \Omega x(t), \quad x(0) = x_0, \quad (6)$$

with $x(t) = [x_1(t), \ldots, x_N(t)]^T \in \mathbb{R}^n$ being the aggregated state vector and $\Omega \equiv \text{diag}(\omega_1, \ldots, \omega_N) \in \mathbb{R}^{n \times N}$. To discuss the performance guarantees, we first introduce the reference model dynamics capturing an ideal leader-follower behavior given by

$$f_l(t) = -\alpha \lambda(t) \left( \sum_{i=1}^{N} (r_i(t) - r_j(t)) + k_i(r_i(t) - p(t)) \right),$$

$r_i(0) = r_{0i}, \quad i \in \{1,2,\ldots,N\}, \quad (7)$

$^3$The result follows from Chapter 4.9 and Exercise 4.58 in [22].
follows that the reference model given by (7) results in a prescribed time interval $t$ where uncertain agents exchange information over a connected, undirected graph $G$.

with $r_i(t) \in \mathbb{R}$, $i \in \{1, 2, \ldots, N\}$ being the ideal reference model state. Utilizing $\lambda(t) = 1/(T - t)$ based on [14], it now follows that the reference model given by (7) results in a bounded trajectory that converges to the time-varying leader given by (4) at the user-defined finite time $T$; in other words,

$$\lim_{t \to T} (r_i(t) - p(t)) = 0, \quad i \in \{1, 2, \ldots, N\}. \tag{8}$$

Note that each state of the reference model in (7) needs to communicate with its neighbor on the graph $G$. In order to remove this dependence and motivated by the results in [25], we consider the reference model given by

$$\dot{x}_{i1}(t) = -\alpha \lambda(t) \left( \sum_{l \neq i} (x_{li}(t) - x_{lj}(t)) + k_i (x_{ii}(t) - p(t)) \right),$$

$$x_{i1}(0) = x_{i0}, \quad i \in \{1, 2, \ldots, N\}, \tag{9}$$

with $x_{i1}(t) \in \mathbb{R}$, $i \in \{1, 2, \ldots, N\}$, being the reference model state.

The objective of this paper is to present a distributed control algorithm to drive the position of each agent to that of the leader in a user-defined finite time $T \in \mathbb{R}_+$ (see Figure 1), while guaranteeing user-defined performance bounds $\epsilon_i$, $i \in \{1, 2, \ldots, N\}$, on the system error trajectories (i.e., the error between the agent's position $x_i(t)$ and an ideal system trajectory $x_{i1}(t)$); that is,

$$\lim_{t \to T} (x_i(t) - p(t)) = 0, \quad i \in \{1, 2, \ldots, N\},$$

$$|x_i(t) - x_{i1}(t)| < \epsilon_i, \quad i \in \{1, 2, \ldots, N\}. \tag{11}$$

IV. PROPOSED CONTROL ALGORITHM

In this section, we present the main result of this paper by proposing a distributed control algorithm for achieving both user-defined performance guarantees and a user-defined finite-time convergence to the position of the time-varying leader. To this end, consider the distributed control algorithm given by

$$u_i(t) = -\alpha \lambda(t) \left( \sum_{l \neq i} (x_{li}(t) - x_{lj}(t)) + k_i (x_{ii}(t) - p(t)) \right)$$

$$-\dot{\omega}_i(t)x_{i1}(t), \quad i \in \{1, 2, \ldots, N\}, \tag{12}$$

with $\dot{\omega}_i(t)$ being an estimation of the system uncertainty $\omega_i$, $i \in \{1, 2, \ldots, N\}$. In (12), $k_i = 1$ for the subset of the agents having access to the position of a time-varying leader in (4) and $k_i = 0$ for other agents. In addition, consider that the update law for $\dot{\omega}_i(t)$, $i \in \{1, 2, \ldots, N\}$, in (12) satisfying

$$\dot{\omega}_i(t) = y_i \text{Proj}(\dot{\omega}_i(t), \phi_{d_i}(|e_i(t)|x_i(t)e_i(t))), \quad \dot{\omega}_i(0) = \dot{\omega}_{i0}, \quad i \in \{1, 2, \ldots, N\}, \tag{13}$$

with $e_i(t) \triangleq x_i(t) - x_{i1}(t)$, $i \in \{1, 2, \ldots, N\}$, being the error between the position of an agent and its corresponding reference model state from (9) and $\dot{\omega}_{\max}$ being the projection bound. One can now express the control signal (12) in the compact form as

$$u(t) = -\alpha \lambda(t) \mathcal{F}(0) \tilde{x}(t) - \hat{\Omega}(t)x(t). \tag{14}$$

where $\tilde{x}(t) \triangleq x(t) - 1_N p(t) \in \mathbb{R}^N$ denotes the aggregated error vector between the position of each agent and that of the leader with the dynamics

$$\dot{\tilde{x}}(t) = \tilde{x}(t) - 1_N v(t),$$

$$= u(t) + \hat{\Omega}(t)x(t) - 1_N v(t), \quad \tilde{x}(0) = \tilde{x}_0, \tag{15}$$

and $\hat{\Omega}(t) \triangleq \text{diag}((\dot{\omega}_1, \ldots, \dot{\omega}_N)) \in \mathbb{R}^{N \times N}$.

Rewriting the reference model in (9) yields

$$\dot{x}_{i1}(t) = -\alpha \lambda(t) \left( \sum_{l \neq i} (x_{li}(t) - x_{lj}(t)) + k_i (x_{ii}(t) - p(t)) \right)$$

$$+ k_i (x_{ii}(t) - p(t)), \quad i \in \{1, 2, \ldots, N\},$$

$$= -\alpha \lambda(t) \left( \sum_{l \neq i} (x_{li}(t) - x_{lj}(t)) + k_i (x_{ii}(t) - p(t)) \right)$$

$$+ \alpha \lambda(t) \sum_{j \neq i} e_j(t), \quad x_{i1}(0) = x_{i0}, \quad i \in \{1, 2, \ldots, N\}. \tag{16}$$

We now define $\tilde{x}_{i1}(t) \triangleq x_{i1}(t) - p(t)$, $i \in \{1, 2, \ldots, N\}$, to capture the error between the reference trajectory and the position of the time-varying leader. The error dynamics for $\tilde{x}_{i1}(t)$ can now be written as

$$\dot{\tilde{x}}_{i1}(t) = -\alpha \lambda(t) \left( \sum_{l \neq i} (\tilde{x}_{li}(t) - \tilde{x}_{lj}(t)) + k_i \tilde{x}_{i1}(t) \right) - v(t)$$

$$+ \alpha \lambda(t) \sum_{j \neq i} e_j(t), \quad \tilde{x}_{i1}(0) = \tilde{x}_{i0}, \quad i \in \{1, 2, \ldots, N\}. \tag{17}$$

One can express the error dynamics (17) in the compact form as

$$\dot{\tilde{x}}(t) = -\alpha \lambda(t) \mathcal{F}(0) \tilde{x}(t) + \alpha \lambda(t) \mathcal{A}_d(0) e(t) - v(t) 1_N,$$

$$\tilde{x}(0) = \tilde{x}_0, \tag{18}$$

where $\tilde{x}(t) \triangleq [\tilde{x}_{11}(t), \ldots, \tilde{x}_{NN}(t)]^T \in \mathbb{R}^N$ and $e(t) \triangleq [e_1(t), \ldots, e_N(t)]^T \in \mathbb{R}^N$ denote the aggregated error vectors. Now, let the time transformation function be given by

$$t = \theta(s) \triangleq T(1 - e^{-s}). \tag{19}$$

This time transformation links a prescribed finite-time interval of interest $t \in [0, T]$ to the stretched infinite-time interval $s \in [0, \infty)$ and vice versa. Based on (19), let $\zeta(t) \in \mathbb{R}^N$, $t \in [0, T]$, be a solution to the dynamical system given by (18) and define $\tilde{x}_{il}(s) = \tilde{x}_{il}(t), s \in [0, \infty)$. It follows
from Remark 1 that

\[ \ddot{x}_i(s) = -\alpha \mathcal{F}(\theta_2) \dot{x}_i(s) + M_i(s) x_i(s) = x_{i0}. \]  

(20)

where \( M_i(s) = -\alpha \mathcal{F}(\theta_2) \dot{x}_i(s) - T e^{-\gamma t} \dot{\omega}_i(s) I_N \in \mathbb{R}_+ \).

Using (5), (9), and (12), we next write the error dynamics for \( e_i(t) \) as

\[ \dot{e}_i(t) = \dot{x}_i(t) - \dot{\dot{x}}_i(t), \]

\[ = -\alpha \lambda(t) \sum_{l=0}^m (x_l(t) - x_i(t)) + k_i e_i(t) - \dot{\omega}_i(t)x_i(t), \]

\[ = -\alpha \lambda(t) (d_i + k_i) e_i(t) - \dot{\omega}_i(t)x_i(t), \quad e_i(0) = e_{i0}, \]

where \( \dot{\omega}_i(t) = \dot{\omega}_i(t) - \omega_i, \quad i \in \{1, 2, \ldots, N\} \) and \( e_{i0} = x_{i0} - x_{i0}, \quad i \in \{1, 2, \ldots, N\} \). Similar to how (20) is derived from (18) using the time transformation function given by (19) according to Remark 1, one can also rewrite (21) as

\[ e'_i(s) = -\alpha (d_i + k_i) e_i(s) - T e^{-\gamma \dot{\omega}_i(s)} x_i(s), \quad e_i(0) = e_{i0}, \]

\[ i \in \{1, 2, \ldots, N\}, \]  

(22)

where the subscript \( s \) is used; see Remark 2. Using (13), the weight estimation error dynamics can be also expressed in the infinite-time interval as

\[ \ddot{\omega}_i(s) = \gamma_1 T e^{-\gamma} \text{Proj}(\dot{\omega}_i(s), \Phi_{d_i}(|e_i(s)|) x_i(s) e_i(s)), \]

\[ \dot{\omega}_i(0) = \dot{\omega}_{i0} - \omega_i, \quad i \in \{1, 2, \ldots, N\}. \]  

(23)

The following theorem presents the main result of this paper.

**Theorem 1:** Consider a multiagent system that consists of \( N \) agents on a connected, undirected graph \( G \), where the dynamics of the agent \( i \in \{1, \ldots, N\} \) satisfy the uncertain dynamics in (5). Furthermore, consider that there exists at least one agent sensing the position of the time-varying leader given by (4), which has bounded (but unknown) velocity. Let the design parameter \( \alpha \) be chosen such that \( \mathcal{F} = \alpha \mathcal{F}(\theta_2) - I_N \) is positive-definite and the distributed control algorithm \( u(t) \) be given by (12) along with its update law in (13). If \( |e_i(0)| < \epsilon_i, \quad i \in \{1, 2, \ldots, N\} \), then the closed-loop system signals (including all the distributed control signals) remain bounded and all agents converge to the position of the leader at the user-defined finite time \( T \) (i.e., \( \lim_{t \to T} x_i(t) = 0, \quad i \in \{1, 2, \ldots, N\} \)) for all initial conditions of agents, while guaranteeing user-defined performance bounds \( |e_i(t)| < \epsilon_i, \quad i \in \{1, 2, \ldots, N\} \).

Due to page limitations, the proof of the above theorem will be reported elsewhere. Here, we only provide a sketch of the proof. Specifically, consider the energy function \( V_i : \mathcal{D}_\epsilon \times \mathbb{R} \to \mathbb{R}_+ \) given by

\[ V_i(e_i, \dot{\omega}_i) = \phi_i(|e_i(s)|) + \gamma_i^{-1} \dot{\omega}_i^2, \quad i \in \{1, 2, \ldots, N\}, \]  

(24)

where \( \mathcal{D}_\epsilon \triangleq \{|e_i(s)| : |e_i(s)| < \epsilon_i\}, \quad i \in \{1, 2, \ldots, N\} \). One can write the derivative of (24) with respect to \( s \) along the closed-loop system trajectories of (22) and (23) as

\[ V'_i(e_i(s), \dot{\omega}_i(s)) \leq -\frac{1}{2} \beta_i V_i(e_i(s), \dot{\omega}_i(s)) + \mu, \]  

(25)

Fig. 2. A multiagent system consisting of four agents on an undirected, connected circle graph \( G \).

with \( \beta_i \triangleq 2 \alpha (d_i + k_i) \in \mathbb{R}_+, \quad \mu \triangleq \frac{1}{2} \beta_i \gamma_i^{-1} (\omega_{\text{max}}^2 + \omega_{\text{max}}^2), \) and \( \omega_{\text{max}} \triangleq \max(\omega_1, \ldots, \omega_N) \). It then follows that the control signal \( u(t) \) remains bounded for all \( t \in [0, T), \quad |e_i(t)| < \epsilon_i, \quad t \in [0, T), \quad i \in \{1, 2, \ldots, N\} \) and \( \lim_{t \to T} e(t) = \lim_{t \to T} x(t) - 1_N p(t) = 0 \).

**V. ILLUSTRATIVE NUMERICAL EXAMPLE**

In this section, we present a numerical example to illustrate the efficacy of the proposed distributed control algorithm. Consider a multiagent system consisting of four agents on an undirected, connected graph \( G \) as depicted in Figure 2. In this leader-follower scenario, the agent 2 has access to the position of a time-varying leader given by \( p(t) = -1.5 - 0.5 \sin(0.15 t) + 0.5 \cos(0.3 t) \). We consider the system uncertainties in (5) be \( \omega_1 = -1, \omega_2 = -1, \omega_3 = 1, \omega_4 = 1, \) and set the initial positions of the agents randomly over the interval \([-0.5, 0.5]\).

Considering a desired finite-time convergence of \( T = 10 \) seconds, one can utilize the results reported in [14] (i.e., (12) without the term \( \ddot{\omega}_i(t) x_i(t) \) with \( \alpha = 7 \) for addressing this temporal constraint problem. However, a (transient) performance guarantee on the system error vector cannot be enforced in such framework. This is evident from Figures 3 and 4, where the dashed line shows the position of the leader, solid lines show the position of agents, and dotted lines show the ideal reference trajectories. Specifically, although the agents converge to the position of the leader in \( T = 10 \) seconds, they are deviating from their ideal reference trajectories during the transient time. Figure 5 presents the evolution of the error signal \( e_i(t) \). It is clear from this figure that if a spacial performance constraint of \( \epsilon_i = 0.15, \quad i \in \{1, 2, \ldots, N\} \), is required such that the agents stay close to the position of their corresponding reference model position, the presented method in [14] is not able to satisfy such requirement.

We now implement the proposed distributed control algorithm in this paper (i.e., (12) along with (13)). We utilize the time transformation function given in (19) with \( T = 10 \) in order to enforce the finite-time convergence of 10 seconds and, once again, set \( \alpha = 7 \) that results in a positive-definite matrix \( \mathcal{F} \). Furthermore, we let the projection bound imposed on each element of the parameter
Fig. 3. Leader-follower performance with the finite-time control algorithm in [14] \((T = 10 \text{ and } \alpha = 7)\).

Fig. 4. Control signals of agents with the finite-time control algorithm in [14] \((T = 10 \text{ and } \alpha = 7)\).

Fig. 5. The evolution of \(e_i(t)\) with the finite-time control algorithm in [14] \((T = 10 \text{ and } \alpha = 7)\).

Fig. 6. Leader-follower performance with the proposed finite-time control algorithm proposed in Theorem 1 \((T = 10 \text{ and } \alpha = 7)\).

Fig. 7. Control signals of agents with the proposed finite-time control algorithm in Theorem 1 \((T = 10 \text{ and } \alpha = 7)\).

Fig. 8. The evolution of \(e_i(t)\) with the proposed finite-time control algorithm in Theorem 1 \((T = 10 \text{ and } \alpha = 7)\).
estimate to be $\dot{W}_{\text{max}} = 2$ and the adaptation rate to be $\gamma_i = 1$, $i \in \{1,2,\ldots,N\}$. For the generalized restricted potential function in Definition 2, we use $\phi_i(|e_i(t)|) = |e_i(t)|^2/(e - |e_i(t)|)$, $e_i(t) \in \mathbb{R}$ with $e_i = e = 0.15$, $i \in \{1,2,\ldots,N\}$ such that $|x_i(t) - x_{ti}(t)| < 0.15$, $i \in \{1,2,\ldots,N\}$ is guaranteed.

Figures 6 and 7 present the performance of the proposed distributed control algorithm for addressing spatial and temporal constraints simultaneously, where it is evident from these figures that the finite-time convergence is obtained at the user-defined finite time of $T = 10$ seconds. Furthermore, Figure 8 shows that the position of each agent is kept within a close neighborhood of the position of their corresponding reference trajectories by guaranteeing user-defined performance bounds $|x_i(t) - x_{ti}(t)| < 0.15$, $i \in \{1,2,\ldots,N\}$. The effective learning rate $\gamma_i \phi_i(|e_i(t)|)$, $i \in \{1,2,\ldots,N\}$ is also depicted in Figures 9. Finally, comparing Figure 7 with Figure 4, we note that the proposed distributed control algorithm here results in less control effort in accomplishing the considered task compared to the presented method in [14].

VI. CONCLUSION

In this paper, we presented a new distributed control algorithm for uncertain multiagent systems for achieving both user-defined system performance guarantee (spatial constraint) and a user-defined finite-time convergence (temporal constraint). In particular, by utilizing a distributed adaptive control law, the proposed algorithm was capable of limiting the worst-case difference between the agent state trajectories and their corresponding reference state trajectories. Furthermore, by utilizing a time transformation approach, the proposed algorithm was capable of achieving a user-defined finite-time convergence. Importantly, these spatiotemporal constraints were addressed without the need for a strict knowledge of agent uncertainties upper bounds and without relying on agent initial conditions.

REFERENCES