Consensus tracking and containment in multiagent networks with state constraints

Yilun Shang

***Abstract*—The ability of tracking is an important prerequisite for multiagent networks to perform collective activities. This paper investigates the problem of containment for a weighted multiagent network with continuous-time agents under state constraints. The network is composed of uninformed and informed agents, where the latter receive external inputs. A new general class of distributed nonlinear controllers is designed for accomplishing both containment and consensus tracking, where the state of each agent is required to stay in its desired convex constraint set. We show that, by using matrix analysis, convex analysis and Lyapunov theory, all agents eventually converge to the convex hull formed by the external inputs while they obey their constraints during the transience. No relationship is assumed between the convex hull and the intersection of all constraint sets. The consensus tracking problem with a single external input is also solved under this framework. As a generalization, we tackle the multi-scaled constrained containment problem, where agents can specify their desired buffer zones by either zooming in or zooming out the convex hull. Numerical examples are provided to illustrate the theoretical results.**

***Index Terms*—consensus problem; tracking; containment control; state constraint; networks.**

I. Introduction

In recent years, distributed cooperative control has attracted a considerable attention in theoretical and applicable fields involving multiagent networks, where a group of agents interact with their local neighbors following some simple rules [1]. In a system of interacting agents, consensus corresponds to the convergence to a common value among all agents [2], [3], which appears to be a critical prerequisite across disciplines like distribution estimation in sensor networks, networked control systems, and robot formation control. Consensus problems can be classed as leaderless consensus and leader-follower consensus (or tracking), where the latter problem consists of one leader agent and all other agents are expected to follow the leader by updating their states based on the information from their nearest neighbors [4], [5]. In the presence of multiple leaders, the tracking problem is referred to as containment control, where followers are to be driven to the convex hull spanned by these leaders. Strategies for designing the control protocol so that containment can be realized over a variety of topologies and system dynamics have been intensively studied [6]–[11].

It is widely known that the cooperative agents can not deviate arbitrarily from their initial states in realistic systems. Examples include speed limitation of vehicles, position constraint due to safety requirement, and opinion bound influenced by social norms. A discrete-time constrained consensus algorithm is proposed in [12] over balanced switching networks. A generalization to the unbalanced topology and systems with communication delay is considered in [13]. For continuous-time consensus problems, various design strategies such as barrier functions [14], [15], projection [16]–[18], and saturation functions [19]–[21] have been used to achieve constrained consensus for both leaderless and tracking situations. However, constrained consensus protocols are not applicable to the containment control problems due to the complexity of set convergence and set tracking involving multiple leaders. The constrained containment problem is motivated by many real applications. For example, a group of robots moving from one location to another with a small set of them equipped with a map of the destination. To reach the destination safely, the maximum velocity has constraints when the robots are trains on track, the position has constraints if they are cars in road, and the altitude has constraints if they are satellites in orbit. Employing a low-gain algebraic Riccati equation method, distributed containment control with input saturation has been studied in [22] for single-integrator multiagent systems. The work [23] addresses the containment control problem for double-integrators with both velocity and acceleration saturations. An anti-windup approach is incorporated to design linear controllers in [24] to facilitate containment under actuator saturation. Based on model transformation techniques, containment control with non-convex velocity constraints has also been recently dealt with for second-order discrete-time [25] and continuous-time [26] systems. An application to the control of high-speed trains is also presented in [27].

All the aforementioned studies in constrained containment control have been focused on velocity constraints and input saturation. Much less has been done regarding the highly relevant question of position or state constraints, where the position of each agent is constrained in a convex set. As discussed in [28], the analysis approach for constrained consensus or containment with input saturations can not be applied in the case of position constraint. In [28], a projection-based distributed control protocol is proposed to accomplish containment for discrete-time agents. A strong assumption made is that the intersection of the convex constraint sets for all agents contains the convex hull formed by all leaders as a subgraph. To the best of our knowledge, containment problem with state constraints has yet to be studied in continuous-time multi-agent systems.

Motivated by the above considerations, we in this paper

Y. Shang is with the Department of Computer and Information Sciences, Northumbria University, Newcastle NE1 8ST, UK (e-mail: yilun.shang@northumbria.ac.uk).

Manuscript received September 4, 2021; revised xxxx xx, xxxx.

investigate distributed containment for continuous-time agents under state constraints, where each agent stays in a closed convex set only known by itself. Employing matrix analysis, convex analysis and the Lyapunov theory, we show that the state of all agents will be driven to the convex hull formed by the leaders when the underlying communication network is connected. The approach is totally different from [28] and no assumption on the relationship between the intersection of all convex constraint sets and the convex hull is made. A non-empty intersection of each individual constraint set and the convex hull turns out to be both sufficient and necessary for reaching constrained containment (c.f. Remark 6). In the case of only one leader, we readily solves a constrained consensus tracking problem as a further contribution. Moreover, the framework is also extended to solve multi-scaled constrained containment, where each agent is able to converge to a specified scaled region. This is potentially useful in applications in that each agent is capable of specifying (i) its individual buffer/safety zone, which may be a zoom-out area of the convex hull or (ii) its individual core zone, which may be a zoom-in area of the convex hull.

We mention that the containment control of multiagent systems with input saturations is often tackled by establishing a closed-loop error dynamics at the low level of system dynamics and solved by noting different system specification and some coordinate rotation-free properties of the control laws [22]–[26]. In the case of switching networks, for instance, an input saturated closed-loop system can be transformed into an equivalent time-varying system under strict assumptions of switching patterns and dwell-times [23], [24]. In the state constrained problem however, this is often difficult even in the case of fixed networks due to the general format of constrained sets. A more geometric approach is a natural choice. Taking this into account, we will rely on a class of distributed gradient-based controllers (Section III) to drive all agents to the convex hull while remain in their respective constraint sets.

The rest of the paper is organized as follows. Section II provides graph theory notions and formulates the model. Section III focuses on the containment analysis of the multiagent system with state constraints. Section IV presents several numerical simulation examples and finally the conclusion is drawn in Section V.

II. Preliminaries

A. Graph theory and notations

For a positive integer n, define the set $[n] = \{1, 2, \cdots, n\}$. A multiagent network can be modeled as a weighted graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, A)$ with the node set $\mathcal{V} = [n]$ and the edge set $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$. The interconnection between the agents or nodes is characterized by the adjacency matrix $A = (a_{ij}) \in \mathbb{R}^{n\times n}$, where $a_{ij} > 0$ if $(i, j) \in \mathcal{E}$ and $a_{ij} = 0$ otherwise. We assume the graph is undirected and hence $a_{ij} = a_{ji}$. For each node $i \in \mathcal{V}$, let $\mathcal{N}_i = \{j \in \mathcal{V} : (i, j) \in \mathcal{E}\}$ be the set of neighbors of i. A path between two nodes i and j is a sequence of consecutive edges connecting i to j by tracing along the neighborhoods. A graph is called connected if there is a path linking any two different nodes. The graph Laplacian matrix of G is defined as $L = (l_{ij}) \in \mathbb{R}^{n\times n}$, where $l_{ij} = \sum_{j=1}^{n} a_{ij}$ for $i = j$ and $l_{ij} = -a_{ij}$ for $i \neq j$.

A matrix is called nonnegative or positive if all its elements are nonnegative or positive, respectively. Clearly, the adjacency matrix A is nonnegative and $A^{\mathsf{T}} = A$ for an undirected graph, where T means the matrix transpose. A matrix $P \in \mathbb{R}^{n\times n}$ is called an M-matrix if there is $s > 0$ satisfying $P = sI_n - S$, where $I_n \in \mathbb{R}^{n\times n}$ is the identity matrix and $S \in \mathbb{R}^{n\times n}$ is nonnegative such that the spectral radius (i.e., maximum-modulus eigenvalue) of S is not larger than s [29].

B. Problem formulation

Consider the multiagent network $\mathcal{G}$ of n agents with single-integrator dynamics given by

$$\dot{x}_i(t) = \xi_i(t), \qquad i \in [n], \qquad t \geq 0, \tag{1}$$

where $d \geq 1$, $x_i(t) \in \mathbb{R}^d$ and $\xi_i(t) \in \mathbb{R}^d$ are the state and the control input of agent i, respectively. A node in $\mathcal{V}$ can be either uninformed or informed, where an informed node will receive external control signal to steer the network towards a desired state. Let $\mathcal{U} = \{u_1, u_2, \cdots, u_m\}$ be the set of external inputs, where $u_l \in \mathbb{R}^d$ is a constant vector for $l \in [m]$. We will design the control input ξ_i in the system (1) in Section III, which is a function of the following

$$y_i(t) = -\sum_{j\in\mathcal{N}_i} a_{ij}(x_i(t) - x_j(t)) - \sum_{l=1}^{m} b_{il}(x_i(t) - u_l), \qquad i \in [n], \qquad t \geq 0. \tag{2}$$

Here, if the agent i is informed, then $b_{il} > 0$ for some l; if the agent i is uninformed, then $b_{il} = 0$ for all $l \in [m]$. Let $\mathrm{co}(\mathcal{U}) = \{\sum_{l=1}^{m} \alpha_l u_l : \sum_{l=1}^{m} \alpha_l = 1, \alpha_l \geq 0\}$ be the convex hull formed by the external inputs.

Remark 1. Taking $\xi_i = y_i$ in (1) gives us a general framework of containment control without constraints. By using the coefficients $\{b_{il}\}_{l\in[m]}$, the equation (2) flexibly combines the traditional separate representation of leader and follower agents [4], [9], where a typical leader follows $\dot{x}_l(t) = \xi_l(t)$ $(l \in [m])$ and a typical follower behaves like $\dot{x}_i(t) = -\sum_{j\in\mathcal{N}_i} a_{ij}(x_i(t) - x_j(t)) - (x_i(t) - x_l(t))$ $(i \in \mathcal{V})$. In traditional tracking consensus protocols, leaders are only influenced by external signals. Technically, we call these nodes with external inputs as 'informed' nodes instead of 'leaders' because they can be influenced by other nodes according to (2). The external inputs $\{u_l\}_{l\in[m]}$ themselves can be viewed as static leaders, namely, $x_l(t) = u_l$ for $l \in [m]$. Moreover, an informed node is allowed to be associated with multiple external inputs in the current framework, which can be conveniently implemented by adjusting the coefficients $\{b_{il}\}_{l\in[m]}$.

Suppose that each node $i \in \mathcal{V}$ has a convex state constraint set $X_i \in \mathbb{R}^d$ and the intersection of all sets is given by $X = \cap_{i=1}^{n} X_i$.

Assumption 1. The constraint set X_i is expressed as $X_i = \{x_i \in \mathbb{R}^d : p_i(x_i) \leq 1\}$ with its boundary $\partial X_i = \{x_i \in \mathbb{R}^d : p_i(x_i) = 1\}$. Assume $p_i : \mathbb{R}^d \to \mathbb{R}$ is a twice differentiable convex function, $X \neq \emptyset$, and $X_i \backslash \partial X_i \neq \emptyset$.

Remark 2. The assumption is mild. A general convex function p_i can be approximated by a twice differentiable convex function in the sense of Hausdorff metric [30]. For example, a cone variant represented by $p_i(x_i) \leq 1$ will be needed to approximate a pyramid in $\mathbb{R}^3$. In fact, approximating polygons by using conic sections in a Hausdorff space has been intensively studied in computational geometry. Some iterative numerical algorithms have been proposed; see e.g. [31], [32]. Also note that the constraint set X_i is non-trivial since its interior is not empty.

Assumption 2. For any $i \in [n]$, if $b_{il} > 0$ then $u_l \in X_i$. Moreover, $X_i \cap \mathrm{co}(\mathcal{U}) \neq \emptyset$ for $i \in [n]$.

Remark 3. The first part of the assumption says that an agent can only be informed by the external signal inside its constraint set. This is natural in many applications since the constraint set is often viewed as the detectable or admissible variable range [33]. We also assume that each individual constraint set intersects the convex hull, which is weaker than the assumption $X_i \cap \mathrm{co}(\mathcal{U}) = \mathrm{co}(\mathcal{U})$ proposed in [28]. See also Remark 6.

Assumption 3. $x_i(0) \in X_i$ for $i \in [n]$.

The objective of the paper is to design distributed constrained containment controllers for all agents to converge to the convex hull spanned by the external inputs. Namely, for any initial condition $x_i(0) \in X_i$, constrained containment is said to be achieved if $\lim_{t\to\infty} x_i(t) \in X_i \cap \mathrm{co}(\mathcal{U})$ for every $i \in \mathcal{V}$.

III. Main result

This section proposes a class of distributed gradient-based nonlinear controllers to solve the constrained containment and tracking problems.

Specifically, for each constraint set X_i expressed in Assumption 1, we define a function $q_i : \mathbb{R}^d \to \mathbb{R}$ by $q_i(x_i) = (p_i(x_i) - \epsilon_i)/(1 - \epsilon_i)$, where $\min_{x_i \in X_i} p_i(x_i) < \epsilon_i < 1$. The constraint set X_i can be divided into two parts: an internal part $X_i^{\mathrm{int}} = \{x_i \in \mathbb{R}^d : q_i(x_i) \in (-\infty, 0]\}$ and a peripheral part $X_i^{\mathrm{per}} = \{x_i \in \mathbb{R}^d : q_i(x_i) \in (0, 1]\}$. Clearly, $X_i = X_i^{\mathrm{int}} \cup X_i^{\mathrm{per}}$. For each node $i \in [n]$, the distributed controller of (1) is designed as

$$\xi_i(t) = \begin{cases} y_i(t), & \text{if } x_i(t) \in X_i^{\mathrm{int}}, \\ y_i(t), & \\ \quad \text{if } x_i(t) \in X_i^{\mathrm{per}}, \nabla q_i(x_i(t))^{\mathsf{T}} y_i(t) \leq 0, & \\ [I_d - q_i(x_i(t))Q_i(x_i(t))]\, y_i(t), & \\ \quad \text{if } x_i(t) \in X_i^{\mathrm{per}}, \nabla q_i(x_i(t))^{\mathsf{T}} y_i(t) > 0, & \end{cases} \tag{3}$$

where $y_i(t) \in \mathbb{R}^d$ is defined by (2) and $Q_i \in \mathbb{R}^{d\times d}$ is a positive semidefinite matrix defined by

$$Q_i(x_i) = \frac{\nabla q_i(x_i) \nabla q_i(x_i)^{\mathsf{T}}}{\|\nabla q_i(x_i)\|^2}, \quad \nabla q_i(x_i) \neq 0_d, \tag{4}$$

where $0_d \in \mathbb{R}^d$ is the vector of all zeros, $\nabla q_i(x_i) = \left(\frac{\partial q_i(x_i)}{\partial x_{i1}}, \frac{\partial q_i(x_i)}{\partial x_{i2}}, \cdots, \frac{\partial q_i(x_i)}{\partial x_{id}}\right)^{\mathsf{T}} \in \mathbb{R}^d$ and $x_i = (x_{i1}, x_{i2}, \cdots, x_{id})^{\mathsf{T}} \in \mathbb{R}^d$. Note that when $\nabla q_i(x_i) = 0_d$, the matrix $Q_i(x_i)$ is not required in light of (3).

Remark 4. Note that ϵ_i can be selected by each agent i individually, which regulates the relative sizes of X_i^{int} and X_i^{per}. For instance, a large ϵ_i i.e., ϵ_i close to 1, leads to a small peripheral part X_i^{per} and a large internal part X_i^{int}. This indicates the matrix Q_i will act at a later stage; see Example 2 below for an illustration. Clearly, the proposed strategy is purely distributed.

Remark 5. Note that the time derivative $\dot{q}_i(x_i(t)) = \nabla q_i(x_i(t))^{\mathsf{T}} \dot{x}_i(t)$. The control input for the multiagent system (1) is $y_i(t)$ when x_i is inside X_i^{int} or moves into X_i^{per} but pointing towards X_i^{int}. When x_i stays in X_i^{per} pointing outwards, the control input is modified as $[I_d - q_i(x_i(t))Q_i(x_i(t))]\, y_i(t)$. This amounts to a projection which pushes the state back into the constraint set; see Lemma 1 below. The gradient-based approaches have been widely applied in signal processing and adaptive control problems; see e.g. [16], [17], [25], [28], [34].

Theorem 1 gives the main result on the constrained containment and tracking of multiagent system (1).

Theorem 1. *Consider the system modeled by (1) with (2) and (3) over the multiagent network $\mathcal{G} = (\mathcal{V}, \mathcal{E}, A)$. With Assumptions 1, 2 and 3 satisfied, if $\mathcal{G}$ is connected, then the constrained containment is achieved, namely, $x_i(t) \to X_i \cap \mathrm{co}(\mathcal{U})$ for every $i \in \mathcal{V}$ as $t \to \infty$.*

Moreover, if $u_l \equiv \bar{u}$ for all $l \in [m]$, then the constrained consensus tracking is achieved, namely, $x_i(t) \to \bar{u}$ for every $i \in \mathcal{V}$ as $t \to \infty$.

The theorem is proved through a series of lemmas (see Fig. 1 for a diagram of the relationship of them). All technical details are given in the Appendix.

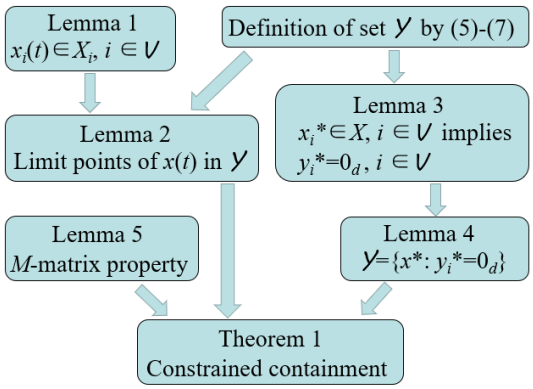

Fig. 1. Schematic illustration of proof structure.

Lemma 1. *Consider the system modeled by (1) with (2) and (3) over the multiagent network $\mathcal{G} = (\mathcal{V}, \mathcal{E}, A)$. With Assumptions 1 and 3 satisfied, $x_i(t) \in X_i$ for any $i \in \mathcal{V}$ and $t \geq 0$.*

Lemma 1 implies that the constraint sets $\{X_i\}_{i\in[n]}$ are invariant sets relevant to the state trajectories.

Let $x^* = (x_1^{*\mathsf{T}}, x_2^{*\mathsf{T}}, \cdots, x_n^{*\mathsf{T}})^{\mathsf{T}} \in \mathbb{R}^{nd}$. Define a set $\mathcal{Y} \in \mathbb{R}^{nd}$ as follows

$$\mathcal{Y} := \Big\{ x^* : \sum_{i\in\mathcal{Q}} y_i^{*\mathsf{T}} \left[I_d - q_i(x_i^*)Q_i(x_i^*)\right] y_i^* + \sum_{i\in\mathcal{V}\setminus\mathcal{Q}} y_i^{*\mathsf{T}} y_i^* = 0 \Big\}, \tag{5}$$

where

$$\mathcal{Q} := \{i \in \mathcal{V} : q_i(x_i^*) \in (0, 1], \nabla q_i(x_i^*)^{\mathsf{T}} y_i^* > 0\} \tag{6}$$

and

$$y_i^* := -\sum_{j\in\mathcal{N}_i} a_{ij}(x_i^* - x_j^*) - \sum_{l=1}^{m} b_{il}(x_i^* - u_l), \qquad i \in \mathcal{V}. \tag{7}$$

We will reserve the notation x^* for points in the set $\mathcal{Y}$ as determined above.

Lemma 2. *Let $x(t) = (x_1(t)^\mathsf{T}, x_2(t)^\mathsf{T}, \cdots, x_n(t)^\mathsf{T})^\mathsf{T} \in \mathbb{R}^{nd}$ be a solution of the system (1) with (2) and (3) over the multiagent network $\mathcal{G} = (\mathcal{V}, \mathcal{E}, A)$. With Assumptions 1 and 3 satisfied, the set of limit points of $x(t)$ is contained in $\mathcal{Y}$.*

Lemma 2 presents a geometric characterization of the set $\mathcal{Y}$, which contains all limit points of the solution of the multiagent system (1). A useful property of points in the set $\mathcal{Y}$ is established below.

Lemma 3. *Consider the system modeled by (1) with (2) and (3) over the multiagent network $\mathcal{G} = (\mathcal{V}, \mathcal{E}, A)$. With Assumptions 1, 2 and 3 satisfied, if $x_i^* \in X$ for every $i \in \mathcal{V}$, then $y_i^* = 0_d$ for every $i \in \mathcal{V}$, where $0_d \in \mathbb{R}^d$ is the vector of all elements being zero.*

With the aid of Lemma 3, we can characterize the invariant set $\mathcal{Y}$ as follows.

Lemma 4. *Consider the system modeled by (1) with (2) and (3) over the multiagent network $\mathcal{G} = (\mathcal{V}, \mathcal{E}, A)$. Under Assumptions 1, 2 and 3, if $\mathcal{G}$ is connected, then $\mathcal{Y} = \{x^* : \sum_{i=1}^n y_i^{*\mathsf{T}} y_i^* = 0\} = \{x^* : y_i^* = 0_d\}$, where y_i^* is determined by (7).*

To prove Theorem 1, the final trick up our sleeves is the following lemma regarding an equivalent condition of invertible M-matrices.

Lemma 5. ([36]) *Let $A \in \mathbb{R}^{n\times n}$ be a matrix with all off-diagonal elements nonpositive. A is an invertible M-matrix if and only if A^{-1} exists and is nonnegative.*

Remark 6. When $\mathcal{G}$ is connected, under some system assumptions (Assumptions 1, 2, and 3), $X_i \cap \mathrm{co}(\mathcal{U})$ for $i \in [n]$ is a sufficient condition for constrained containment. On the other hand, if constrained containment is achieved, it is obvious that $X_i \cap \mathrm{co}(\mathcal{U})$ must hold for every $i \in [n]$.

Remark 7. The gradient-based control scheme considered here is for undirected communication topology. The symmetry of the system Laplacian L is essential to construct the Lyapunov candidate $V(x)$, where the LaSalle invariance principle can be leveraged (see the proof of Lemma 2 in Appendix). To consider directed networks or bidirectional networks, a symmetrized version of Laplacian such as $L + L^\mathsf{T}$ should be proposed, which seems to be challenging in the current framework. The main difficulty in the system analysis comes from the vector analysis of the limit points in the invariant set $\mathcal{Y}$ concerning their distance to the common constraint set X (see the proof of Lemma 4 in Appendix). This differentiates the current work from the input saturation scenarios [22]–[26], where we literally have $X = \mathbb{R}^d$.

Remark 8. We have only considered the single integrator dynamics, where nonlinear constraints are designed in the control input to the agents' states. In some complicated real-world applications, higher-order dynamics may be desirable. In those scenarios, it may be possible to adopt the projector design similar to [28] and apply them on, for instance, velocity and position. However, new analysis will be needed to recover the properties parallel to Lemmas 3 and 4 for the invariant set $\mathcal{Y}$.

As an extension of our main result Theorem 1, we here consider a multi-scaled version of containment problems. To be specific, let $(\theta_1, \theta_2, \cdots, \theta_n) \in \mathbb{R}^n$ be a list of non-zero numbers. Given this list, for any initial condition $x_i(0) \in \theta_i^{-1}X_i$, we say that multi-scaled constrained containment is achieve if $\lim_{t\to\infty} x_i(t) \in \theta_i^{-1}(X_i \cap \mathrm{co}(\mathcal{U}))$ for any $i \in \mathcal{V}$. Clearly, if we choose $\theta_i \equiv 1$ for all $i \in \mathcal{V}$, we reduce to the original containment problem.

To achieve the multi-scaled constrained containment, we redesign $y_i(t)$ in (2) as

$$y_i'(t) = -\operatorname{sgn}(\theta_i)\sum_{j\in\mathcal{N}_i} a_{ij}(\theta_i x_i(t) - \theta_j x_j(t)) - \sum_{l=1}^{m} b_{il}(\theta_i x_i(t) - u_l), i \in [n], t \geq 0, \tag{8}$$

where sgn is the signum function returning 1 for a positive and -1 for a negative. The gradient-based controller $\xi_i(t)$ in (3) can be modified accordingly as

$$\xi_i'(t) = \begin{cases} y_i'(t), & \text{if } x_i(t) \in \theta_i^{-1}X_i^{\mathrm{int}}, \\ y_i'(t), & \\ \quad \text{if } x_i(t) \in \theta_i^{-1}X_i^{\mathrm{per}}, \theta_i\nabla q_i(\theta_i x_i(t))^\mathsf{T} y_i'(t) \leq 0, \\ [I_d - q_i(\theta_i x_i(t))Q_i(\theta_i x_i(t))]\, y_i'(t), & \\ \quad \text{if } x_i(t) \in \theta_i^{-1}X_i^{\mathrm{per}}, \theta_i\nabla q_i(\theta_i x_i(t))^\mathsf{T} y_i'(t) > 0, \end{cases} \tag{9}$$

where $y_i'(t)$ is given by (8) and the matrix Q_i is defined the same as in (4) (with the free variable replaced by $\theta_i x_i$ in (9)).

We update Assumption 3 as follows.

Assumption 4. $x_i(0) \in \theta_i^{-1}X_i$ for $i \in [n]$.

With these preparation, we present the following corollary regarding multi-scaled constrained containment.

Corollary 1. *Consider the system modeled by (1) with (8) and (9) over the multiagent network $\mathcal{G} = (\mathcal{V}, \mathcal{E}, A)$. With Assumptions 1, 2 and 4 satisfied, if $\mathcal{G}$ is connected, then the multi-scaled constrained containment is achieved, namely, $x_i(t) \to \theta_i^{-1}(X_i \cap \mathrm{co}(\mathcal{U}))$ for every $i \in \mathcal{V}$ as $t \to \infty$.*

Moreover, if $u_l \equiv \bar{u}$ for all $l \in [m]$, then the multi-scaled constrained consensus tracking is achieved, namely, $x_i(t) \to \theta_i^{-1}\bar{u}$ for every $i \in \mathcal{V}$ as $t \to \infty$.

IV. Numerical simulations

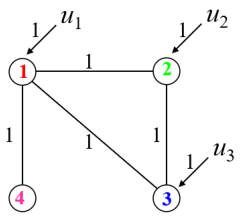

Fig. 2. Communication network $\mathcal{G}$ with $\mathcal{V} = [4]$ and $m = 3$ external inputs for Example 1.

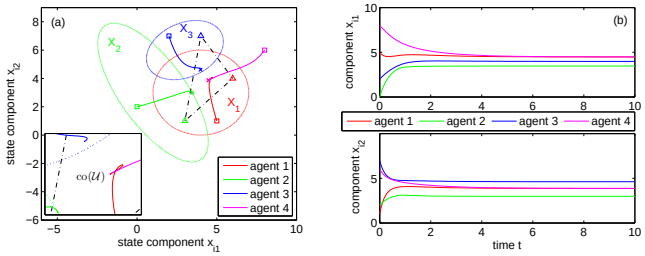

Fig. 3. Constrained containment for Example 1: (a) State trajectories $\{x_i\}_{i\in[4]}$. Initial and final positions are indicated by squares and crosses, respectively. External inputs are indicated by triangles. The triangle area formed by black dotted lines indicates $\mathrm{co}(\mathcal{U})$. The inset highlights the finial legs of trajectories. (b) Time evolution for the two state components.

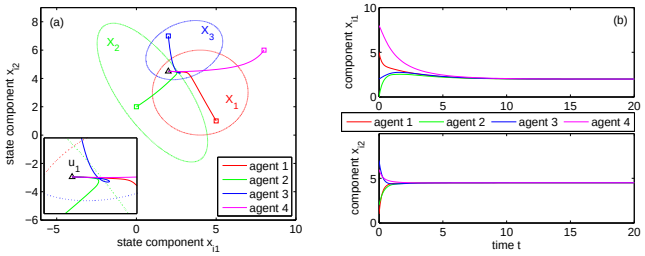

Fig. 5. Constrained consensus tracking for Example 2: (a) State trajectories $\{x_i\}_{i\in[4]}$. Initial positions are indicated by squares. The external input u_1 is indicated by a black triangle. The inset highlights the finial legs of trajectories. (b) Time evolution for the two state components.

In this section, we present some numerical examples to illustrate and discuss some different scenarios of the theoretical framework.

Example 1. (Constrained containment) We consider a connected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, A)$ with $\mathcal{V} = \{1, 2, 3, 4\}$ and $A \in \mathbb{R}^{4\times 4}$ being a binary $(0, 1)$-adjacency matrix; see Fig. 2. Let $m = 3$ and $d = 2$. The external inputs are $u_1 = (6, 4)^{\mathsf{T}}$, $u_2 = (3, 1)^{\mathsf{T}}$ and $u_3 = (4, 7)^{\mathsf{T}}$. The weights in $B \in \mathbb{R}^{4\times 3}$ is also binary.

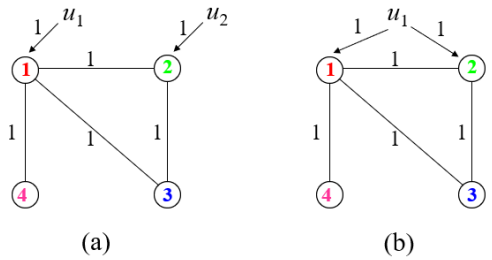

Fig. 4. Communication network $\mathcal{G}$ with $\mathcal{V} = [4]$ and $m^{(a)} = 2$ and $m^{(b)} = 1$ for Example 2.

Write $x = (x_1, \cdots, x_4)^{\mathsf{T}}$ and $x_i = (x_{i1}, x_{i2})^{\mathsf{T}}$ for $i \in [4]$. We define the following constraint sets $\{X_i\}_{i\in[4]}$, which are characterized by

$$\begin{aligned} p_1(x_1) =& (x_{11} - 4)^2 + (x_{12} - 3)^2 - 8, \\ p_2(x_2) =& 2(x_{21} - 1)^2 + (x_{22} - 3)^2 \\ &+ 2(x_{21} - 1)(x_{22} - 3) - 11, \\ p_3(x_3) =& 3(x_{31} - 3)^2 \\ &+ 4(x_{32} - 6)^2 - 2(x_{31} - 3)(x_{32} - 6) - 15, \\ p_4(x_4) =& -1. \end{aligned} \tag{10}$$

Clearly, the agents 1, 2, and 3 have closed convex constraints and the agent 4 is unconstrained, i.e., its constraint set is the entire plane. Regarding the parameters $\{\epsilon_i\}_{i\in[4]}$, we have the ranges $\epsilon_1 \in (-8, 1)$, $\epsilon_2 \in (-11, 1)$, $\epsilon_3 \in (-15, 1)$, $\epsilon_4 \in (-1, 1)$ according to the definition in the beginning of Section III. In this example, we set $\epsilon_1 = -1, \epsilon_2 = -2$, $\epsilon_3 = -3$, and $\epsilon_4 = 0$.

Fig. 3 shows the containment result for system (1) with (2) and (3) over $\mathcal{G}$ under the above state constraints. The initial conditions are taken as $x_1(0) = (5, 1)^{\mathsf{T}}$, $x_2(0) = (0, 2)^{\mathsf{T}}$, $x_3(0) = (1, 0)^{\mathsf{T}}$, and $x_4(0) = (-1, 5)^{\mathsf{T}}$. They are indicated by squares in Fig. 3(a). It is direct to check that all conditions of Theorem 1 are satisfied. The result shows that the constrained containment is achieved in line with the theoretical prediction. The states of agents 1 and 4 converge as one would expect from the communication topology in Fig. 2. Agent 3 arrives at an interior point within X_3 while agent 2 smoothly glides along the boundary ∂X_2. It is worth noting that in this example, $X \cap \mathrm{co}(\mathcal{U}) = \emptyset$. It helps to clarify that such an empty intersection does not impede the containment according to Theorem 1. The algorithm in [28], however, would not apply in this case.

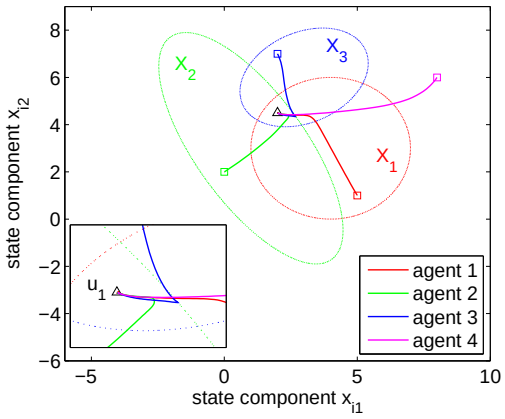

Fig. 6. Constrained consensus tracking for Example 2 with $\epsilon_1 = -7$, $\epsilon_2 = -10$, $\epsilon_3 = -14$ and $\epsilon_4 = 0$. All other parameters are the same as Fig. 5.

Example 2. (Constrained consensus tracking) We next consider a similar connected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, A)$ with $\mathcal{V} = \{1, 2, 3, 4\}$ as shown in Fig. 4. We again consider 2D space, namely $d = 2$. Regarding the external input, we display two scenarios, where $m^{(a)} = 2$,

$$B^{(a)} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}^{\mathsf{T}}, \tag{11}$$

$u^{(a)} = (u_1^{\mathsf{T}}, u_2^{\mathsf{T}})^{\mathsf{T}}$ with $u_1 = u_2 = (2, 4.5)^{\mathsf{T}}$ in Fig. 4(a) and $m^{(b)} = 1$, $B^{(b)} = (1, 1, 0, 0)^{\mathsf{T}}$ and $u^{(b)} = u_1 = (2, 4.5)^{\mathsf{T}}$ in Fig. 4(b), where the matrix B is defined in the proof of Lemma 2. It is straightforward to verify that $\mathrm{co}(\mathcal{U}^{(a)}) = \mathrm{co}(\mathcal{U}^{(b)}) =$

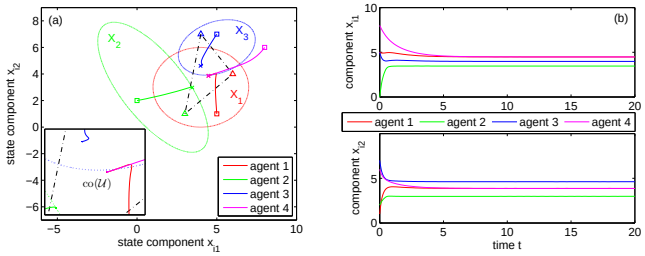

Fig. 7. Constrained containment for Example 3: (a) State trajectories $\{x_i\}_{i\in[4]}$. Initial and final positions are indicated by squares and crosses, respectively. External inputs are indicated by triangles. The triangle area formed by black dotted lines indicates $\mathrm{co}(\mathcal{U})$. The inset highlights the finial legs of trajectories. (b) Time evolution for the two state components.

$\{u_1\}$,

$$\mathrm{diag}(B^{(a)}1_{m^{(a)}}) = (1,1,0,0)^{\mathsf{T}} = \mathrm{diag}(B^{(b)}1_{m^{(b)}}), \quad (12)$$

and

$$(B^{(a)} \otimes I_2)u^{(a)} = (u_1^{\mathsf{T}}, u_1^{\mathsf{T}}, 0_2^{\mathsf{T}}, 0_2^{\mathsf{T}})^{\mathsf{T}} = (B^{(b)} \otimes I_2)u^{(b)}. \quad (13)$$

Therefore, the two scenarios are essentially the same as one would expected. Nevertheless, our general framework provides flexibility for different configurations of inputs. For example, multiple inputs are allowed for a single informed agent.

The constraint sets $\{X_i\}_{i\in[4]}$ and $\{\epsilon_i\}_{i\in[4]}$ are taken as Example 1. Fig. 5 shows the consensus tracking result for system (1) with (2) and (3) over $\mathcal{G}$ under the above state constraints and the same initial states as Example 1. The external input $u_1 \in X$ is represented as a triangle. We observe that all agents are driven to the fixed input state u_1, which is in consistent with Theorem 1.

Finally, we note that the parameters $\{\epsilon_i\}_{i\in[4]}$ chosen above and in Example 1 are relatively large given the possible ranges. As commented in Remark 4, this means the constraints are in effect at a later stage. As a comparison, we plot in Fig. 6 the state evolution for a set of smaller values: $\epsilon_1 = -7$, $\epsilon_2 = -10$, $\epsilon_3 = -14$ and $\epsilon_4 = 0$. This in principle would activate the constraints at an earlier stage. Comparing the two insets of Fig. 6 and Fig. 5(a), we notice a clear evidence is that agent 2 is farther repulsed from the boundary of X_2 in Fig. 6.

Example 3. (The case of $X = \emptyset$) The condition $X = \cap_{i\in[n]}X_i \neq \emptyset$ (in Assumption 1) plays a key role in our proof of Theorem 1. Obviously, if this condition does not hold, constrained consensus tracking would not be achieved. However, it is not clear whether this condition is necessary for constrained containment. Here, we give a preliminary example by modifying Example 1 to show that constrained containment is still possible when $X = \emptyset$. Analytical study will be a future work.

We make a minimal modification by considering the same graph $\mathcal{G}$ in Example with binary weights in both A and B. We only shift the constraint set X_3 for agent 3 so that the intersection $X = \cap_{i\in[4]}X_i$ becomes empty. Specifically, we redefine

$$\begin{aligned} p_3(x_3) =& 3(x_{31}-5)^2 + 4(x_{32}-6)^2 \\ & - 2(x_{31}-5)(x_{32}-6) - 15. \end{aligned} \quad (14)$$

With the initial condition of agent 3 changed to $x_3(0) = (5,7)^{\mathrm{T}} \in X_3$ and all other conditions remain the same, we show the evolution of agents in Fig. 7. It can be seen that the constrained containment is still achieved as in Example 1, where each agent converges to the intersection of its constraint set and the convex hull $\mathrm{co}(\mathcal{U})$.

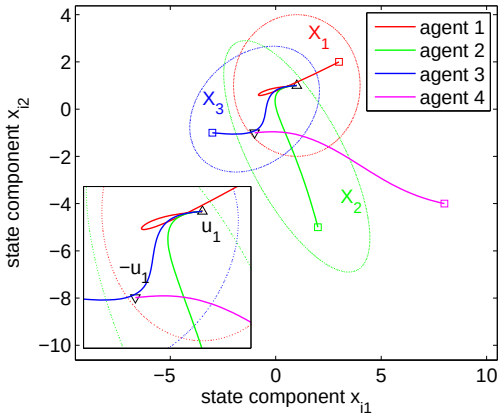

Fig. 8. Centrosymmetric constrained consensus tracking for Example 4. The external input $u_1 = (1,1)^{\mathsf{T}}$ is indicated by the upper triangle and the centrosymmetric point $-u_1$ is indicated by the lower triangle.

Example 4. (Centrosymmetric consensus tracking) In this final example, we consider a centrosymmetric consensus tracking over the connected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, A)$ described in Fig. 4(b). Let $d = 2$. For the external input, we set $m = 1$, $B = (1,1,0,0)^{\mathsf{T}}$ and $u = u_1 = (1,1)^{\mathsf{T}}$, where B is defined in the proof of Lemma 2. It is straightforward to verify that $\mathrm{co}(\mathcal{U}) = \{u_1\}$. The constraint sets $\{X_i\}_{i\in[4]}$ are taken as follows

$$\begin{aligned} p_1(x_1) =& (x_{11}-1)^2 + (x_{12}-1)^2 - 8, \\ p_2(x_2) =& 2(x_{21}-1)^2 + (x_{22}+2)^2 \\ & + 2(x_{21}-1)(x_{22}+2) - 11, \\ p_3(x_3) =& 3(x_{31}+1)^2 \\ & + 4x_{32}^2 - 2(x_{31}+1)x_{32} - 25, \\ p_4(x_4) =& -1. \end{aligned} \quad (15)$$

The parameters $\{\epsilon_i\}_{i\in[4]}$ are given by $\epsilon_1 = -1, \epsilon_2 = -2$, $\epsilon_3 = -3$, and $\epsilon_4 = 0$ so that they are inside the allowed ranges.

Suppose that we aim to drive the three agents 1, 2 and 3 to the external input point u_1 and drive agent 4 to the centrosymmetric point $-u_1 = (-1,-1)^{\mathsf{T}}$. To this end, we set $\theta_1 = \theta_2 = \theta_3 = 1$ and $\theta_4 = -1$ and apply the multi-scaled constrained containment strategy given in (8) and (9). It is straightforward to check that centrosymmetric consensus tracking can be achieved according to Corollary 1. The trajectories of the state components are shown in Fig. 8.

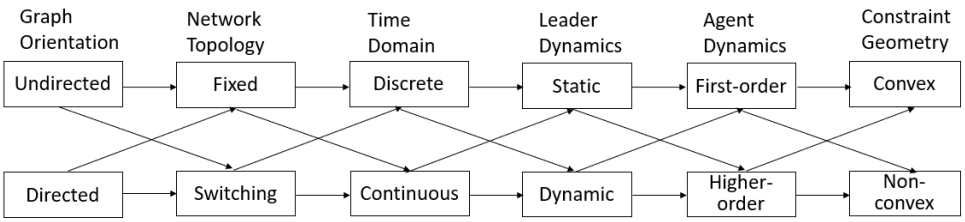

Fig. 9. Classification of containment control with state constraints.

V. Conclusion

Distributed containment control problem under state constraints has been studied in this paper. The agents follow continuous-time dynamics over a fixed undirected network. Informed agents in the network are able to receive information from external inputs. The newly developed nonlinear controllers drive all agents to the convex hull spanned by the external inputs, and at the same time, limit their states within the constraint sets. Each constraint set is only supposed to intersect the convex hull, which is the weakest possible assumption for this problem. The constrained consensus tracking problem and the multi-scaled containment problem are solved as a byproduct.

For future work, it would be interesting to explore the necessity of some conditions such as $X \neq \emptyset$ and the proposed method in this paper may be generalized to double integrator agents, which are favorable in areas such as robotics and formation control. In addition, as the containment control with state constraints is a nascent field, we present in Fig. 9 a diagram for a classification of some possible combinations of network and system conditions for future investigation.

Acknowledgment

The author is grateful to the four anonymous reviewers and the editor for their constructive comments that helped improve the paper significantly.

Appendix

Proof of Lemma 1

Fix any agent $i \in [n] = \mathcal{V}$. The initial state $x_i(0)$ belongs to the convex constraint set X_i by Assumptions 1 and 3. For the its state to move outside the constraint set X_i, agent i must first move into the peripheral part $X_i^{\rm per}$ and point outwards. It follows from (1)-(4) that

$$\begin{aligned}\dot{q_i}(x_i(t)) =&\nabla q_i(x_i(t))^{\mathsf{T}}\dot{x}_i(t)\\ =&\nabla q_i(x_i(t))^{\mathsf{T}}\left[I_d - q_i(x_i(t))Q_i(x_i(t))\right]y_i(t)\\ =&\nabla q_i(\alpha_i x_i(t))^{\mathsf{T}}y_i(t)\\ &- q_i(x_i(t))\nabla q_i(x_i(t))^{\mathsf{T}}\frac{\nabla q_i(x_i)\nabla q_i(x_i)^{\mathsf{T}}}{\|\nabla q_i(x_i)\|^2}y_i(t)\\ =&(1-q_i(x_i(t)))\nabla q_i(x_i(t))^{\mathsf{T}}y_i(t),\end{aligned} \tag{16}$$

where the term $1 - q_i(x_i(t))$ is positive but decreasingly diminishing as the state x_i approaches $\partial X_i = \{x_i \in \mathbb{R}^d : q_i(x_i) = 1\}$, and $\nabla q_i(x_i(t))^{\mathsf{T}}y_i(t) > 0$. Recall that X_i is closed. The state $x_i(t)$ will be pushed back into X_i instead of moving beyond X_i as it approaches ∂X_i.

Proof of Lemma 2

Let $B = (b_{il}) \in \mathbb{R}^{n\times m}$ and $L_B = L + \mathrm{diag}(B1_m)$, where $1_m \in \mathbb{R}^m$ is the vector with all elements being one. Let $u = (u_1^{\mathsf{T}}, u_2^{\mathsf{T}}, \cdots, u_m^{\mathsf{T}})^{\mathsf{T}} \in \mathbb{R}^{md}$. We consider the following Lyapunov candidate $V : \mathbb{R}^{nd} \to \mathbb{R}$:

$$V(x) = \frac{1}{2}x^{\mathsf{T}}(L_B \otimes I_d)x - u^{\mathsf{T}}(B^{\mathsf{T}} \otimes I_d)x, \tag{17}$$

which is a continuous and locally Lipschitz function. Here, $\otimes$ represents the Kronecker product [29].

In view of Lemma 1 and Assumptions 1 and 3, by (17) we derive

$$\begin{aligned}\dot{V}(x(t)) =&x(t)^{\mathsf{T}}(L_B \otimes I_d)\dot{x}(t) - u^{\mathsf{T}}(B^{\mathsf{T}} \otimes I_d)\dot{x}(t)\\ =&-\sum_{i=1}^{n} y_i(t)^{\mathsf{T}}\dot{x}_i(t)\\ =&-\sum_{i\in\mathcal{Q}(t)} y_i(t)^{\mathsf{T}}\left[I_d - q_i(x_i(t))Q_i(x_i(t))\right]y_i(t)\\ &-\sum_{i\in\mathcal{V}\setminus\mathcal{Q}(t)} y_i(t)^{\mathsf{T}}y_i(t),\end{aligned} \tag{18}$$

where the subset of nodes $\mathcal{Q}(t)$ is defined as $\mathcal{Q}(t) := \{i \in \mathcal{V} : q_i(x_i(t)) \in (0,1], \nabla q_i(x_i(t))^{\mathsf{T}}y_i(t) > 0\}$.

The idea is to show $\dot{V}(x(t)) \leq 0$ and invoke LaSalle's invariance principle [35]. For the second term of (18), $\sum_{i\in\mathcal{V}\setminus\mathcal{Q}(t)} y_i(t)^{\mathsf{T}}y_i(t) \geq 0$. We only need to examine the first term. We have for all $t \geq 0$,

$$\begin{aligned}I_d - q_i(x_i(t))Q_i(x_i(t)) =&I_d - Q_i(x_i(t))\\ &+ (1 - q_i(x_i(t)))Q_i(x_i(t)).\end{aligned} \tag{19}$$

The matrix $Q_i(x_i(t))$ is positive semidefinite by the definition in (4). In view of Lemma 1, we know $1 - q_i(x_i(t)) \geq 0$. Furthermore, by direct calculation for any vector $y_i = (y_{i1}, y_{i2}, \cdots, y_{id})^{\mathsf{T}} \in \mathbb{R}^d$,

$$\begin{aligned}&y_i^{\mathsf{T}}\left[I_d - Q_i(x_i(t))\right]y_i =\\ &\frac{\sum_{1\leq r_1<r_2\leq d}\left[\nabla q_i(x_i(t))_{r_1}y_{ir_2} - \nabla q_i(x_i(t))_{r_2}y_{ir_1}\right]^2}{\sum_{r=1}^{d}\nabla q_i(x_i(t))_r^2},\end{aligned} \tag{20}$$

where $\nabla q_i(x_i(t)) := (\nabla q_i(x_i(t))_1, \nabla q_i(x_i(t))_2, \cdots, \nabla q_i(x_i(t))_d)^{\mathsf{T}} \in \mathbb{R}^d$.

It is obvious from (20) that $I_d - Q_i(x_i(t))$ is a positive semidefinite matrix. Moreover, $y_i^{\mathsf{T}}\left[I_d - Q_i(x_i(t))\right]y_i = 0$ if and only if $\nabla q_i(x_i(t))_{r_1}y_{ir_2} = \nabla q_i(x_i(t))_{r_2}y_{ir_1}$ for any $1 \leq r_1 < r_2 \leq d$. Combining the above comments with (18) and (19) we know that $\dot{V}(x(t)) \leq 0$ for $t \geq 0$. In view of the definition in (5)-(7), we readily complete the proof of Lemma 2 by using LaSalle's invariance principle.

PROOF OF LEMMA 3

Recall that $X = \cap_{i=1}^n X_i$ is the intersection of convex constraint sets. We will prove the result by using an argument of contradiction. Suppose there is a node $k \in [n]$ such that $y_k^* \neq 0_d$. By using (2),

$$\begin{aligned}&\frac{-y_k^*}{\sum_{j\in\mathcal{N}_k} a_{kj} + \sum_{l=1}^m b_{kl}}\\ =&x_k^* - \frac{\sum_{j\in\mathcal{N}_k} a_{kj}x_j^* + \sum_{l=1}^m b_{kl}u_l}{\sum_{j\in\mathcal{N}_k} a_{kj} + \sum_{l=1}^m b_{kl}}.\end{aligned} \tag{21}$$

By our condition, $x_k^* \in X \subseteq X_k$. Suppose that $x_k^* \in X_k \backslash \partial X_k$, which is not empty by Assumption 1. Under this assumption, we know $q_k(x_k^*) < 1$. Note that

$$\begin{aligned}I_d - q_k(x_k^*)Q_k(x_k^*) =&I_d - Q_k(x_k^*)\\ &+ (1 - q_k(x_k^*))Q_k(x_k^*).\end{aligned} \tag{22}$$

By the definition in (4), $Q_k(x_k^*)$ is positive semidefinite. Using the comment in Lemma 2, if for some vector $y_k \in \mathbb{R}^d$ we have $y_k^{\mathsf{T}} Q_k(x_k^*) y_k = 0$, then $y_k^{\mathsf{T}}[I_d - Q_k(x_k^*)]y_k > 0$. In other words, the matrix expressed by (22) must be positive definite even though both of the two additive parts are just positive semidefinite. Since $x^* = (x_1^{*\mathsf{T}}, x_2^{*\mathsf{T}}, \cdots, x_n^{*\mathsf{T}})^{\mathsf{T}} \in \mathcal{Y}$ in (5), the node $k \notin \mathcal{Q}$. But again, since (5) holds, $y_k^* = 0_d$. This contradicts our assumption. Hence, Lemma 3 is proved under the additional assumption $x_k^* \in X_k \backslash \partial X_k$.

Now, suppose $x_k^* \in \partial X_k$. Notice that the second term on the righthand side of (21) sits inside $X \subseteq X_k$ by our conditions and Assumption 2. Therefore, the vector $y_k^*/(\sum_{j\in\mathcal{N}_k} a_{kj} + \sum_{l=1}^m b_{kl})$ points towards the inside of X_k at the boundary point x_k^*. In light of the geometric property of gradient, we have $\nabla q_k(x_k^*)^{\mathsf{T}} y_k^* < 0$. By (6), $k \in \mathcal{V}\backslash\mathcal{Q}$. Since $x^* = (x_1^{*\mathsf{T}}, x_2^{*\mathsf{T}}, \cdots, x_n^{*\mathsf{T}})^{\mathsf{T}} \in \mathcal{Y}$ in (5), we derive $y_k^* = 0_d$. This contradicts our assumption, and accordingly completes the entire proof of Lemma 3.

PROOF OF LEMMA 4

We shall prove the result by using an argument of contradiction. Suppose $\mathcal{Y} \neq \{x^* : \sum_{i=1}^n y_i^{*\mathsf{T}} y_i^* = 0\}$. By (5), there must be some node $k_0 \in \mathcal{Q}$. In view of (6), $y_{k_0}^* \neq 0_d$. Define

$$\Delta := \max\Big\{ \max_{i\in\mathcal{V}} \mathrm{d}(x_i^*, X), \max_{l:\ b_{il}>0} \mathrm{d}(u_l, X)\Big\}, \tag{23}$$

where $\mathrm{d}(\cdot, X)$ represents the distance of a vector from X. By Assumptions 1 and 2, Δ is well defined. Using Lemma 3, we know there is a node $k_1 \in \mathcal{V}$ satisfying $x_{k_1}^* \notin X$. Since X is closed, $\Delta > 0$. We will examine the following three cases. (i) There is a node $k \in \mathcal{V}$ such that $\Delta = \mathrm{d}(x_k^*, X)$ and $y_k^* = 0_d$; (ii) For every i satisfying $\Delta = \mathrm{d}(x_i^*, X)$, $y_i^* \neq 0_d$ holds; and (iii) $\Delta = \mathrm{d}(u_l, X)$ for some $l \in [m]$. In each of these cases, we will aim to derive a contradiction.

(i). There is a node $k \in \mathcal{V}$ such that $\Delta = \mathrm{d}(x_k^*, X)$ and $y_k^* = 0_d$. Denote by $\tilde{x} \in \partial X$ be the projection of x_k^* on X and write the vector $h := x_k^* - \tilde{x} \in \mathbb{R}^d$. Hence, $\|h\| = \Delta$.

Define a hyperplane $\mathcal{H} := \{y \in \mathbb{R}^d : h^{\mathsf{T}} y = h^{\mathsf{T}} x_k^*\}$ passing the point x_k^*, which is parallel to the tangent space $\{y \in \mathbb{R}^d : h^{\mathsf{T}} y = h^{\mathsf{T}} \tilde{x}\}$ at $\tilde{x}$. By (23) and the convexity of X, we have

$$h^{\mathsf{T}} x_i^* \leq h^{\mathsf{T}} x_k^* \qquad \text{and} \qquad h^{\mathsf{T}} u_l^* \leq h^{\mathsf{T}} x_k^* \tag{24}$$

for all $i \in [n]$ and $l \in [m]$ with $b_{kl} > 0$. Hence, all $\{x_i^*\}_{i\in[n]}$ and $\{u_l\}_{l:b_{kl}>0}$ are on the same side of the hyperplane $\mathcal{H}$.

Recall that we have $y_{k_0}^* \neq 0_d$. Since $\mathcal{G}$ is connected, this implies there exists a node $i_1 \in \mathcal{N}_k$ such that $x_{i_1}^* \neq x_k^*$ by (5) and Lemma 2. In light of (23), we know that $\mathrm{d}(x_{i_1}^*, X) \leq \Delta$. Since X is a convex set, the point $x_{i_1}^*$ is not located on the hyperplane $\mathcal{H}$ and thus

$$h^{\mathsf{T}} x_{i_1}^* < h^{\mathsf{T}} x_k^*. \tag{25}$$

By our assumption, $0_d = y_k^* = -\sum_{j\in\mathcal{N}_k} a_{kj}(x_k^* - x_j^*) - \sum_{l=1}^m b_{kl}(x_k^* - u_l)$. We obtain

$$0_d = -\sum_{j\in\mathcal{N}_k} a_{kj} h^{\mathsf{T}}(x_k^* - x_j^*) - \sum_{l=1}^m b_{kl} h^{\mathsf{T}}(x_k^* - u_l). \tag{26}$$

Combining (25), (26) and the fact that all coefficients a_{kj} and b_{kl} are nonnegative, we conclude that there exists $i_1 \in \mathcal{N}_k$ satisfying $h^{\mathsf{T}} x_{i_1}^* > h^{\mathsf{T}} x_k^*$ or u_l with $b_{kl} > 0$ satisfying $h^{\mathsf{T}} u_l^* > h^{\mathsf{T}} x_k^*$. However, this conflicts with (24).

(ii). For every i satisfying $\Delta = \mathrm{d}(x_i^*, X)$, $y_i^* \neq 0_d$ holds. We fix a node $k \in [n]$ satisfying $\Delta = \mathrm{d}(x_k^*, X)$ and $y_k^* \neq 0_d$. In view of (5), we know $k \in \mathcal{Q}$. Accordingly,

$$q_k(x_k^*) \in (0, 1] \qquad \text{and} \qquad \nabla q_k(x_k^*)^{\mathsf{T}} y_k^* > 0 \tag{27}$$

by (6).

Since $x^* \in \mathcal{Y}$, by (5) we have

$$\sum_{i\in\mathcal{Q}} y_i^{*\mathsf{T}} \left[I_d - q_i(x_i^*)Q_i(x_i^*)\right] y_i^* + \sum_{i\in\mathcal{V}\backslash\mathcal{Q}} y_i^{*\mathsf{T}} y_i^* = 0 \tag{28}$$

and hence $y_k^{*\mathsf{T}} \left[I_d - q_k(x_k^*)Q_k(x_k^*)\right] y_k^* = 0$. Furthermore,

$$\begin{aligned}0 &= y_k^{*\mathsf{T}} \left[I_d - q_k(x_k^*)Q_k(x_k^*)\right] y_k^*\\ &= q_k(x_k^*) y_k^{*\mathsf{T}} \left[I_d - Q_k(x_k^*)\right] y_k^* + [1 - q_k(x_k^*)] y_i^{*\mathsf{T}} y_i^*.\end{aligned} \tag{29}$$

Noting that (27) holds and that $I_d - Q_k(x_k^*)$ is positive semidefinite, we obtain

$$y_k^{*\mathsf{T}} \left[I_d - Q_k(x_k^*)\right] y_k^* = 0 \tag{30}$$

and $q_k(x_k^*) = 1$ by (29). Thereby we derive $x_k^* \in \partial X_k$ and

$$\nabla q_k(x_k^*)_{r_1} y_{kr_2}^* = \nabla q_k(x_k^*)_{r_2} y_{kr_1}^* \tag{31}$$

for all $1 \leq r_1 < r_2 \leq d$ following the same analysis of Lemma 2, where $y_k^* = (y_{k1}, y_{k2}, \cdots, y_{kd}^*)^{\mathsf{T}} \in \mathbb{R}^d$.

Let $\tilde{x} \in \partial X$ denote the projection of x_k^* on X and write the vector $h = x_k^* - \tilde{x} \in \mathbb{R}^d$. Hence, $\|h\| = \Delta$. Similarly as in the case (i), we introduce the hyperplane $\mathcal{H} := \{y \in \mathbb{R}^d : h^{\mathsf{T}} y = h^{\mathsf{T}} x_k^*\}$ at x_k^*, which is parallel to the tangent space $\{y \in \mathbb{R}^d : h^{\mathsf{T}} y = h^{\mathsf{T}} \tilde{x}\}$ passing the point $\tilde{x}$. By the definition of k and noting that X is a closed convex set, we have

$$h^{\mathsf{T}} x_i^* \leq h^{\mathsf{T}} x_k^* \qquad \text{and} \qquad h^{\mathsf{T}} u_l^* \leq h^{\mathsf{T}} x_k^* \tag{32}$$

for all $i \in [n]$ and $l \in [m]$ with $b_{kl} > 0$. Hence, $\{x_i^*\}_{i\in[n]}$ and $\{u_l\}_{l:b_{kl}>0}$ are on the same side of the hyperplane $\mathcal{H}$. It follows from (7) and (32) that

$$\begin{aligned}&- h^{\mathsf{T}} \sum_{i\in\mathcal{N}_k} a_{ki}(x_k^* - x_i^*) - h^{\mathsf{T}} \sum_{l=1}^m b_{kl}(x_k^* - u_l)\\ =&h^{\mathsf{T}} y_k^* \leq 0.\end{aligned} \tag{33}$$

Since $x_k^* \in \partial X_k$ and $\tilde{x} \in X \subseteq X_k$, the vector $-h$ based at x_k^* points towards the inside of X_k. Moreover, the vector $\nabla q_k(x_k^*)$ is normal to the tangent space of X_k at x_k^*, and it points towards the outside of X_k. Consequently, $h^\mathsf{T}\nabla q_k(x_k^*) > 0$. By (27), we have $h^\mathsf{T}\nabla q_k(x_k^*)\nabla q_k(x_k^*)^\mathsf{T} y_k^* > 0$. Thanks to the relationship (31), this can be rewritten as follows

$$\begin{aligned} 0 &< h^\mathsf{T}\nabla q_k(x_k^*)\nabla q_k(x_k^*)^\mathsf{T} y_k^* \\ &= \sum_{r_1,r_2=1}^{d} h_{r_1}\nabla q_k(x_k^*)_{r_1}\nabla q_k(x_k^*)_{r_2} y_{kr_2}^* \\ &= \sum_{r_1,r_2=1}^{d} h_{r_1}\nabla q_k(x_k^*)_{r_2}^2 y_{kr_1}^* \\ &= h^\mathsf{T} y_k^* \sum_{r_2=1}^{d} \nabla q_k(x_k^*)_{r_2}^2, \end{aligned} \tag{34}$$

where $h = (h_1, h_2, \cdots, h_d)^\mathsf{T} \in \mathbb{R}^d$ and $y_k^* = (y_{k1}^*, y_{k2}^*, \cdots, y_{kd}^*)^\mathsf{T} \in \mathbb{R}^d$. Hence, $h^\mathsf{T} y_k^* > 0$. This conflicts with (33).

(iii). $\Delta = \mathrm{d}(u_l, X)$ for some $l \in [m]$. If $\tilde{\Delta} := \max_{i\in[n]} \mathrm{d}(x_i^*, X) > 0$, then we can replace Δ with $\tilde{\Delta}$. This case can be proved along the same lines as above. Otherwise, we have $x_i^* \in X$ for any $i \in [n]$. This conflicts with the condition $x_{k_1}^* \notin X$.

Proof of Theorem 1

It follows from Lemma 4 and (7) that

$$-\sum_{j\in\mathcal{N}_i} a_{ij}(x_i^* - x_j^*) - \sum_{l=1}^{m} b_{il}(x_i^* - u_l) = 0 \tag{35}$$

for all $i \in [n]$. Let $u = (u_1^\mathsf{T}, u_2^\mathsf{T}, \cdots, u_m^\mathsf{T})^\mathsf{T} \in \mathbb{R}^{md}$. It is straightforward to check that

$$(L_B \otimes I_d)(x_1^{*\mathsf{T}}, x_2^{*\mathsf{T}}, \cdots, x_n^{*\mathsf{T}})^\mathsf{T} = (B \otimes I_d)u, \tag{36}$$

where $L_B = L + \mathrm{diag}(B1_m) \in \mathbb{R}^{n\times n}$.

Since L_B is a symmetric and diagonally dominant matrix, all eigenvalues of L_B are nonnegative [29]. Suppose the smallest eigenvalue is $\lambda = 0$. Let $\gamma \in \mathbb{R}^n$ be the normalized eigenvector associated with λ. This means $0_n = L_B\gamma = L\gamma + \mathrm{diag}(B1_m)\gamma$. Multiplying γ^T on both sides of the expression yields

$$0 = \gamma^\mathsf{T} L_B \gamma = \gamma^\mathsf{T} L\gamma + \gamma^\mathsf{T}\mathrm{diag}(B1_m)\gamma. \tag{37}$$

Notice that $\gamma^\mathsf{T} L\gamma \geq 0$ and $\gamma^\mathsf{T}\mathrm{diag}(B1_m)\gamma \geq 0$. The equality (37) implies

$$\gamma^\mathsf{T} L\gamma = 0 \qquad \text{and} \qquad \gamma^\mathsf{T}\mathrm{diag}(B1_m)\gamma = 0. \tag{38}$$

Since $\mathcal{G}$ is connected, the first equality in (38) indicates $\gamma = \frac{1}{\sqrt{n}}1_n$. Hence, $\gamma^\mathsf{T}\mathrm{diag}(B1_m)\gamma > 0$ as there is at least one informed agent. This conflicts with the second equality in (38). Hence, by deducing contradiction, we have proved the smallest eigenvalue is $\lambda > 0$. Thus, L_B is invertible. It then follows from (36) that $x^* = (L_B^{-1}B \otimes I_d)u$.

We can write L_B as $L_B = sI_n - S$ with

$$s = \max_{i\in[n]} \Big\{ \sum_{j=1}^{n} a_{ij} + \sum_{l=1}^{m} b_{il} \Big\} \tag{39}$$

and $S \in \mathbb{R}^{n\times n}$ being a nonnegative matrix. Moreover, by using the Gershgorin disc theorem, the spectral radius of S is not larger than s. Therefore, L_B is an M-matrix. Lemma 5 implies that L_B^{-1} is nonnegative.

By the definition of L_B and B, we have

$$(L_B, -B)1_{n+m} = 0_n. \tag{40}$$

In other words, $L_B 1_n - B1_m = 0_n$ and hence $L_B^{-1}B1_m = 1_n$. As all elements of L_B^{-1} and B are nonnegative, $x_i^* \in \mathrm{co}(\mathcal{U})$ for $i \in [n]$. By Lemma 1, $x_i^* \in X_i \cap \mathrm{co}(\mathcal{U})$. Applying Lemma 2, we obtain $x_i(t) \to X_i \cap \mathrm{co}(\mathcal{U})$ for $i \in [n]$ as $t \to \infty$. Namely, the constrained containment can be achieved.

If $u_l \equiv \bar{u}$ for all $l \in [m]$, $\mathrm{co}(\mathcal{U}) = \{\bar{u}\} \subseteq X$. An application of Lemma 2 yields $x_i(t) \to \bar{u}$ for $i \in [n]$ as $t \to \infty$. The constrained consensus tracking is achieved. This completes the proof of Theorem 1.

Proof of Corollary 1

We can prove Corollary 1 following the same line of Theorem 1 through a series of lemmas. In particular, Lemmas 1-4 will be updated by the following Lemmas 6-9.

Lemma 6. *Consider the system modeled by (1) with (8) and (9) over the multiagent network $\mathcal{G} = (\mathcal{V}, \mathcal{E}, A)$. With Assumptions 1 and 4 satisfied, $x_i(t) \in \theta_i^{-1}X_i$ for any $i \in \mathcal{V}$ and $t \geq 0$.*

Let $\mathrm{diag}(\theta_1, \cdots, \theta_n) \in \mathbb{R}^{n\times n}$ be the diagonal matrix with diagonal objects $\theta_1, \cdots, \theta_n$. Let $\otimes$ be the Kronecker product [29]. Write $x^* = (x_1^{*\mathsf{T}}, x_2^{*\mathsf{T}}, \cdots, x_n^{*\mathsf{T}})^\mathsf{T} \in \mathbb{R}^{nd}$. Define a set $\mathcal{Y}' \in \mathbb{R}^{nd}$ as follows

$$\begin{aligned} \mathcal{Y}' := \Big\{ &(\mathrm{diag}(\theta_1, \cdots, \theta_n) \otimes I_d)x^* : \\ &\sum_{i\in\mathcal{Q}'} |\theta_i| y_i^{*\mathsf{T}} \left[I_d - q_i(\theta_i x_i^*)Q_i(\theta_i x_i^*)\right] y_i^* \\ &+ \sum_{i\in\mathcal{V}\setminus\mathcal{Q}'} |\theta_i| y_i^{*\mathsf{T}} y_i^* = 0 \Big\}, \end{aligned} \tag{41}$$

where

$$\mathcal{Q}' := \{i \in \mathcal{V} : q_i(\theta_i x_i^*) \in (0, 1], \theta_i \nabla q_i(\theta_i x_i^*)^\mathsf{T} y_i^* > 0\} \tag{42}$$

and

$$\begin{aligned} y_i^* := &-\mathrm{sgn}(\theta_i) \sum_{j\in\mathcal{N}_i} a_{ij}(\theta_i x_i^* - \theta_j x_j^*) \\ &- \sum_{l=1}^{m} b_{il}(\theta_i x_i^* - u_l), \qquad i \in \mathcal{V}. \end{aligned} \tag{43}$$

Lemma 7. *Let $x(t) = (x_1(t)^\mathsf{T}, x_2(t)^\mathsf{T}, \cdots, x_n(t)^\mathsf{T})^\mathsf{T} \in \mathbb{R}^{nd}$ be a solution of the system (1) with (8) and (9) over the multiagent network $\mathcal{G} = (\mathcal{V}, \mathcal{E}, A)$. With Assumptions 1 and 4 satisfied, the set of limit points of $(\mathrm{diag}(\theta_1, \cdots, \theta_n) \otimes I_d)x(t)$ is contained in $\mathcal{Y}'$.*

Lemma 8. *Consider the system modeled by (1) with (8) and (9) over the multiagent network $\mathcal{G} = (\mathcal{V}, \mathcal{E}, A)$. With Assumptions 1, 2 and 4 satisfied, if $x_i^* \in \theta_i^{-1}X$ for every $i \in \mathcal{V}$, then*

$y_i^* = 0_d$ for every $i \in \mathcal{V}$, where $0_d \in \mathbb{R}^d$ is the vector of all elements being zero.

Lemma 9. *Consider the system modeled by (1) with (8) and (9) over the multiagent network $\mathcal{G} = (\mathcal{V}, \mathcal{E}, A)$. Under Assumptions 1, 2 and 4, if $\mathcal{G}$ is connected, then $\mathcal{Y}' = \big\{(\mathrm{diag}(\theta_1, \cdots, \theta_n) \otimes I_d)x^* : y_i^* = 0_d\big\}$, where y_i^* is determined by (43).*

REFERENCES

[1] F. Bullo, *Lectures on Network Systems*, Kindle Direct Publishing, 2020

[2] R. Olfati-Saber, R. M. Murray, Consensus problems in networks of agents with switching topology and time-delays. *IEEE Trans. Autom. Contr.*, 49(9)(2004) 1520–1533

[3] J. Qin, Q. Ma, Y. Shi, L. Wang, Recent advances in consensus of multi-agent systems: A brief survey. *IEEE Trans. Ind. Electron.*, 64(6)(2017) 4972–4983

[4] H. Wang, W. Yu, Z. Ding, X. Yu, Tracking consensus of general nonlinear multiagent systems with external disturbances under directed networks. *IEEE Trans. Autom. Contr.*, 64(11)(2019) 4772–4779

[5] J. Huang, W. Wang, C. Wen, J. Zhou, G. Li, Distributed adaptive leader-follower and leaderless consensus control of a class of strict-feedback nonlinear systems: A unified approach. *Automatica*, 118(2020) 109021

[6] M. Ji, G. Ferrari-Trecate, M. Egerstedt, A. Buffa, Containment control in mobile networks. *IEEE Trans. Autom. Contr.*, 53(8)(2008) 1972–1975

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

[8] H. Liang, L. Zhang, Y. Sun, T. Huang, Containment control of semi-Markovian multiagent systems with switching topologies. *IEEE Trans. Syst. Man Cybern. Syst.*, 51(6)(2021) 3889–3899

[9] W. Zou, Y. Huang, C. K. Ahn, Z. Xiang, Containment control of linear multiagent systems with stochastic disturbances via event-triggered strategies. *IEEE Syst. J.*, 14(4)(2020) 4810–4819

[10] S. Xiao, J. Dong, Distributed adaptive fuzzy fault-tolerant containment control for heterogeneous nonlinear multiagent systems. *IEEE Trans. Syst. Man Cybern. Syst.*, 52(2)(2022) 954–965

[11] X. Guo, H. Ma, H. Liang, H. Zhang, Command-filter-based fixed-time bipartite containment control for a class of stochastic multiagent systems. *IEEE Trans. Syst. Man Cybern. Syst.*, 52(6)(2022) 3519–3529

[12] A. Nedic, A. Ozdaglar, P. A. Parrilo, Constrained consensus and optimization in multi-agent networks. *IEEE Trans. Autom. Contr.*, 55(4)(2010) 922–938

[13] P. Lin, W. Ren, Constrained consensus in unbalanced networks with communication delays. *IEEE Trans. Autom. Contr.*, 59(3)(2014) 775–781

[14] J. Fu, Y. Lv, G. Wen, X. Yu, T. Huang, Velocity and input constrained coordination of second-order multi-agent systems with relative output information. *IEEE Trans. Netw. Sci. Eng.*, 7(3)(2020) 1925–1938

[15] N. Wang, Y. Wang, G. Wen, M. Lv, F. Zhang, Fuzzy adaptive constrained consensus tracking of high-order multi-agent networks: a new event-triggered mechanism. *IEEE Trans. Syst. Man Cybern. Syst.*, 52(9)(2022) 5468–5480

[16] Z. Zhou, X. Wang, Constrained consensus in continuous-time multiagent systems under weighted graph. *IEEE Trans. Autom. Contr.*, 63(6)(2018) 1686–1693

[17] Y. Shang, Resilient consensus in multi-agent systems with state constraints. *Automatica*, 122(2020) 109288

[18] Y. Shang, Scaled consensus and reference tracking in multiagent networks with constraints. *IEEE Trans. Netw. Sci. Eng.*, 9(3)(2022) 1620–1629

[19] H. Chu, D. Yue, L. Gao, X. Lai, Consensus of multiagent systems with relative state saturations. *IEEE Trans. Syst. Man Cybern. Syst.*, 51(4)(2021) 2391–2402

[20] Y. Shang, Resilient interval consensus in robust networks. *Int. J. Robust Nonlin. Contr.*, 30(17)(2020) 7783–7790

[21] Y. Shang, Interval consensus of switched multiagent systems. *Int. J. Syst. Sci.*, 53(2)(2022) 351–362

[22] H. Su, M. Z. Q. Chen, Multi-agent containment control with input saturation on switching topologies. *IET Contr. Theor. Appl.*, 9(3)(2015) 399-409

[23] T. Liu, J. Qi, Z.-P. Jiang, Distributed containment control of multi-agent systems with velocity and acceleration saturations. *Automatica*, 117(2020) 108992

[24] P. Li, F. Jabbari, X.-M. Sun, Containment control of multi-agent systems with input saturation and unknown leader input. *Automatica*, 130(2021) 109677

[25] Q. Xiong, Q. Zhang, P. Lin, W. Ren, W. Gui, Containment problems for multiagent systems with nonconvex velocity constraints. *IEEE Trans. Cybern.*, 51(9)(2021) 4716–4721

[26] Y. Huang, M. Duan, L. Mo, Multiagent containment control with nonconvex states constraints, nonuniform time delays, and switching directed networks. *IEEE Trans. Neural Netw. Learn. Syst.*, 31(11)(2020) 5021–5028

[27] P. Lin, Y. Huang, Q. Zhang, Z. Yuan, Distributed velocity and input constrained tracking control of high-speed train systems. *IEEE Trans. Syst. Man Cybern. Syst.*, 51(12)(2021) 7882–7888

[28] P. Lin, G. Li, K. Huang, Position-constrained containment for second-order discrete-time multi-agent systems. *Syst. Contr. Lett.*, 142(2020) 104708

[29] R. A. Horn, C. R. Johnson, *Matrix Analysis*. 2nd Ed. Cambridge University Press, New York, 2013

[30] S. Boyd, L. Vandenberghe, *Convex Optimization*, Cambridge University Press, Cambridge, 2004

[31] M. A. Lopez, S. Reisner, Hausdorff approximation of convex polygons. *Comput. Geom.*, 32(2)(2005) 139–158

[32] H.-K. Ahn, S. W. Bae, O. Cheong, J. Gudmundsson. *Discrete Comput. Geom.*, 40(2008) 414–429

[33] Y. Shang, Consensus formation in networks with neighbor-dependent synergy and observer effect. *Commun. Nonlinear Sci. Numer. Simulat.*, 95(2021) 105632

[34] V. Monga, Y. Li, Y. C. Eldar, Algorithm unrolling: Interpretable, efficient deep learning for signal and image processing. *IEEE Signal Process. Mag.*, 38(2)(2021) 18–44

[35] W. M. Haddad, V. Chellaboina, *Nonlinear Dynamical Systems and Control: A Lyapunov-Based Approach*, Princeton University Press, New Jersey, 2008

[36] R. Bru, F. Pedroche, D. B. Szyld, Subdirect sums of nonsingular M-matrices and of their inverses. *Electron. J. Linear Algebra*, 13(2005) 162–174