Distributed formation stabilization for mobile agents using virtual tensegrity structures

Yang, Qingkai; Cao, Ming; Fang, Hao; Chen, Jie

Distributed formation stabilization for mobile agents using virtual tensegrity structures

YANG Qingkai[1,3], CAO Ming[2], FANG Hao[1,3] CHEN Jie[1,3] HUANG Jie[1,3]

1. School of Automation, Beijing Institute of Technology, Beijing 100081, P. R. China
E-mail: fangh@bit.edu.cn

2. Faculty of Mathematics and Natural Sciences, ITM, University of Groningen, Groningen 9747 AG, The Netherlands

3. Key Laboratory of Intelligent Control and Decision of Complex Systems, Beijing 100081, P. R. China

Abstract: This paper investigates the distributed formation control problem for a group of mobile Euler-Lagrange agents to achieve global stabilization by using virtual tensegrity structures. Firstly, a systematic approach to design tensegrity frameworks is elaborately explained to confine the interaction relationships between agents, which allows us to obtain globally rigid frameworks. Then, based on virtual tensegrity frameworks, distributed control strategies are developed such that the mobile agents converge to the desired formation globally. The theoretical analysis is further validated through simulations.

Key Words: Formation stabilization, Tensegrity structure, Global convergence, Euler-Lagrange dynamics

1 Introduction

In recent years, distributed control of multi-agent systems has attracted a significant amount of research efforts due to its broad applications, such as search and rescue, area coverage and reconnaissance, and exploration in unknown environment [1–3]. Among various topics of coordinated control, one active research direction is the formation control problem, where the mobile agents are guided to a prescribed formation, likely then maneuvering as a cohesive whole.

Even though a wide range of issues have been studied, and hence several theoretical frameworks have been established to design control strategies, see, for example, [4] [5] establishing estimation strategy for Euler-Lagrange systems with partial states available, [6][7] using matrix theory and graph theory, [8] based on gradient-descent control approach, graph rigidity theory [9][10], networked small-gain theory [11], sample-data for circle formation [12], to name a few, it should be noted that the desired formation shape can only be guaranteed to be locally stable in most of the research. In particular, based on the graph rigidity approach, it is challenging to coordinate a group of mobile robots globally converging to the prescribed formation [13].

Efforts have been made on the topic of global stability of distributed formation control. For instance, the global behavior of three agents maintaining triangular formations is discussed in [14][15], where distance based gradient-like control laws are proposed, respectively. To analyze global stability for autonomous robots, a differential geometric approach is addressed and applied to the triangular formation control [16]. The global asymptotic performance is achieved by adding an adaptive perturbation to any agent's movement direction in [17]. It is worth mentioning that the control strategies in these works are only valid in the case of three agents forming triangular formations, which requires all-to-all interactions. Besides, the position estimation based formation control problem for single-integrators in the plane is

studied in [18]. It has been shown that the global convergence can be realized if and only if the interaction graph has a spanning tree.

In contrast to previous work, we focus on dealing with the distributed formation stabilization problem for the configurations in *general position* [1] in the Euclidean space of any dimension. Motivated by the deployable and stable properties of *tensegrity structure* [20], we propose to use such a virtual structure, a class of geometry structures from architectural engineering, to analyze the characteristics of global stability for a set of mobile agents modeled by Euler-Lagrange equations. In this paper, we firstly design a novel algorithm to compute the sparse stress matrix based on the given desired configuration, whose elements determine the members of the structure. Then, the virtual tensegrity structure will be constructed through the mapping between the agents (resp. edges) and the nodes (resp. inextendable cables and incompressible struts). Finally, under the interaction constrains, we propose distributed control strategies to steer the agents to prescribed formation globally up to translation.

The applications of tensegrity structure in formation control have gradually draw the researchers' attentions, see, e.g.,[21–23]. However, most of the existing results are only applicable to the one-dimensional (collinear shape) [21] or planar formations [22][23]. In addition, even though in [23], the construction of virtual tensegrity structure has been taken into consideration, the proposed algorithm is highly likely to result in complete underlying graph, which is not practical in most of the applications.

The main contributions of this paper lie in a set of new methodologies to achieve global stability in distributed formation control using virtual tensegrity structures. More precisely, we propose a novel algorithm to assign the members among all the agents, such that universally (thus globally) rigid tensegrity structures can be obtained. The distinct point here is that we can guarantee the global property without requirement for complete graphs based on our algorithm. Further, we effectively apply the virtual tensegrity structures in

[1]A configuration is in general position if no k points lie in a $(k-1)$ dimensional affine space for $1 \le k \le d$ [19].

the formation control for a group of nonlinear mobile agents, yielding global convergence to desired formation shapes up to translation.

2 Problem formulation

We consider a team of $n > 1$ fully actuated, heterogeneous mobile robots, each of which is modeled by a Euler-Lagrange system

$$M_i(q_i)\ddot{q}_i + C_i(q_i, \dot{q}_i)\dot{q}_i = \tau_i, \quad i = 1, \cdots, n \quad (1)$$

where $q_i \in \mathbb{R}^m$ is the generalized coordinate of robot i in some fixed coordinate system, $M_i(q_i) \in \mathbb{R}^m \times \mathbb{R}^m$ is robot i's inertia matrix that is symmetric and positive definite, $C_i(q_i, \dot{q}_i) \in \mathbb{R}^m \times \mathbb{R}^m$ is the Coriolis and centripetal term satisfying the property that $\dot{M}_i(q_i) - 2C_i(q_i, \dot{q}_i)$ is skew symmetric, and $\tau_i \in \mathbb{R}^m$ is the control input. We call $q = (q_1, q_2, \cdots, q_n) \in \mathbb{R}^{m \times n}$ the *configuration* of the robotic team.

In addition, the left-hand side of the dynamics (1) can be linearly parameterized as:

$$M_i(q_i)x + C_i(q_i, \dot{q}_i)y = Y_i(q_i, \dot{q}_i, y, x)\Theta_i, \quad \forall x, y \in \mathbb{R}^m \quad (2)$$

where $Y_i(q_i, \dot{q}_i, x, y)$ is the known regressor and Θ_i is a constant parameter vector but unknown.

The neighboring relationships between the robots are defined by an undirected graph $\mathbb{G}$ with the vertex set $\mathcal{V} = \{1, 2, \cdots, n\}$ and the edge set $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ where there is an edge (i, j) if and only if robots i and j are neighbors of each other. We use $\mathcal{N}_i$ to denote the set of neighbors of robot i. $\mathbb{G}$ is embedded in $\mathbb{R}^m$ when $q = (q_1, q_2, \cdots, q_n)$ is realized and the pair $(\mathbb{G}, q)$ is called a *framework*. Two frameworks $(\mathbb{G}, q)$ and $(\mathbb{G}, p)$ are said to be *equivalent* if the distance between q_i and q_j is always the same as that between p_i and p_j whenever $(i, j) \in \mathcal{E}$. Now we formulate the formation stabilization problem as follows.

Given a desired configuration q^* for the team of n fully actuated robots modeled by Euler-Lagrange agents (1), first assign neighbor relationships to the team, to be described by $\mathbb{G}$, and then for each robot $i = 1, \ldots, n$, design distributed control laws $\tau_i(q_i - q_j, \dot{q}_i - \dot{q}_j), j \in \mathcal{N}_i$, such that the robots' positions are driven to the target set

$$\mathcal{T} = \{q \in \mathbb{R}^{mn} | q_i - q_j = q_i^* - q_j^*, \quad \forall (i, j) \in \mathcal{E}\}. \quad (3)$$

Obviously, to make the control less complicated and scalable with n, $\mathbb{G}$ is better to be sparse than dense. In order to solve the formation stabilization problem that have just been formulated, we will propose control laws by assigning a virtual tensegrity frameworks to the given formation. Towards this end, we first introduce the related notions and properties of tensegrity frameworks.

3 Tensegrity

In this section, we follow the convention in [20, 24] to present a brief overview of tensegrity frameworks. A *tensegrity* $\mathbb{T}(\mathbb{G})$ is obtained by embedding an undirected graph $\mathbb{G}$ in a Euclidean space and replacing each edge of $\mathbb{G}$ by an inextensible *cables* or incompressible *struts*, or inextensible and incompressible *bars*. Together all the cables, struts

and bars are called the *members* of $\mathbb{T}$ and the embedded vertices of $\mathbb{G}$ are called the *nodes* of $\mathbb{T}$. So the same graph $\mathbb{G}$ may lead to different tensegrity frameworks when $\mathbb{G}$'s edges are realized into different combinations of cables, struts and bars.

We use the labels in the vertex set $\mathcal{V}$ of $\mathbb{G}$ for the nodes of $\mathbb{T}$. For each member (i, j) of $\mathbb{T}$, we assign a scalar $\omega_{ij} = \omega_{ji}$, and use $\omega \in \mathbb{R}^{|\mathcal{E}|}$, where $|\mathcal{E}|$ is the number of members of $\mathbb{T}$, to denote the concatenated vector $\omega = (\cdots, \omega_{ij}, \cdots)^T$. Then ω is called a *stress* of $\mathbb{T}$; if further, each ω_{ij} satisfies $\omega_{ij} \geq 0$ whenever (i, j) is a cable and $\omega_{ij} \leq 0$ whenever (i, j) is a strut, then ω is said to be a *proper* stress.

For a given tensegrity $\mathbb{T}$, when its nodes are embedded in different locations, it corresponds to different configurations q and consequently corresponds to different frameworks $(\mathbb{G}, q)$ with different geometric shapes. Let q^* be the configuration that defines the desired shape. Then we call that ω an *equilibrium stress* of $\mathbb{T}$ if it is a solution to the equation set

$$\sum_{j \in \mathcal{N}_i} \omega_{ij}(q_j^* - q_i^*) = \mathbf{0}, \quad i = 1, \cdots, n. \quad (4)$$

Given ω, the associated *stress matrix* Ω is defined by letting $\Omega_{ij} = -\omega_{ij}$ for $i \neq j$ and $\Omega_{ii} = \sum_{j \neq i} \omega_{ij}$ for $i = 1, \ldots, n$.

For a tensegrity $\mathbb{T}$ with the desired configuration q^*, we are interested in its associated configurations p that satisfy the following *tensegrity constraints*

$$\begin{cases} |p_i - p_j| \leq |q_i^* - q_j^*|, & \text{when } (i, j) \text{ is a cable,} \\ |p_i - p_j| \geq |q_i^* - q_j^*|, & \text{when } (i, j) \text{ is a strut and} \\ |p_i - p_j| = |q_i^* - q_j^*|, & \text{when } (i, j) \text{ is a bar.} \end{cases} \quad (5)$$

Such constraints can be naturally used to define the "rigidity" properties of $\mathbb{T}$. We say that the tensegrity $\mathbb{T}$ whose shape is determined by the configuration q^* is *rigid* if its any other configuration p is always congruent to q^* whenever p is sufficiently close to q^* and satisfies the tensegrity constraints (5); furthermore, if the congruent relationship between p and q^* holds for all $p \in \mathbb{R}^{m \times n}$, then we say $\mathbb{T}$ is *globally rigid*; and even more strongly, if this congruent relationship still holds for all q living in any higher-dimensional spaces than $\mathbb{R}^{m \times n}$, we say $\mathbb{T}$ is *universally rigid*.

There are several conditions to guarantee the rigidity of a tensegrity framework. We list one of them below.

Lemma 1. *[25] Let $(\mathbb{G}, p)$ be an r-dimensional tensegrity framework on n vertices in $\mathbb{R}^r$, for some $r \leq n - 2$. Then $(\mathbb{G}, p)$ is universally rigid if the following two conditions hold.*

1). *$(\mathbb{G}, p)$ admits a proper positive semidefinite stress matrix Ω with rank $n - r - 1$.*
2). *Vertex i and its neighbors are in general position in $\mathbb{R}^r$, $\forall i = 1, \cdots, n$.*

With the knowledge about tensegrity frameworks and their rigidity properties at hand, now we are ready to propose our solutions to the formation stabilization.

4 Formation stabilization

We first deal with the formation stabilization problem. To stabilize the shape of a formation of n mobile robots to a de-

sired configuration q^*, we propose to assign an appropriate virtual tensegrity structure to enforce a number of distance constraints between some pairs of robots; consequently corresponding to those constraints, the tensegrity structure determines which robots need to sense the relative positions of which other robots. The second is to design local control laws for each robot to use its sensed information to maintain the displacement constraints that they are involved.

4.1 Assignment of the virtual tensegrity structure

We take each robot to be a node of a virtual tensegrity whose cables in tension and struts in compression give rise to attractive and repulsive forces between the robots respectively. In this section, we are only interested in universally rigid tensegrity frameworks that have only cables and structs but no bars as their members. Since the row sums of Ω are all zero, $\mathbf{1}_n$ always lives in null(Ω). One can further check that the columns of $(q^*)^T$ are in null(Ω) as well. In fact, the column span of $N \triangleq [(q^*)^T, \mathbf{1}_n]$ always belongs to null(Ω). Therefore,

$$\Omega N = \mathbf{0}_{n \times (m+1)}. \tag{6}$$

Given q^*, to assign a virtual tensegrity $\mathbb{T}$ to the robotic team is equivalent to use N to determine the matrix Ω since once Ω is determined, all the needed cables and structs together with their stresses are determined as well. Obviously, such Ω's are in general not unique and naturally we want to obtain sparse Ω which leads to fewer distance constraints and thus likely simpler controllers. Towards this end, we convert our problem into the *sparse null space problem* first considered in [26], namely, given a $m \times n$ matrix A of rank r, $(r \leq m \leq n)$, to find a sparse $n \times (n-r)$ matrix B such that B is full rank and its column span is null(A) [27]. We take the transpose of both sides of (6), yielding

$$N^T \Omega^T = N^T \Omega = \mathbf{0}_{(m+1) \times n}. \tag{7}$$

From Lemma 1, we need the sparse matrix Ω to be positive semi-definite and $rank(\Omega) = n - d - 1$. However, since Ω in (7) is not full rank, we cannot directly solve the sparse null space problem. Instead, we try to construct a column full-rank matrix $D \in \mathbb{R}^{n \times (n-m-1)}$ such that

$$N^T D = \mathbf{0}_{(m+1) \times (n-m-1)}. \tag{8}$$

If indeed such a D can be constructed, it must be true that

$$N^T D D^T = \mathbf{0}_{(m+1) \times (n-m-1)} D^T = \mathbf{0}_{(m+1) \times n} \tag{9}$$

and hence the matrix $D D^T$ can serve as the stress matrix Ω. So the construction of a sparse matrix Ω is equivalent to the design of such a sparse D. In addition, we make an even stronger requirement that Ω is in its *band form*, whose non-zero entries are confined to be in a diagonal band containing the main diagonal. This additional requirement is motivated by the fact that it is more convenient in practice to have robots to track nearby robots. Now we present our algorithm to construct the stress matrix Ω, which is inspired by the classical "turning back" method for computing the sparse null space basis [28].

Step 1: Construct $\bar{N} \in \mathbb{R}^{(m+1) \times n}$ such that its first $m+1$ columns are linearly independent. Since the configuration is in general position, the natural choice of $\bar{N}$ is N^T.

Step 2: Now we construct D by finding a sparse basis for null($\bar{N}$). We first find the smallest $k_1 > 0$ such that $\bar{N}$'s columns with the indices $m+2, m+1, \ldots, m+2-k_1$ are linearly dependent. We record that the $(m+2-k_1)$th, ..., $(m+2)$th elements of D's first column are nonzero. Then, to record the nonzero positions for the second column of D, finding a smallest $k_2 > 0$ such that columns with the indices $m+3, m+2, \ldots, m+3-k_2$ of $\bar{N}$ are linearly dependent. During this procedure, we do not let the column with index $m+2-k_1$ involve in the second round operation. Again, the indices correspond to the nonzero positions of D's second column. This process finishes until we have determined the positions of the nonzero elements of the last column of D. Naturally, we set all those elements of D that have not been recorded in the process to be zero, and for those that have been recorded as nonzero, we take them to be the solutions to the following equation

$$\bar{N} D = \mathbf{0}_{(m+1) \times (n-m-1)}, \tag{10}$$

which is underdetermined since it is a set of $|D_i| - 1$ linear equations with $|D_i|$ unknowns, where $|D_i|$ denotes the number of nonzeros in column D_i. Hence, we can always find a set of nonzero elements of D and thus fully determine D. In addition, it is easy to check that the constructed D is always column full-rank.

Step 3: Compute the stress matrix based on $\Omega = D D^T$. It can be shown that Ω is positive semi-definite with rank being $n - m - 1$. With the stress matrix, it is straightforward to design the edge set and the corresponding members, namely, the edge (i, j) would be a cable if $\Omega_{ij} < 0$, a strut if $\Omega_{ij} > 0$, and no edge between agent i and j if $\Omega_{ij} = 0$.

Remark 1. *The problem on how to compute the stress matrix is also considered in [29]. However, in general, it is difficult to yield a sparse stress matrix based on his method, which often leads to a complete graph. In this paper, we present an more efficient algorithm not only in determining the sparse stress matrix, also in solving the assignment problem of the virtual tensegrity structure.*

Proposition 1. *Given a configuration $q^* \in \mathbb{R}^m$ in general position, we can construct universally rigid virtual tensegrity structure $\mathbb{T} \in \mathbb{R}^m$, whose underlying graph is not complete.*

Proof. Given configuration $q^* \in \mathbb{R}^m$, we can always find the matrix $\bar{N}$ such that the first $d + 1$ columns are linear independent. Then based on our algorithm, we can obtain the positive semi-definite matrix Ω of rank $n - m - 1$. Consider that the configuration is in general position, hence it can be concluded from Lemma 1 that the constructed tensegrity is universally rigid, thus globally rigid in $\mathbb{R}^m$. □

4.2 Design of the control law

In the proposed tensegrity structure, the edges are represented by the virtual springs of nonzero rest length. The spring constant for two connecting agents i and j is positive scalars satisfying $k_{ij} = k_{ji}$ and the rest length $l_{ij} = -l_{ji}$. Accordingly, the force applied to agent i is given by

$$F_{j \to i} = k_{ij}(r_{ij} - l_{ij}) = -F_{i \to j} \tag{11}$$

where r_{ij} is the relative displacement between agent i and agent j, which is defined as

$$r_{ij} = q_j - q_i \qquad (12)$$

In order to coincide with the stress for cables (struts), we assign the rest length for cables (struts) to be β_{ij}^c (β_{ij}^s) the prescribed displacement between the agents, namely,

$$l_{ij} = \begin{cases} \beta_{ij}^c r_{ij}^* & \text{if } (i,j) \in \mathcal{E}_C \\ \beta_{ij}^s r_{ij}^* & \text{if } (i,j) \in \mathcal{E}_S \end{cases} \qquad (13)$$

where $\beta_{ij}^c \in (0,1)$, $\beta_{ij}^s \in (1,+\infty)$ are constants, and r_{ij}^* is the prescribed displacement between agent i and j, i.e., $r_{ij}^* = q_j^* - q_i^*$. $\mathcal{E}_C$ and $\mathcal{E}_S$ are used to represent the set of cables and struts, respectively. Correspondingly, the spring constant k_{ij} is as follows

$$k_{ij} = \begin{cases} \dfrac{\Omega_{ij}}{1 - \beta_{ij}^c}, & \text{if } \Omega_{ij} < 0 \\ \dfrac{\Omega_{ij}}{1 - \beta_{ij}^s}, & \text{if } \Omega_{ij} > 0 \end{cases} \qquad (14)$$

In the context of virtual springs, the potential energy $P(q)$ caused by the disagreement between r_{ij} and r_{ij}^* is defined as

$$P(q) = \frac{1}{2} \sum_{(i,j) \in \mathcal{E}} k_{ij} \| r_{ij} - r_{ij}^* \|^2 \qquad (15)$$

It is worth mentioning that the virtual cables (struts) are in tension (in compression) at the equilibrium configuration $q^* \in \mathbb{R}^m$ due to $l_{ij} = \beta r_{ij}^*$. Therefore, the formation achieved based on the virtual tensegrity structure has the property of robustness.

We are now left with developing the local control laws driving the agents to formulate the desired formation. In what follows, we shall use M_i and C_i to replace $M_i(q_i)$ and $C_i(q_i, \dot{q}_i)$ for simplification.

Define the auxiliary variable

$$s_i = \dot{q}_i + g_i(q) \qquad (16)$$

where

$$g_i(q) = \frac{\partial P}{\partial q_i} = -\sum_{j \in \mathcal{N}_i} k_{ij}(r_{ij} - r_{ij}^*) \qquad (17)$$

In view of (1) and (16), one has

$$\begin{aligned} M_i \dot{s}_i + C_i s_i &= M_i(\ddot{q}_i + \dot{g}_i(q)) + C_i(\dot{q}_i + g_i(q)) \\ &= \tau_i + M_i \dot{g}_i(q) + C_i g_i(q) \\ &= \tau_i + Y_i(q_i, \dot{q}_i, g_i, \dot{g}_i)\Theta_i \end{aligned} \qquad (18)$$

The distributed control input τ_i is designed as

$$\tau_i = -k_p s_i - g_i(q) - Y_i(q_i, \dot{q}_i, g_i, \dot{g}_i)\hat{\Theta}_i \qquad (19)$$

where k_p is a positive scalar, and $\hat{\Theta}_i$ is the estimation of Θ_i, which is updated according to

$$\dot{\hat{\Theta}}_i = \Gamma_i Y_i(q_i, \dot{q}_i, g_i, \dot{g}_i)^T s_i \qquad (20)$$

where Γ_i is an arbitrary positive definite matrix in compatible dimension.

Theorem 1. *For the networked Euler-Lagrange systems modeled by* (1)*, the agents can be driven to the prescribed formation globally using the control law* (19) *and* (20)*.*

Proof. We introduce the Lyapunov function candidate as

$$V = \frac{1}{2} \sum_{i=1}^{n} (s_i^T M_i s_i + \tilde{\Theta}_i^T \Gamma_i^{-1} \tilde{\Theta}_i) + P(q) \qquad (21)$$

where $\tilde{\Theta}_i = \hat{\Theta}_i - \Theta_i$ is the estimation error, thus $\dot{\tilde{\Theta}}_i = \dot{\hat{\Theta}}_i$. Taking the time derivative of V, we have

$$\dot{V} = \frac{1}{2} \sum_{i=1}^{n} (s_i^T \dot{M}_i s_i + 2 s_i^T M_i \dot{s}_i) + \sum_{i=1}^{n} (\tilde{\Theta}_i^T \Gamma_i^{-1} \dot{\hat{\Theta}}_i + g_i(q)^T \dot{q}_i) \qquad (22)$$

Substituting (18)-(19) into $\dot{V}$ yields

$$\begin{aligned} \dot{V} &= \sum_{i=1}^{n} \left[-k_p s_i^T s_i - s_i^T g_i(q) + g_i(q)^T (s_i - g_i(q)) \right] \\ &= -\sum_{i=1}^{n} (k_p s_i^T s_i + g_i(q)^T g_i(q)) \end{aligned} \qquad (23)$$

Hence, it can be concluded that $\tilde{\Theta}_i \in \mathcal{L}_\infty$, $s_i \in \mathcal{L}_\infty \cap \mathcal{L}_2$, $g_i(q) \in \mathcal{L}_\infty$ and therefore $\dot{q}_i \in \mathcal{L}_\infty$ from (16). Then, it follows that $\dot{g}_i(q) \in \mathcal{L}_\infty$, which further implies $Y_i(q_i, \dot{q}_i, g_i, \dot{g}_i) \in \mathcal{L}_\infty$ and thus $\tau_i \in \mathcal{L}_\infty$ according to (19). We can also get $M_i(q_i)$ and $C_i(q_i, \dot{q}_i)$ are bounded due to the fact that they are only decided by the states q_i and $\dot{q}_i$. Thus, it is straightforward to know that $\dot{s}_i \in \mathcal{L}_\infty$ from (18), which, together with $\dot{g}_i(q) \in \mathcal{L}_\infty$, implies $\ddot{V}(s_i, \dot{s}_i, g_i, \dot{g}_i)$ is bounded. It can be concluded from Barbalat's Lemma that $\dot{V} \to 0$, as $t \to \infty$. Therefore, for each agent i,

$$\lim_{t \to \infty} s_i = \mathbf{0} \qquad (24a)$$

$$\lim_{t \to \infty} g_i(q) = \mathbf{0} \qquad (24b)$$

In view of the definition of variable s_i in (16), it follows

$$\lim_{t \to \infty} \dot{q}_i(t) \to \mathbf{0} \qquad (25)$$

The equation (24b) can be grouped as

$$-(\bar{\mathcal{K}} \otimes I_m) q_e = 0 \qquad (26)$$

where $q_e = [(q_1 - q_1^*)^T, (q_2 - q_2^*)^T, \cdots, (q_n - q_n^*)^T]^T$ and the "spring constant matrix" $\bar{\mathcal{K}} \in \mathbb{R}^{n \times n}$ is defined in the same way as the standard Laplacian matrix, i.e.,

$$\bar{\mathcal{K}}_{ii} = \sum_{j \in \mathcal{N}_i} k_{ij}, \qquad \bar{\mathcal{K}}_{ij} = -k_{ij}, \; i \neq j \qquad (27)$$

Then, it can be concluded directly from Lemma 2.10 of [30] that $q_1 - q_1^* = q_2 - q_2^* = \cdots = q_n - q_n^*$, which implies $r_{ij} = r_{ij}^*$. Consider tthe framework $(\mathbb{G}, q^*)$ is globally rigid with well-designed stresses by Lemma 1. Hence, the agents globally converge to the target set $\mathcal{T}$ in (3), namely, the desired formation is achieved based on the proposed virtual tensegrity structure. $\qquad \square$
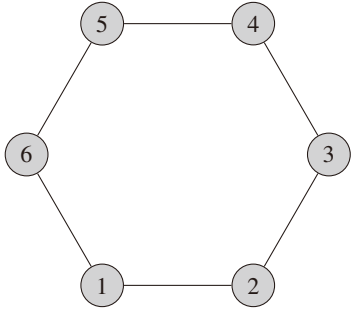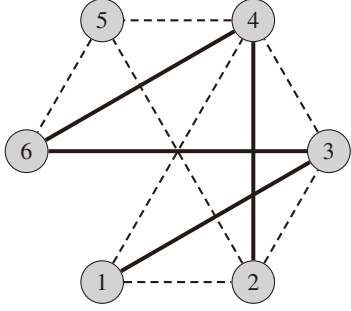
Fig. 1: Desired shape for six agents.

Fig. 2: Proposed tensegrity structure with the dashed lines as cables and the thick solid lines as struts.

Fig. 3: Formation evolution

5 Simulations

In this section, we will validate the theoretical results derived in the preceding sections. Consider a regular hexagon with configuration as follows

$$q^* = \begin{bmatrix} 0 & 2 & 3 & 2 & 0 & -1 \\ 0 & 0 & \sqrt{3} & 2\sqrt{3} & 2\sqrt{3} & \sqrt{3} \end{bmatrix}^T$$

Hence, the corresponding matrix

$$\bar{N} = \begin{bmatrix} 0 & 2 & 3 & 2 & 0 & -1 \\ 0 & 0 & \sqrt{3} & 2\sqrt{3} & 2\sqrt{3} & \sqrt{3} \\ 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix}$$

Then, based on the proposed algorithm proposed, we get

$$D = \begin{bmatrix} -1 & 2 & -2 & 1 & 0 & 0 \\ 0 & -1 & 2 & -2 & 1 & 0 \\ 0 & 0 & -1 & 2 & -2 & 1 \end{bmatrix}^T$$

and the corresponding stress matrix as follows

$$\Omega = \begin{bmatrix} 1 & -2 & 2 & -1 & 0 & 0 \\ -2 & 5 & -6 & 4 & -1 & 0 \\ 2 & -6 & 9 & -4 & 0 & 1 \\ -1 & 4 & -4 & 9 & -6 & 2 \\ 0 & -1 & 0 & -6 & 5 & -2 \\ 0 & 0 & 1 & 2 & -2 & 1 \end{bmatrix}$$

Hence, based on Ω the virtual tensegrity structure is shown in 2.

For simplicity, we assume that $\beta_{ij}^c = 0.8$, $\beta_{ij}^s = 1.25$, $i, j = 1, 2, \cdots, n$. The control gain k_p is set as 2, and the initial coordinates of the six agents are generated randomly as $q_i^* + rands(2, 1) - 0.5 * ones(2, 1)$ through

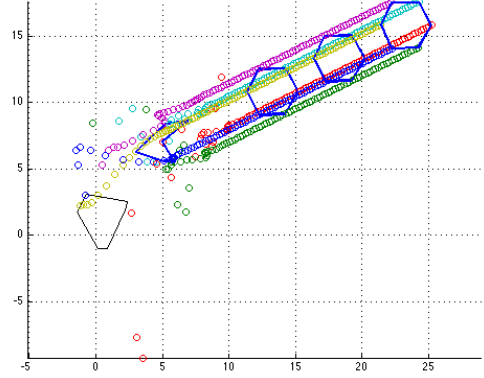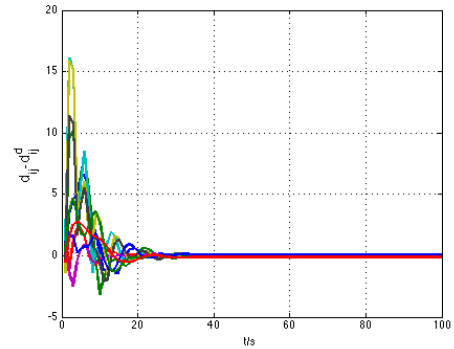Fig. 4: Errors between d_{ij} and d_{ij}^d

Matlab. $d_{ij} = \|r_{ij}\|$, and d_{ij}^d is the desired length between agent i and j. The results are shown in Figure 3-4.

The simulation results using the control law (19) and (20) based on the virtual tensegrity are shown in Fig. 2. It can be seen from Fig. 3 and Fig. 4 that the agents finally evolve into the desired formation. All of these indicate the effectiveness of our proposed virtual tensegrity based formation control strategy.

6 Conclusion

In this paper, we have presented a geometry structure based distributed control for stabilizing a set of mobile agents in space of any dimension. Given the configuration in general position, the proposed algorithm can effectively assign a virtual tensegrity, such that it is universally rigid.

To steer the mobile agents to the target set globally, we have provided the distributed control laws, whose effectiveness are further demonstrated in the simulations.

References

[1] W. Ren, R. Beard, and E. Atkinsand. Information consensus in multivehicle cooperative control. *IEEE Control systems magazine*, 27(2):71–82, 2007.

[2] F. Bullo, J. Cortés, and S. Martinez. *Distributed control of robotic networks: a mathematical approach to motion coordination algorithms*. Princeton University Press, 2009.

[3] N. E. Leonard, D. A. Paley, F. Lekien, R. Sepulchre, D. M. Fratantoni, and R. E. Davis. Collective motion, sensor networks, and ocean sampling. *Proceedings of the IEEE*, 95(1):48–74, 2007.

[4] Q. Yang, H. Fang, J. Chen, and X. Wang. Distributed observer-based coordination for multiple lagrangian systems using only position measurements. *IET Control Theory & Applications*, 8(17):2102–2114, 2014.

[5] Q. Yang, H. Fang, Y. Mao, and J. Huang. Distributed tracking for networked euler-lagrange systems without velocity measurements. *Journal of Systems Engineering and Electronics*, 25(4):671–680, 2014.

[6] J. A. Fax and R. M. Murray. Information flow and cooperative control of vehicle formations. *IEEE Transactions on Automatic Control*, 49(9):1465–1476, 2004.

[7] Z. Lin, B. Francis, and M. Maggiore. Necessary and sufficient graphical conditions for formation control of unicycles. *IEEE Transactions on Automatic Control*, 50(1):121–127, 2005.

[8] L. Krick, M. E. Broucke, and B. A. Francis. Stabilisation of infinitesimally rigid formations of multi-robot networks. *International Journal of Control*, 82(3):423–439, 2009.

[9] B. D. O. Anderson, C. Yu, B. Fidan, and J. M. Hendrickx. Rigid graph control architectures for autonomous formations. *IEEE Control Systems*, 28(6):48–63, 2008.

[10] M. Cao, C. Yu, and B. Anderson. Formation control using range-only measurements. *Automatica*, 47(4):776–781, 2011.

[11] T. Liu and Z. Jiang. Distributed formation control of nonholonomic mobile robots without global position measurements. *Automatica*, 49(2):592–600, 2013.

[12] C. Wang, G. Xie, and M. Cao. Forming circle formations of anonymous mobile agents with order preservation. *IEEE Transactions on Automatic Control*, 58(12):3248–3254, 2013.

[13] J. Cortés. Global and robust formation-shape stabilization of relative sensing networks. *Automatica*, 45(12):2754–2762, 2009.

[14] M. Cao, A. S. Morse, C. Yu, B. D. O. Anderson, and S. Dasgupta. Controlling a triangular formation of mobile autonomous agents. In *46th IEEE Conference on Decision and Control*, pages 3603–3608, 2007.

[15] M. Cao, B. D. O. Anderson, A. S. Morse, and C. Yu. Control of acyclic formations of mobile autonomous agents. In *47th IEEE Conference on Decision and Control.*, pages 1187–1192, 2008.

[16] F. Dorfler and B. Francis. Geometric analysis of the formation problem for autonomous robots. *IEEE Transactions on Automatic Control*, 55(10):2379–2384, 2010.

[17] Q. Wang, Y. Tian, and Y. Xu. Globally asymptotically stable formation control of three agents. *Journal of Systems Science and Complexity*, 25(6):1068–1079, 2012.

[18] K. K. Oh and H. S. Ahn. Formation control of mobile agents based on distributed position estimation. *IEEE Transactions on Automatic Control*, 58(3):737–742, 2013.

[19] R. Connelly and S. Gortler. Iterative universal rigidity. *arXiv*, 1401.7029v1, 2014.

[20] R. Connelly. Tensegrities and global rigidity. In *Shaping Space*, pages 267–278. Springer, 2013.

[21] D. Pais, M. Cao, and N. E. Leonard. Formation shape and orientation control using projected collinear tensegrity structures. In *Proceedings of the 2009 American Control Conference*, pages 610–615, St. Louis, MO, June 2009.

[22] S. Lau and W. Naeem. Dynamic tensegrity based cooperative control of uninhabited vehicles. In *Intelligent Computing for Sustainable Energy and Environment*, pages 486–495. Springer, 2013.

[23] B. Nabet and N. E. Leonard. Tensegrity models and shape control of vehicle formations. *arXiv: 0902.3710*, 2009.

[24] R. Connelly. Tensegrity structures: why are they stable? In M. F. Thorpe and P. M. Duxbury, editors, *Rigidity theory and applications*, pages 47–54. Plenum Press, New York, 1999.

[25] A. Y. Alfakih and Viet-Hang Nguyen. On affine motions and universal rigidity of tensegrity frameworks. *Linear Algebra and its Applications*, 439(10):3134–3147, 2013.

[26] A. Pothen. *Sparse null bases and marriage theorems*. PhD thesis, Cornell University, 1984.

[27] L. Gottlieb and T. Neylon. Matrix sparsification and the sparse null space problem. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques*, pages 205–218. Springer, 2010.

[28] John R. Gilbert and Michael T. Heath. Computing a sparse basis for the null space. *SIAM Journal on Algebraic Discrete Methods*, 8(3):446–459, 1987.

[29] B. Nabat. *Dynamics and Control in Natural and Engineered Multi-Agent Systems*. PhD thesis, Princeton University, 2009.

[30] W. Ren and R. Beard. *Distributed consensus in multi-vehicle cooperative control: theory and applications*. Springer-Verlag, 2008.