Distributed formation stabilization for mobile agents using virtual tensegrity structures
Yang, Qingkai; Cao, Ming; Fang, Hao; Chen, Jie

Published in:
Proceedings of the 34th Chinese Control Conference (CCC 2015)

IMPORTANT NOTE: You are advised to consult the publisher's version (publisher's PDF) if you wish to cite from it. Please check the document version below.

Document Version
Publisher's PDF, also known as Version of record

Publication date:
2015

Link to publication in University of Groningen/UMCG research database

Citation for published version (APA):

Copyright
Other than for strictly personal use, it is not permitted to download or to forward/distribute the text or part of it without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license (like Creative Commons).

The publication may also be distributed here under the terms of Article 25fa of the Dutch Copyright Act, indicated by the “Taverne” license. More information can be found on the University of Groningen website: https://www.rug.nl/library/open-access/self-archiving-pure/taverne-amendment.

Take-down policy
If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

Downloaded from the University of Groningen/UMCG research database (Pure): http://www.rug.nl/research/portal. For technical reasons the number of authors shown on this cover page is limited to 10 maximum.

Download date: 17-09-2023
Distributed formation stabilization for mobile agents using virtual tensegrity structures

YANG Qingkai¹,², CAO Ming², FANG Hao¹,³, CHEN Jie¹,³, HUANG Jie¹,²
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. inextensible cables and incompressible struts). Finally, under the interaction constraints, 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’ attention, 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

1A configuration is in general position if no k points lie in a (k − 1) dimensional affine space for 1 ≤ k ≤ 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

\[
M_i(q_i)x + C_i(q_i, \dot{q}_i)\dot{x} + \tau_i, \quad i = 1, \ldots, n
\]  

(1)

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

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

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

(2)

where \( Y_i(q_i, q_i, y, x) \) is the known regressor and \( \Theta_i \) is a constant parameter vector but unknown.

The neighboring relationships between the robots are defined by an undirected graph \( G \) with the vertex set \( V = \{1, 2, \ldots, n\} \) and the edge set \( E \subseteq V \times V \) where there is an edge \((i, j)\) if and only if robots \( i \) and \( j \) are neighbors of each other. We use \( N_i \) to denote the set of neighbors of robot \( i \). \( G \) is embedded in \( \mathbb{R}^m \) when \( q = (q_1, q_2, \ldots, q_n) \) is realized and the pair \((G, q)\) is called a framework. Two frameworks \((G, q)\) and \((G, p)\) 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 \((i, j)\) is \( E \). 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 \( G \), and then for each robot \( i = 1, \ldots, n \), design distributed control laws \( \tau_i(q_i - q_i^*, \dot{q}_i - \dot{q}_i^*) \), \( j \in N_i \), such that the robots' positions are driven to the target set

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

Obviously, to make the control less complicated and scalable with \( n \), \( G \) 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 \( T(G) \) is obtained by embedding an undirected graph \( G \) in a Euclidean space and replacing each edge of \( G \) by an inextensible cables or incompressible struts, or inextensible and incompressible bars. Together all the cables, struts and bars are called the members of \( T \) and the embedded vertices of \( G \) are called the nodes of \( T \). So the same graph \( G \) may lead to different tensegrity frameworks when \( G \)'s edges are realized into different combinations of cables, struts, and bars.

We use the labels in the vertex set \( V \) of \( G \) for the nodes of \( T \). For each member \((i, j)\) of \( T \), we assign a scalar \( \omega_{ij} = \omega_{ji}, \) and use \( \omega \in \mathbb{R}^{|E|} \), where \(|E|\) is the number of members of \( T \), to denote the concatenated vector \( \omega = (\omega_{i1}, \omega_{i2}, \ldots, \omega_{in}, \omega_{j1}, \omega_{j2}, \ldots, \omega_{jm})^T \). Then \( \omega \) is called a stress of \( T \); if further, each \( \omega_{ij} \) satisfies \( \omega_{ij} \geq 0 \) whenever \((i, j)\) is a cable and \( \omega_{ij} \leq 0 \) whenever \((i, j)\) is a strut, then \( \omega \) is said to be a proper stress.

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

\[
\sum_{j \in N_i} \omega_{ij}(q^*_j - q^*_i) = 0, \quad i = 1, \ldots, n. \quad (4)
\]

Given \( \omega \), the associated stress matrix \( \Omega \) 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 \( T \) with the desired configuration \( q^* \), we are interested in its associated configurations \( p \) that satisfy the following tensegrity constraints

\[
\begin{cases}
|p_i - p_j| \leq |q^*_i - q^*_j|, & \text{when } (i, j) \text{ is a cable}, \\
|p_i - p_j| \geq |q^*_i - q^*_j|, & \text{when } (i, j) \text{ is a strut} \\
|p_i - p_j| = |q^*_i - q^*_j|, & \text{when } (i, j) \text{ is a bar}.
\end{cases} \quad (5)
\]

Such constraints can be naturally used to define the “rigidity” property of \( T \). We say that the tensegrity \( T \) 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 \( p \in \mathbb{R}^{mn} \), then we say \( T \) is globally rigid; and even more strongly, if this congruent relationship still holds for all \( q \) living in any higher-dimensional spaces than \( \mathbb{R}^{mn} \), we say \( T \) 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 \((G, p)\) be an \( r \)-dimensional tensegrity framework on \( n \) vertices in \( \mathbb{R}^r \), for some \( r \leq n - 2 \). Then \((G, p)\) is universally rigid if the following two conditions hold.

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

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 struts but no bars as their members. Since the row sums of \( \Omega \) are all zero, \( 1_n \) always lives in null(\( \Omega \)). One can further check that the columns of \( (q^*)^T \) are in null(\( \Omega \)) as well. In fact, the column span of \( N \) \( \cong (q^*)^T \) always belongs to null(\( \Omega \)). Therefore,

\[
\Omega N = 0_{n \times (m+1)}.
\] (6)

Given \( q^* \), to assign a virtual tensegrity \( T \) to the robotic team is equivalent to use \( N \) to determine the matrix \( \Omega \) since once \( \Omega \) is determined, all the needed cables and struts together with their stresses are determined as well. Obviously, such \( \Omega \)'s are in general not unique and naturally we want to obtain sparse \( \Omega \) 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 an \( m \times n \) 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

\[
N^T \Omega^T = N^T \Omega = 0_{(m+1) \times n}.
\] (7)

From Lemma 1, we need the sparse matrix \( \Omega \) to be positive semi-definite and rank(\( \Omega \)) = \( n - d - 1 \). However, since \( \Omega \) 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

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

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

\[
N^T D D^T = 0_{(m+1) \times (n-m-1)}\quad D^T = 0_{(m+1) \times n}
\] (9)

and hence the matrix \( DD^T \) can serve as the stress matrix \( \Omega \). So the construction of a sparse matrix \( \Omega \) is equivalent to the design of such a sparse \( D \). In addition, we make an even stronger requirement that \( \Omega \) 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 \( \Omega \), which is inspired by the classical “turning buck” method for computing the sparse null space basis [28].

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

Step 2: Now we construct \( D \) by finding a sparse basis for null(\( N \)). We first find the smallest \( k_1 > 0 \) such that \( \widetilde{N}'s \) columns with the indices \( m+2, m+1, \ldots, m+2-k_1 \) are linearly dependent. We record that the \((m+2-k_1)th, \ldots, (m+2)th\) 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 \( N \) 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 \( D \).

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

\[
F_{j \rightarrow i} = k_{ij}(r_{ij} - l_{ij}) = -F_{i \rightarrow j}
\] (11)
where \( r_{ij} \) is the relative displacement between agent \( i \) and agent \( j \), which is defined as
\[
r_{ij} = q_i - q_j \quad (12)
\]
In order to coincide with the stress for cables (struts), we assign the rest length for cables (struts) to be \( \beta_j^* = 1 \) the prescribed displacement between the agents, namely,
\[
l_{ij} = \begin{cases} 
\beta_{ij} r_{ij}^* & \text{if } (i, j) \in \mathcal{E}_C \\
\beta_{ij}^* r_{ij}^* & \text{if } (i, j) \in \mathcal{E}_S 
\end{cases} \quad (13)
\]
where \( \beta_{ij}^* \in (0, 1) \), \( \beta_{ij}^* \in (1, +\infty) \) 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
\[
k_{ij} = \begin{cases} 
\frac{\Omega_{ij}}{1 - \beta_{ij}^*} & \text{if } \Omega_{ij} < 0 \\
\frac{\Omega_{ij}}{1 - \beta_{ij}^*} & \text{if } \Omega_{ij} > 0 
\end{cases} \quad (14)
\]
In the context of virtual springs, the potential energy \( P(q) \) caused by the disagreement between \( r_{ij} \) and \( r_{ij}^* \) is defined as
\[
P(q) = \frac{1}{2} \sum_{(i, j) \in \mathcal{E}} k_{ij} \|r_{ij} - r_{ij}^*\|^2 \quad (15)
\]
It is worth mentioning that the virtual cables (struts) are in tension (compression) at the equilibrium configuration \( q^* \in \mathbb{R}^m \) due to \( l_{ij} = \beta_{ij} r_{ij}^* \). Therefore, the formation achieved based on the virtual tensile 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
\[
s_i = \dot{q}_i + g_i(q) \quad (16)
\]
where
\[
g_i(q) = \frac{\partial P}{\partial q_i} = -\sum_{j \in \mathcal{N}_i} k_{ij}(r_{ij} - r_{ij}^*) \quad (17)
\]
In view of (1) and (16), one has
\[
M_i \ddot{s}_i + C_i \dot{s}_i = M_i(s_i + \dot{q}_i + \dot{g}_i(q)) + C_i(\dot{q}_i + g_i(q)) \]
\[
= \tau_i + M_i g_i(q) + C_i g_i(q) \]
\[
= \tau_i + Y_i(q_i, \dot{q}_i, g_i, \dot{g}_i) \dot{q} \quad (18)
\]
The distributed control input \( \tau_i \) is designed as
\[
\tau_i = -k_p s_i - g_i(q) - Y_i(q_i, \dot{q}_i, g_i, \dot{g}_i) \dot{q} \quad (19)
\]
where \( k_p \) is a positive scalar, and \( \dot{q} \) is the estimation of \( q \), which is updated according to
\[
\dot{q} = \Gamma_i Y_i(q_i, \dot{q}_i, g_i, \dot{g}_i)^T s_i \quad (20)
\]
where \( \Gamma_i \) 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
\[
V = \frac{1}{2} \sum_{i=1}^{n} (s_i^T M_i s_i + \dot{\Theta}_i^T \Gamma_i^{-1} \dot{\Theta}_i + P(q)) \quad (21)
\]
where \( \dot{\Theta}_i = \dot{\Theta}_i - \Theta_i \) is the estimation error, thus \( \dot{\Theta}_i = \dot{\Theta}_i \).

Taking the time derivative of \( V \), we have
\[
\dot{V} = \frac{1}{2} \sum_{i=1}^{n} (s_i^T M_i s_i + 2s_i^T M_i \dot{s}_i) + \sum_{i=1}^{n} (\dot{\Theta}_i^T \Gamma_i^{-1} \dot{\Theta}_i + \dot{g}_i(q)^T \dot{g}_i(q)) \quad (22)
\]
Substituting (18)-(19) into \( \dot{V} \) yields
\[
\dot{V} = \sum_{i=1}^{n} \left[ -k_p s_i^T s_i - s_i^T g_i(q) + g_i(q)^T (s_i - g_i(q)) \right] \]
\[
= -\sum_{i=1}^{n} (k_p s_i^T s_i + g_i(q)^T g_i(q)) \quad (23)
\]

Hence, it can be concluded that \( \dot{\Theta}_i \in \mathcal{L}_\infty, s_i \in \mathcal{L}_\infty \cap \mathcal{L}_2, g_i(q) \in \mathcal{L}_\infty \) and therefore \( q_i \in \mathcal{L}_\infty \) from (16). Then, it follows that \( g_i(q) \in \mathcal{L}_\infty \), 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 \( s_i \in \mathcal{L}_\infty \) from (18), which, together with \( \dot{q}_i(q) \in \mathcal{L}_\infty \), implies \( V(s_i, \dot{s}_i, g_i, \dot{g}_i) \) is bounded. It can be concluded from Barbalat’s Lemma that \( \dot{V} \to 0 \), as \( t \to \infty \). Therefore, for each agent \( i \),
\[
\lim_{t \to \infty} s_i = 0 \quad (24a)
\]
\[
\lim_{t \to \infty} g_i(q) = 0 \quad (24b)
\]

In view of the definition of variable \( s_i \) in (16), it follows
\[
\lim_{t \to \infty} \dot{q}_i(t) = 0 \quad (25)
\]

The equation (24b) can be grouped as
\[
- \left( \mathcal{K} \otimes I_m \right) q_n = 0 \quad (26)
\]
where \( q_n = [(q_1 - q_1^*)^T, (q_2 - q_2^*)^T, \ldots, (q_n - q_n^*)^T]^T \) and the “spring constant matrix” \( \mathcal{K} \in \mathbb{R}^{n \times n} \) is defined in the same way as the standard Laplacian matrix, i.e.,
\[
\mathcal{K}_{ii} = \sum_{j \in \mathcal{N}_i} k_{ij}, \quad \mathcal{K}_{ij} = -k_{ij}, \ i \neq j \quad (27)
\]

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 the framework \( (G, q^*) \) is globally rigid with well-designed stresses by Lemma 1. Hence, the agents globally converge to the target set \( T \) in (3), namely, the desired formation is achieved based on the proposed virtual tensile structure.
**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 \( \Omega \) the virtual tensegrity structure is shown in Fig. 2.

Matlab. \( d_{ij} = \|r_{ij}\| \), and \( d^d_{ij} \) 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


