Distributed Formation Stabilization Using Relative Position Measurements in Local Coordinates

Miguel Aranda, Member, IEEE, Gonzalo López-Nicolás, Senior Member, IEEE, Carlos Sagüés, Senior Member, IEEE, and Michael M. Zavlanos, Member, IEEE

Abstract—In this paper, we present a novel distributed method to stabilize a set of agents moving in a two dimensional environment to a desired rigid formation. In our approach, each agent computes its control input using the relative positions of a set of formation neighbors but, contrary to most existing works, this information is expressed in the agent’s own independent local coordinate frame, without requiring any common reference. The controller is based on the minimization of a Lyapunov function that includes locally computed rotation matrices, which are required due to the absence of a common orientation. Our contribution is that the proposed distributed coordinate-free method achieves global stabilization to a rigid formation with the agents using only partial information of the team, does not require any leader units, and is applicable to both single-integrator or unicycle agents. To guarantee global stability, we require that the network induced by the agent interactions belongs to a certain class of undirected rigid graphs in two dimensions, which we explicitly characterize. The performance of the proposed method is illustrated with numerical simulations.

Index Terms—Distributed control, multi-agent systems, formation stabilization, autonomous mobile robots.

I. INTRODUCTION

Teams of mobile agents capable of autonomous perception, localization, and navigation, can be used to address diverse application scenarios, such as environment surveillance, mapping, exploration, or search and rescue missions, among others. In this paper, we study multiagent formations, which are fundamental to the emergence of many interesting group behaviors. We address, in particular, the problem of distributed formation stabilization [1]. Our goal is to ensure that the positions of a set of mobile agents moving in a two-dimensional space form a desired rigid shape, defined up to translation and rotation.

A formation is often specified by a set of absolute positions attained by the agents in a team [2]–[6]. In this case, controlling the formation requires the use of global positioning sensors (e.g., GPS) on board the agents, or an external localization system. Nevertheless, the availability of such localization systems is often difficult to ensure as, e.g., in the case of agents that operate indoors, where GPS signal is poor. These limitations can be overcome by using relative measurements, as is done in the so-called relative position-based formation stabilization methods [7]–[13]. Still, though, these methods need the agents to have a common sense of orientation. Position-based approaches use linear consensus-based control laws, and ensure global stabilization if the graph that models the interactions between agents (the formation graph) is connected. Some of the cited works that use relative measurements assume the common orientation reference is available via sensing, whereas others such as [10], [13] need the agents to agree upon, and subsequently maintain, this required global orientation in a decentralized manner [14].

Relaxing this need for a common sense of orientation is important as it can enhance flexibility of the system by, e.g., permitting operation in GPS-denied environments. Moreover, it can reduce the dependence on complex and expensive sensing and increase the agents’ autonomy by enabling them to rely only on low-cost, on-board sensors that do not provide any absolute position or orientation information. In this paper, unlike in all the works cited above, we assume such a coordinate-free scenario, where the agents plan their motion relying only on their own independent local coordinates. Similar frameworks that do not require global references have been considered recently in relevant literature on formation stabilization. Next, we review these methods and discuss the differences with our proposed approach. A recent survey of formation controllers focusing on their information requirements and the topology of agent interactions employed can be found in [15].

The method in [16] uses relative pose (i.e., position and orientation) information and considers mobile platforms whose positions and orientations are controlled in a decoupled fashion. Then, for general connected undirected formation graphs, the formation is globally stabilized via two separate consensus-based control laws. In contrast, we assume the agents have single-integrator or unicycle kinematics, and we employ only relative position information. Works that consider relative information commonly specify the formation via interagent distances only, to avoid the difficulty of dealing with inconsistent (i.e., expressed in unaligned frames) position coordinates. This is the approach followed in distance-based formation stabilization methods [17]–[22]. When applied to rigid-shape stabilization problems for arbitrary numbers of agents, these strategies require the formation graph to be rigid [23] and provide only local stability guarantees. Global stabilization to a rigid shape poses significant challenges for these approaches;
indeed, as shown in [18], [24], distance-based global formation stabilization is infeasible using gradient descent controllers, which are the most common in the literature. Distance-based schemes require exactly the same information as the method we propose, as knowledge of the directions to the neighboring agents is needed to compute the motion vectors. However, compared to these methods, our approach is globally stable.

Global stabilization to a rigid formation shape using only relative position information expressed in independent local coordinate frames, as in our method, has been achieved by relying on leader agents. Specifically, the distance-based method in [20] achieves global stability for a triangulated graph structure with two leaders. In [25], local relative positions (not just distances) are used in a linear control law based on the complex Laplacian matrix of the formation graph. For a 2-rooted undirected graph, the method globally stabilizes a rigid formation, using two leader agents to fix its scale. Unlike [20], [25], our approach is leaderless. This provides advantages in terms of flexibility and robustness, since it prevents the team’s performance from relying heavily on the leaders which, in addition, operate without feedback from the other agents. The work [26] presents a distance-based modified gradient controller where all the agents share a common clock (contrary to our approach) and each adds to its control input an adaptive time-parameterized perturbation. Then, global rigid-shape stabilization is obtained if the formation graph is rigid and no two agents are co-located initially. In the context of formation control, unicycle-type agents are important from a practical perspective [3], [5]–[7], [9], [18], [27]–[30] but introduce additional challenges due to the nonholonomic constraints that restrict their executable motions. To the best of our knowledge, none of the existing distributed, globally convergent, coordinate-free rigid formation controllers [16], [20], [25], [26] considers unicycle-type agents. To the contrary, our method is directly applicable for this kinematics. We also note that some formation control schemes use camera-equipped external units to compute the commands [29]–[31]. In particular, [30] employs multiple partial information-based least-squares image transformations to create the formation. However, as they rely on centralizing external units, this group of methods are neither distributed nor coordinate-free.

The method we propose in this paper stabilizes a group of agents to a rigid shape, using the relative positions of each agent’s formation graph neighbors, expressed in local coordinate frames. We capture this control objective by the minimizer of a Lyapunov function that includes this relative position information in full (contrary to distance-based methods), and propose a gradient descent controller that allows us to globally achieve this minimum configuration. Specifically, the proposed Lyapunov function is the sum of cost functions associated with maximal cliques, i.e., groups of mutually adjacent agents, in the formation graph. Due to the lack of a shared orientation reference, our Lyapunov function necessarily contains rotation matrices acting on the local relative position vectors, which makes the system dynamics nonlinear. The key idea that enables our approach is to define these rotations as minimizers of the cost functions associated with every maximal clique, and then substitute these expressions in the proposed gradient descent controllers, for which we show that they ensure global stability both for single-integrator and unicycle-type agents. Global stability guarantees require an interaction topology modeled by a class of undirected rigid graphs in two dimensions, which we explicitly characterize.

For this class of graphs, in our distributed method each agent computes locally its motion commands, maintains interactions (via sensing or communications) only with its formation graph neighbors, and requires only partial information (specifically, the relative positions of its formation graph neighbors) of the team.

Let us summarize our contribution: to the best of our knowledge, this paper proposes for the first time a method for distributed rigid-shape formation stabilization that uses only locally expressed relative position measurements (i.e., without any common reference), requires each agent to know only partial information of the group, and is globally convergent, leaderless, and applicable for unicycle kinematics. Existing works enjoy a subset of these properties, but not all of them, as discussed in the literature review presented above. Furthermore, an important aspect of our contribution is that we provide a characterization of specific topological conditions for which global stability is ensured. Our related works [32], [33] address coordinate-free formation control and, as we do here, employ rotation matrices computed locally by the agents. However, contrary to the work we present, in both of these prior methods the agents employ global information. [33] uses multi-hop communication that is subject to time delays to propagate the necessary global information to all the agents in the network in a distributed way, while [32] addresses a target enclosing task with agents that move in 3D space.

The contents of the paper are structured as follows. Section II introduces the problem we address and provides necessary background regarding several graph-theoretic concepts. In Section III, we describe the proposed multiagent control strategy, in which the motion commands are locally computed by each of the agents. Section IV presents the stability analysis of our method. We discuss in Section V the class of formation graphs for which the controller is ensured to be stable. Section VI describes results from simulations carried out to evaluate our approach. Finally, the conclusion of the paper and directions for future work are presented in Section VII.

II. PROBLEM FORMULATION AND BACKGROUND

Consider a group of agents in \( \mathbb{R}^2 \). Let us denote, in an arbitrary global reference frame, the position of agent \( i, i = 1, ..., N \), as \( \mathbf{q}_i = [q_{ix}, q_{iy}]^T \in \mathbb{R}^2 \) and its orientation as \( \phi_i \in \mathbb{R} \). We assume each agent obeys unicycle kinematics, as follows:

\[
q_i^x = -v_i \sin \phi_i, \quad q_i^y = v_i \cos \phi_i, \quad \dot{\phi}_i = -\omega_i,
\]

where \( v_i \in \mathbb{R} \) is its linear velocity input and \( \omega_i \in \mathbb{R} \) is its angular velocity input. We will additionally consider the single-integrator model, where the orientation of each agent is not relevant and its dynamics is determined by a velocity input \( \mathbf{u}_i \in \mathbb{R}^2 \):

\[
\dot{\mathbf{q}}_i = \mathbf{u}_i.
\]

Let us denote \( \mathbf{q} \equiv [q_1^T, ..., q_N^T]^T \) and \( \mathbf{q}^* \equiv [q_{i_1}^*, ..., q_{i_k}^*]^T \) the position of the \( k \) agents with the given formation, \( \mathbf{v} \equiv [v_1, ..., v_N]^T \) the linear velocity of the \( N \) agents, \( \mathbf{w} \equiv [\omega_1, ..., \omega_N]^T \) the angular velocity of the \( N \) agents, and \( \mathbf{u} \equiv [u_1, ..., u_N]^T \) the control input of the \( N \) agents.

For the agents with the given formation, we define the position error as:

\[
\mathbf{e}_i = \mathbf{q}_i - \mathbf{q}_i^*.
\]

The position error is the difference between the position of agent \( i \) and its desired position in the formation. The goal is to design a control law that drives the agents to a desired formation while maintaining their relative positions and orientations.

In order to achieve this, we define the following cost functions:

\[
C_i = \frac{1}{2} (\mathbf{e}_i^T \mathbf{e}_i) + \frac{1}{2} (\mathbf{e}_i - \mathbf{e}_j)^T \mathbf{L} (\mathbf{e}_i - \mathbf{e}_j),
\]

where \( \mathbf{L} \) is the Laplacian matrix of the formation graph, and \( \mathbf{e}_i - \mathbf{e}_j \) represents the relative position of agent \( i \) with respect to agent \( j \).

The Lyapunov function is then defined as:

\[
\gamma = \sum_{i=1}^{N} C_i.
\]

The objective is to design a control law that minimizes \( \gamma \) while ensuring that the agents converge to the desired formation.

To achieve this, we propose the following control law:

\[
\mathbf{u}_i = \mathbf{K} \mathbf{e}_i + \mathbf{K}_p \mathbf{e}_i + \mathbf{K}_d (\mathbf{e}_i - \mathbf{e}_{i-1}),
\]

where \( \mathbf{K} \), \( \mathbf{K}_p \), and \( \mathbf{K}_d \) are gain matrices.

The control law consists of three terms:

1. The first term, \( \mathbf{K} \mathbf{e}_i \), corresponds to the resolution of \( \gamma \).
2. The second term, \( \mathbf{K}_p \mathbf{e}_i \), corresponds to the momentum of the agents.
3. The third term, \( \mathbf{K}_d (\mathbf{e}_i - \mathbf{e}_{i-1}) \), corresponds to the directional error.

By designing the gain matrices \( \mathbf{K} \), \( \mathbf{K}_p \), and \( \mathbf{K}_d \) appropriately, we can ensure that the agents converge to the desired formation and maintain their relative positions and orientations.

The proposed control law guarantees global stability of the system, and it is easily computable by each agent using only local information.

Finally, the conclusion of the paper and directions for future work are presented in Section VII.

VII. CONCLUSION

In this paper, we have presented a novel control strategy for the global formation stabilization of a group of agents with unicycle-type kinematics. The proposed method is leaderless, coordinate-free, and applicable for agents with unicycle kinematics. The method ensures global stability, and it is easily distributed, as each agent only needs to know the relative positions and orientations of its neighbors.

The proposed method can be applied in various scenarios, such as robotic swarms, autonomous vehicles, and unmanned aerial vehicles. The method is particularly useful in situations where centralized control is not feasible, such as in large-scale systems or systems where communication is limited.

Future work could include the extension of the proposed method to more complex formation shapes, the incorporation of additional constraints, and the analysis of the method in more general scenarios, such as higher-dimensional spaces or systems with non-uniform velocities.
We define a desired configuration, or formation shape, by a certain, fixed, reference layout of the positions of the $N$ agents in their configuration space. The way in which we encode the desired configuration is through a set of interagent relative position vectors. To model the interactions between agents, we define a static undirected formation graph $G_f = (V, E)$, as is typical in related work on formation control, e.g., [20], [25], [26]. Each node in $V$ is associated with an agent, and we assume that each agent can obtain, using its sensing or communication capabilities, an estimate of the relative positions of its fixed set of neighbors in $G_f$.

Then, for every neighbor $j$ of agent $i$, we denote as $c_{ji} \in \mathbb{R}^2$ the vector from $i$ to $j$ in the reference layout of the agents that defines the desired configuration. The agents are not interchangeable, i.e., each of them has a fixed place in the target formation. We then consider that the $N$ agents are in the desired configuration if the reference layout has been achieved, up to an arbitrary rotation and translation, i.e., if it holds that:

$$q_{ji} = Rc_{ji}, \quad \forall i, j = 1, \ldots, N,$$  

where we define $q_{ji} = q_i - q_j$, and $R \in \text{SO}(2)$ is an arbitrary rotation matrix. Thus, the problem that we set out to solve in this paper is specified as follows:

**Problem 1.** Given an initial configuration in which the agents are in arbitrary positions, find a control strategy that stabilizes them in a set of final positions such that the group is in the desired configuration.

### A. Graph theory

Next, we discuss a series of definitions relevant to undirected graphs that are used throughout the paper. A **clique** in a graph $G$ is a complete subgraph, i.e., a subset of its nodes and edges such that every two nodes are adjacent. An $n$-node clique is a clique containing $n$ nodes. A **maximal clique** is one that cannot be augmented by incorporating one more node. The intersection of two cliques is given by the sets of nodes and edges they share. The $p$-clique graph, with $p \geq 1$, of $G$, denoted as $C_p(G)$, is a graph whose nodes are the maximal cliques of $G$, and where two nodes are adjacent if the intersection of their associated cliques contains at least $p$ nodes [34]. A graph is called a tree if any two of its nodes are connected by exactly one sequence of edges. A leaf in a tree graph is a node of degree one. An induced subgraph of $G$ is a graph that includes a subset of its nodes and all those edges of $G$ that join two nodes in the subset.

### III. CONTROL STRATEGY

Let us assume there are $M$ maximal cliques in $G_f$. For $m = 1, \ldots, M$, we denote as $I_m$ the set of indices of the nodes that form the $m$-th clique, and $N_m = \text{card}(I_m)$. We interchangeably refer in the paper to the nodes of $G_f$ as nodes or agents. We define, in an arbitrary global coordinate frame, the following cost function for each maximal clique:

$$\gamma_m = \frac{1}{2N_m} \sum_{j \in I_m} \| \sum_{k \in I_m} q_{jk} - R_m c_{jk} \|^2,$$  

where $R_m \in \text{SO}(2)$ is a rotation matrix whose value, equal for all the agents in the clique, is discussed in the following section. Note that, for simplicity of the notation, we include in (4) the null terms occurring when $j = k$. Now, observe that if $\gamma_m = 0$, we can write, considering in (4) the addends associated with two given agents $j = i_1$ and $j = i_2$:

$$\sum_{k \in I_m} q_{i_1k} - R_m c_{i_1k} = 0, \quad \sum_{k \in I_m} q_{i_2k} - R_m c_{i_2k} = 0.$$  

Subtracting the two equations, we have that $q_{i_1i_2} = R_m c_{i_1i_2}$, which holds for every pair $i_1, i_2$ in $I_m$. Hence, if $\gamma_m = 0$, the subset of agents in the $m$-th clique are in the desired configuration with one another (we will refer to this as a sub-formation). We can then see that $\gamma_m$ encodes how distant the agents are from reaching the $m$-th sub-formation. We define the global cost function for our system as follows:

$$\gamma = \sum_{m=1,\ldots,M} \gamma_m,$$  

so that if $\gamma$ vanishes, all the sub-formation are attained. Note that every node and every edge of $G_f$ are part of one or multiple maximal cliques. Thus, if $\{i, j\} \in E$, the relative vectors between $i$ and $j$ contribute to at least one $\gamma_m$ (4). This means that, by encompassing all the $M$ maximal cliques, the global function $\gamma$ captures all the edges in $G_f$. An illustration of the structure of maximal cliques that is the basis of our controller is provided in Fig. 1.

### A. Rotation matrices

We set out to drive $\gamma$ to zero, which implies doing so for every $\gamma_m$. Accordingly, the rotation matrix in each function $\gamma_m$ (4) is chosen so as to minimize it, as shown next. Let us note that the analysis in this section is analogous to finding the solution to the orthogonal Procrustes problem [35].

![An arbitrary formation graph $G_f$ with $N = 17$ nodes and $M = 10$ maximal cliques, denoted as $K_m$, $m = 1, \ldots, M$. The size of the maximal cliques ranges from two to five nodes. For instance, $K_1$ contains two agents, $K_3$ contains three agents, $K_4$ contains four, and $K_9$ contains five agents. Trivial single-node maximal cliques, which would represent isolated nodes of $G_f$, are not contemplated. Our formation controller is based on the minimization of a global cost function $\gamma$ that is the aggregate of partial functions $\gamma_m$ associated with the maximal cliques. Each agent operates on the set of maximal cliques it belongs to, e.g., the motion of agent $i$ pursues the minimization of the sum of $\gamma_3, \gamma_4$, and $\gamma_9$, for which it uses the knowledge of the locally expressed relative positions of its neighbors in $G_f$. To ensure stabilization to a rigid formation, $G_f$ must satisfy certain rigidity-related conditions, as explained throughout the paper.](image)
We define $R_m$ as a rotation by an angle $\alpha_m$, i.e.:

$$R_m = \begin{bmatrix} \cos \alpha_m & -\sin \alpha_m \\ \sin \alpha_m & \cos \alpha_m \end{bmatrix}.$$  \hfill (7)

Let us express $\gamma_m$ in terms of $\alpha_m$ and the components of the relative position vectors, $q_{jk} = [q_{jk}^x, q_{jk}^y]^T$, $c_{jk} = [c_{jk}^x, c_{jk}^y]^T$:

$$\gamma_m = \frac{1}{2N_m} \sum_{j \in I_m} \left[ (\sum_{k \in I_m} q_{jk}^x - c_{jk}^x \cos \alpha_m + c_{jk}^y \sin \alpha_m)^2 + (\sum_{k \in I_m} q_{jk}^y - c_{jk}^y \sin \alpha_m - c_{jk}^x \cos \alpha_m)^2 \right].$$  \hfill (8)

Let us introduce the notation: $S_{qj} = [S_{qj}^x, S_{qj}^y]^T = \sum_{k \in I_m} q_{jk}$ and $S_{cj} = [S_{cj}^x, S_{cj}^y]^T = \sum_{k \in I_m} c_{jk}$. To minimize $\gamma_m$ with respect to $\alpha_m$, we solve $\frac{\partial \gamma_m}{\partial \alpha_m} = 0$. After manipulation, this derivative is:

$$\frac{\partial \gamma_m}{\partial \alpha_m} = \frac{1}{N_m} \left[ \sin \alpha_m \sum_{j \in I_m} (S_{qj}^x S_{cj}^x + S_{qj}^y S_{cj}^y) - \cos \alpha_m \sum_{j \in I_m} (-S_{qj}^x S_{cj}^y + S_{qj}^y S_{cj}^x) \right].$$  \hfill (9)

Then, the condition $\frac{\partial \gamma_m}{\partial \alpha_m} = 0$ is expressed as:

$$\sin \alpha_m \sum_{j \in I_m} S_{qj}^x S_{cj} - \cos \alpha_m \sum_{j \in I_m} S_{qj}^y S_{cj} = 0,$$  \hfill (10)

where the superscript $\perp$ denotes a rotation of a vector by $\pi/2$ radians, as follows: $S_{qj}^\perp = [(1, 0)^T, (-1, 0)^T][S_{cj}]$. Solving (10) with respect to the rotation angle $\alpha_m$, we get:

$$\alpha_m = \arctan \frac{\sum_{j \in I_m} S_{qj}^x S_{cj}}{\sum_{j \in I_m} S_{qj}^y S_{cj}}.$$  \hfill (11)

Observe from (11) that there are two possible solutions for $\alpha_m$, separated by $\pi$ radians. In order to select the correct one, we compute the second order derivative from (9):

$$\frac{\partial^2 \gamma_m}{\partial \alpha_m^2} = \frac{1}{N_m} \left[ \cos \alpha_m \sum_{j \in I_m} S_{qj}^x S_{cj} + \sin \alpha_m \sum_{j \in I_m} S_{qj}^y S_{cj} \right].$$  \hfill (12)

By considering together (10) and (12), it can be readily seen that one of the solutions for (11) minimizes $\gamma_m$, while the other maximizes the function. The solution that is a minimum satisfies the condition $\frac{\partial^2 \gamma_m}{\partial \alpha_m^2} > 0$. If we isolate the term $\cos \alpha_m$ in (10) and then substitute it in (12), we easily get that this condition holds when $\sin(\alpha_m) / \sum_{j \in I_m} S_{qj}^x S_{cj} > 0$, i.e., $\sin(\alpha_m)$ must have the same sign as the numerator in the arctan function in (11). This implies that, among the two possible values of $\alpha_m$, the one that minimizes $\gamma_m$, i.e., the value used in our controller, is given by:

$$\alpha_m = \arctan 2 \left( \sum_{j \in I_m} S_{qj}^x S_{cj} \right) \sum_{j \in I_m} S_{qj}^y S_{cj},$$  \hfill (13)

where the $\arctan 2$ function returns the solution of (11) for which $\alpha_m$ is in the quadrant that corresponds to the signs of the two input arguments. Note that the case $\arctan 2(0, 0)$, for which $\alpha_m$ is not defined, is theoretically possible in (13) for degenerate configurations of the agents’ positions where $\gamma_m$ is constant for all $\alpha_m$, see (10). In general terms, the singular or degenerate cases are linked to the desired geometry and not to our control strategy. Multiple agents occupying the same position is another particular example of a degenerate arrangement. All these possible configurations are measure zero, i.e., they will never occur in practice and, therefore, we do not consider them in our analysis.

B. Control law

Our controller is based on each agent $i$ following the negative gradient of the cost function $\gamma$ with respect to $q_i$. Let us look at one given clique, $m$, that contains agent $i$. We have:

$$\nabla_{q_i} \gamma_m = \frac{\partial \gamma_m}{\partial q_i} = \frac{\partial q_i}{\partial q_i} + \frac{\partial q_i}{\partial q_i} = \frac{\partial q_i}{\partial q_i},$$  \hfill (14)

given that, as discussed in Section III-A, $\frac{\partial \gamma_m}{\partial q_i} = 0$. Thus, we focus next on the partial differentiation with respect to $q_i$. For clarity of the exposition, let us express $\gamma_m$ as a sum of components:

$$\gamma_m = \sum_{j \in I_m} \gamma_{mj}, \quad \gamma_{mj} = \frac{1}{2N_m} \left[ \sum_{k \in I_m} q_{jk} - R_m c_{jk} \right]^2.$$  \hfill (15)

For the component corresponding to $j = i$, we have:

$$\frac{\partial \gamma_{mj}}{\partial q_i} = \frac{\partial \gamma_{mj}}{\partial q_i} = \frac{1}{N_m} \sum_{k \in I_m} (q_{ki} - q_{ki} - R_m c_{ik})(N_m - 1)$$

whereas each of the components in (15) such that $j \neq i$ gives:

$$\frac{\partial \gamma_{mj}}{\partial q_i} = \frac{1}{N_m} \sum_{k \in I_m} (q_{ki} - q_{ki} - R_m c_{ik}) - \frac{1}{N_m} \sum_{k \in I_m} (q_{ij} - R_m c_{ij}).$$  \hfill (17)

From (15), and substituting and grouping (16) and (17), we get:

$$\frac{\partial \gamma_m}{\partial q_i} = \frac{\partial q_i}{\partial q_i} + \sum_{j \in I_m} \frac{\partial q_i}{\partial q_i}$$

$$= \sum_{k \in I_m} (q_{ik} - R_m c_{ik}) - \frac{1}{N_m} \sum_{j \in I_m} \sum_{k \in I_m} (q_{jk} - R_m c_{jk}).$$  \hfill (18)

Observe now that:

$$\sum_{j \in I_m} \sum_{k \in I_m} q_{jk} = N_m \sum_{j \in I_m} q_j - \sum_{j \in I_m} q_k = \sum_{j \in I_m} \sum_{k \in I_m} c_{jk} = 0.$$  \hfill (19)

Substituting (19) in (18) and then renaming the index $k$ as $j$, for convenience, we finally get:

$$\frac{\partial \gamma_m}{\partial q_i} = \sum_{j \in I_m} q_{ij} - R_m c_{ij}.$$  \hfill (20)

Let us denote, for every agent $i$, the set of maximal cliques to which it belongs as $C_i$, $i = 1, ..., N$. Note that, clearly,
\[
\frac{\partial \gamma}{\partial \mathbf{q}_i} = 0 \text{ if } m \text{ is not in } C_i. \]
We can now differentiate the global cost function (6). Substituting (14) and (20), we have:
\[
\nabla_{\mathbf{q}_i} \gamma = \sum_{m=1,\ldots,M} \nabla_{\mathbf{q}_i} \gamma_m = \sum_{m=1,\ldots,M} \left[ \sum_{j \in I_m} \mathbf{q}_{ij} - \mathbf{R}_m \mathbf{c}_{ij} \right].
\]
(21)

Let us define the partial desired motion vector for agent \(i\) due to clique \(m\) as:
\[
\mathbf{d}_{im} = \sum_{j \in I_m} \mathbf{q}_{ij} - \mathbf{R}_m \mathbf{c}_{ji}.
\]
(22)

Negating the gradient in (21), we obtain what we will call the partial desired motion vector for agent \(i\) with unicycle kinematics:
\[
\mathbf{d}_{i} = -\nabla_{\mathbf{q}_i} \gamma = \sum_{m \in C_i} \left[ \sum_{j \in I_m} \mathbf{q}_{ij} - \mathbf{R}_m \mathbf{c}_{ji} \right] = \sum_{m \in C_i} \mathbf{d}_{im}.
\]
(23)

Considering single-integrator kinematics, we propose to define each agent’s control input directly as:
\[
\mathbf{u}_i = \dot{\mathbf{q}}_i = k_c \mathbf{d}_i,
\]
(24)
where \(k_c > 0\) is a control gain. If, instead, the agents have unicycle kinematics, we define \(\beta_i\) as the angular alignment error, measured in the interval \([-\pi, \pi]\), between agent \(i\)’s current heading and the direction of its desired motion vector (see Fig. 2, left). If \(\mathbf{d}_i = \mathbf{0}\), we define \(\beta_i = 0\). Then, we propose the following control law:
\[
\left\{ \begin{array}{ll}
\nu_i = \kappa_v ||\mathbf{d}_i||, & \text{if } |\beta_i| < \frac{\pi}{2} \\
0, & \text{if } |\beta_i| \geq \frac{\pi}{2}
\end{array} \right.
\]
(25)
where \(\kappa_v > 0\) and \(\kappa_w > 0\) are control gains. Observe that when \(|\beta_i| \geq \frac{\pi}{2}\), the agent only rotates in place and does not translate.

\(\text{C. Information requirements}\)

Let us specify the information that a given agent needs so as to compute its control input.

Note that the control laws above define a distributed system which relies on the use of only partial information of the multiagent team. For each agent \(i\), its control input is obtained using only the knowledge of \(\mathbf{d}_i\), which is obtained, (23), (13), from the relative position vectors corresponding to the agents belonging to the cliques in \(C_i\), i.e., the agents which are neighbors of \(i\) in \(G_f\). That is, defining \(N_i\) as \(i\)’s set of formation graph neighbors, agent \(i\) needs to know the measurements \(\mathbf{q}_{ji}, \forall j \in N_i\). In particular, note that \(i\) can directly compute, by itself, the rotation angles \(\alpha_m\) (13) for \(m \in C_i\) by using these measurements. It is clear that agent \(i\) also has to know the labels, or identifications, of its neighboring agents. In addition, it needs the local structure of the formation graph, i.e., which agents form each of the maximal cliques \(i\) belongs to. In our notation: \(I_m\) for \(m \in C_i\).

Finally, note that agent \(i\) only needs to know the above quantities expressed in its own independent local coordinates, as explained in the next section.

\(\text{D. Computation of the control inputs in the local frames}\)

A central property of the method we propose is that each agent can compute its control input in the absence of a global orientation reference, as shown next. We denote as \(\theta_i\) the rotation angle between the arbitrary global frame and the local frame in which agent \(i\) operates, and by \(\mathbf{P}_i(\theta_i) \in \mathbb{SO}(2)\) the corresponding rotation matrix. Let us now write down the partial desired motion vector for clique \(m\) (the analysis can be trivially extended to the desired motion vectors \(\mathbf{d}_i\) computed locally by \(i\) (22), using a superscript \(\prime\)’\(s\) control law, expressed in an arbitrary global frame \(G\).

\[
\mathbf{d}_{im}^\prime = \sum_{j \in I_m} \mathbf{q}_{ij}^\prime - \mathbf{R}_m^\prime \mathbf{c}_{ji}.
\]
(26)

Let us show how the rotation matrices computed in the global and local frames are related. We recall (4), which expresses \(\gamma_m\) in an arbitrary global frame:
\[
\gamma_m = \frac{1}{2N_m} \sum_{j \in I_m} || \sum_{k \in I_m} \mathbf{q}_{jk} - \mathbf{R}_m \mathbf{c}_{jk} ||^2.
\]
(27)

Agent \(i\) minimizes the cost function expressed in its local frame, i.e., \(\gamma_i^L\). Given that \(\mathbf{q}_{ij}^\prime = \mathbf{P}_i \mathbf{q}_{ij}\) for all \(j, k \in I_m\), we can write, using simple manipulation:
\[
\gamma_i^L = \frac{1}{2N_m} \sum_{j \in I_m} || \sum_{k \in I_m} \mathbf{q}_{ij}^L - \mathbf{R}_m \mathbf{c}_{jk} ||^2 = \frac{1}{2N_m} \sum_{j \in I_m} || \sum_{k \in I_m} \mathbf{q}_{jk} - \mathbf{P}_i^{-1} \mathbf{R}_m \mathbf{c}_{jk} ||^2.
\]
(28)

Given that \(\mathbf{R}_m\) minimizes \(\gamma_m\) and \(\mathbf{R}_m^L\) minimizes \(\gamma_i^L\), and this minimum is unique (Section III-A), we clearly have from (27) and (28) that \(\gamma_m = \gamma_i^L\) and hence \(\mathbf{R}_m = \mathbf{P}_i^{-1} \mathbf{R}_m^L\), i.e., \(\mathbf{R}_m^L = \mathbf{P}_i \mathbf{R}_m\). Thus, from (26):
\[
\mathbf{d}_{im}^L = \sum_{j \in I_m} \mathbf{q}_{ij}^L - \mathbf{R}_m^L \mathbf{c}_{ji} = \sum_{j \in I_m} \mathbf{P}_i \mathbf{q}_{ij} - \mathbf{P}_i \mathbf{R}_m \mathbf{c}_{ji} = \mathbf{P}_i \mathbf{d}_{im},
\]
(29)
which means that the computations referred to the two frames give an identical desired motion vector. Figure 2 (right) illustrates the variables used by the controller with respect to the global and local frames.
IV. Stability Analysis

In this section, we study the stability of the proposed control strategy. Throughout the section, we assume \( G_f \) to be a static graph. Note that, unless otherwise stated, all the entities in the Euclidean plane are expressed in an arbitrary global reference frame. Our control methodology, described in the previous section, is based on minimizing a cost function (6) defined over the set of maximal cliques in the formation graph. We present a stability analysis that relies on a description of this graph in terms of its maximal-clique intersection sets. Specifically, we use 2-clique graphs to capture these intersections. Let us denote the 2-clique graph of \( G_f \) as \( C_2(G_f) = (V_{C_2}, E_{C_2}) \). We will refer equivalently to the maximal cliques of \( G_f \) or to the nodes of \( C_2(G_f) \). Consider the following assumption:

**A1** \( C_2(G_f) \) is connected.

Observe that this assumption immediately implies that there cannot be maximal cliques of size one or two in \( G_f \). Therefore, every agent is in at least one 3-node clique of \( G_f \). Figure 3 shows an example \( G_f \) for which A1 is satisfied, and its 2-clique graph.

**Theorem 1.** If A1 holds, the multiagent system under the control laws (24), for single-integrator kinematics, or (25), for unicycle kinematics, is locally stable with respect to the desired configuration.

**Proof.** We will use Lyapunov analysis to prove the stability of the system. Let us define a Lyapunov candidate function as \( V = \gamma \). It is straightforward to see that \( V \) is positive semi-definite and radially unbounded. In addition, the equilibrium \( V = 0 \) occurs if and only if the \( N \) agents are in the desired configuration, as shown next.

Let us assume the agents are in the desired configuration and see that this leads to \( V = 0 \). Observe that this situation implies that for every pair \( i, j \) in \( 1, ..., N \), \( q_{ij} = R_{C_{ij}} \), where \( R \) expresses the rotation of the formation pattern (3). Since this holds for every pair of agents, notice in (4) that \( \gamma_m \) is zero, i.e., has its minimum possible value, \( \forall m = 1, ..., M \), if \( R_m = R \). Clearly, since every \( R_m \) must be such that its associated \( \gamma_m \) is minimum (Section III-A), we have that all of the rotations are equal to \( R \). Thus, \( V = \gamma = 0 \).

On the other hand, assume \( V = 0 \), which implies, since \( \gamma_m = 0 \) \( \forall m = 1, ..., M \), that all the sub-formations for each of the \( M \) maximal cliques have been achieved (Section III). Note, however, that this does not imply, in general, that the agents are in the global desired formation. To guarantee this, we use assumption A1 next. Note that the assumption implies that the agents form a structure of maximal cliques which have, at least, three nodes each, and for every maximal clique \( m_1 \), there exists at least one other maximal clique \( m_2 \) such that \( m_1 \) and \( m_2 \) have at least two agents in common. Then, consider two given agents \( i, j \) which are in the intersection of two given cliques \( m_1 \) and \( m_2 \). It is clear, since \( \gamma = 0 \), that \( q_{ij} = R_{m_1}c_{ij} = R_{m_2}c_{ij} \) and, therefore, \( R_{m_1} = R_{m_2} \). Now, due to connectedness of \( C_2(G_f) \), this equality can be trivially propagated throughout the \( M \) maximal cliques. Thus, \( R_m = R \) \( \forall m = 1, ..., M \), which means that \( q_{ij} = R_{C_{ij}} \) \( \forall i, j = 1, ..., N \), i.e., the \( N \) agents are in the desired configuration. After showing that \( V = 0 \) provides a characterization of the desired formation, we study next the stability of the system by analyzing the dynamics of \( V \). Notice that, given the negative gradient-based control strategy expressed in (23), we can write:

\[
\dot{V} = \sum_{i=1,...,N} (\nabla q_i V) V q_i = - \sum_{i=1,...,N} d_i^T q_i. \tag{30}
\]

Then, considering our controller for single-integrator kinematics (24), we have, by direct substitution:

\[
\dot{V} = -k_e \sum_{i=1,...,N} ||d_i||^2 \leq 0. \tag{31}
\]

Let us now consider unicycle kinematics. We denote as \( S_v \) the time-varying set of agents for which it holds that \( |\beta_i| < \pi/2 \). Since the displacement of a unicycle agent always occurs along the direction of its current heading, we have in our case that, from the linear velocity in (25), the motion vector executed by each agent \( i \) (see Fig. 2) is:

\[
\dot{q}_i = \begin{cases} k_v Q(\beta_i) d_i, & i \in S_v \\ 0, & i \not\in S_v, \end{cases} \tag{32}
\]

where \( Q(\beta_i) \in SO(2) \) expresses a rotation by the angular alignment error. Then, substituting (32) in (30):

\[
\dot{V} = -k_v \sum_{i \in S_v} \cos(\beta_i)||d_i||^2 + \sum_{i \not\in S_v} (0) \leq 0, \tag{33}
\]

where the condition that \( \dot{V} \) can never be positive results from the fact that \( |\beta_i| < \pi/2 \) \( \forall i \in S_v \), i.e., \( \cos(\beta_i) > 0 \) \( \forall i \in S_v \). By virtue of the global invariant set theorem, (31) and (33) ensure that, under the proposed control laws for single-integrator (24) or unicycle (25) kinematics, the system converges asymptotically to the largest invariant set in the set \( W = \{ q_i, i = 1, ..., N \mid V = 0 \} \). Therefore, it can be concluded that the multiagent system is locally stable with respect to the desired formation (i.e., \( V = 0 \)). \( \square \)

**Corollary 1.** If A1 is satisfied, then all stable equilibriums of the multiagent system under the controllers for single-integrator (24) or unicycle (25) kinematics are static configurations, and occur if and only if \( d_i = 0 \) \( \forall i = 1, ..., N \).
Proof. By a stable equilibrium we precisely mean a configuration that the system will not get out of, i.e., a configuration for which it holds that $\dot{V} = 0$ for all time. Let us examine these equilibriums. We look at the controller for single-integrator agents first. We immediately see from (31) that $\dot{V} = 0 \iff d_i = 0 \forall i = 1, \ldots, N$. If all $d_i$ are null, all the agents are static, see (24). This also clearly implies the equilibrium is stable. Thus, the statement of the Corollary holds.

For unicycle kinematics, suppose $\dot{V} > 0$ at some instant. Notice from (33) that this implies $d_i = 0 \forall i \in S_v$. These agents are static (25). However, it is possible that for some of the agents not belonging to $S_v$, $d_i \neq 0$. Assume this is the case. Note that even if their desired vectors are not null, the agents not in $S_v$ can never translate, due to the linear velocity defined in (25). As a result, we have that $\dot{V} = 0$ implies that none of the $N$ agents’ positions, $q_i$, can change. Therefore, from (23), no $d_i$ can change. The vectors $d_i$ being constant implies that every agent not belonging to $S_v$ such that its $d_i \neq 0$ will rotate in place, thanks to the angular velocity control in (25), seeking to align itself with the direction of its constant $d_i$. This will eventually lead, at some time instant, to $|\beta_i| < \pi/2$ for one of these agents, i.e., $\cos(\beta_i) > 0$ and, given that $d_i \neq 0$, to $V < 0$ (33), i.e., the assumed equilibrium is not stable. Hence, we can conclude that if the system is in a stable equilibrium, i.e., $\dot{V} = 0$ for all time, it must hold that $d_i = 0 \forall i = 1, \ldots, N$. The converse statement $d_i = 0 \forall i \in S_v \Rightarrow V = 0$ for all time is immediate to see from equation (33) for unicycle kinematics. In addition, observe from (25) that all $d_i$ being null implies that the unicycle agents are static, i.e., the stable equilibrium is a static configuration. \hfill \Box

Following the discussion above on local stability results for our system, let us now start the study of global convergence by presenting a Lemma that will be useful in the subsequent development.

**Lemma 1.** Let $m_1$ and $m_2$ be two maximal cliques of $G_f$ corresponding to two adjacent nodes in $C_2(G_f)$. Assume the following conditions are satisfied:

$L1$) $d_i = 0$, $i = 1, \ldots, N$.

$L2$) card$(I_{m_1} \cap I_{m_2}) = 2$, denote $I_{m_1} \cap I_{m_2} = \{i_1, i_2\}$.

$L3$) $C_i = \{m_1, m_2\}, i = i_1, i_2$.

$L4$) $d_{m_1} = 0, \forall i \in I_{m_1}, i \neq i_1, i \neq i_2$.

Then, it holds that $d_{m_1} = 0 \forall i \in I_{m_1}$, $d_{m_1} = 0 \forall i \in I_{m_2}$, and the rotation matrices in (7) satisfy $R_{m_1} = R_{m_2}$, and $\gamma_{m_1}$ (4) is zero.

Proof. Let us choose, without loss of generality, the global reference frame for which $R_{m_1} = I_2$, i.e., $\alpha_{m_1} = 0$ (13). Considering L1 and L4, we can write, using (23):

$$d_i = d_{m_1} = \sum_{j \in I_{m_1}} q_{ji} - c_{ji} = 0, \forall i \in I_{m_1}, i \neq i_1, i \neq i_2. \quad (34)$$

Let us use that $q_{ji} = -q_{ij}$, $c_{ji} = -c_{ji}$ and interchange the names of the subscripts $j$ and $i$ in (34), to obtain:

$$\sum_{i \in I_{m_1}} q_{ji} = \sum_{i \in I_{m_1}} c_{ij}, \forall j \in I_{m_1}, j \neq i_1, j \neq i_2. \quad (35)$$

Imposing the condition $\alpha_{m_1} = 0$ in (13), we can write:

$$\sum_{j \in I_{m_1}} S_{m_1 j} C_{m_1 j} = \sum_{j \neq i_1, i_2} S_{m_1 j} C_{m_1 j} + \sum_{j = i_1, i_2} S_{m_1 j} C_{m_1 j} = 0, \quad (36)$$

where the sums are for the clique $m_1$, i.e., $S_{m_1 j} = \sum_{j \in I_{m_1}} q_{ji}$. $C_{m_1 j} = \sum_{j \in I_{m_1}} c_{ji}$. Observe that, due to (35), $S_{m_1 j} C_{m_1 j} = 0$ for all $j \neq i_1, i_2$. Thus, each of the addends in the first summation in the second line of (36) is a dot product of two orthogonal vectors, and therefore vanishes. Then, we have:

$$\sum_{j = i_1, i_2} S_{m_1 j} C_{m_1 j} = \sum_{j = i_1, i_2} S_{m_1 j} C_{m_1 j} = \sum_{i \in I_{m_1}} q_{ji} + \sum_{i \in I_{m_2}} q_{ji} = 0. \quad (37)$$

Let us now focus on agents $i_1$ and $i_2$ and find constraints on their desired motion vectors which, along with the condition in (37), will lead to our result. From L3, these two agents belong to cliques $m_1$ and $m_2$ only and, due to L1, we have:

$$d_{i_1} = d_{i_1 m_1} + d_{i_1 m_2} = 0,$$

$$d_{i_2} = d_{i_2 m_1} + d_{i_2 m_2} = 0. \quad (38)$$

Observe that the sum of partial desired vectors for any given clique is null, as can be directly seen by considering the expression for the partial vectors in (22), and using equation (19), as follows:

$$\sum_{i \in I_m} d_{i m} = \sum_{i \in I_m} \sum_{j \in I_m} q_{ji} - R_m c_{ji} = 0, \quad m = 1, \ldots, M. \quad (39)$$

Consider the above condition for clique $m = m_1$ in particular. Due to L4, its sum of vectors includes only agents $i_1$ and $i_2$. Thus:

$$\sum_{i \in I_{m_1}} d_{i m_1} = d_{i_1 m_1} + d_{i_2 m_1} = 0, \quad (40)$$

an expression which will be useful later on in the proof. Observe now, from (22), that:

$$d_{i_1 m_2} = \sum_{j \in I_{m_1}} q_{ji} - R_{m_2} c_{ji}, \quad i_1, i_2. \quad (41)$$

$$d_{i_2 m_2} = \sum_{j \in I_{m_2}} q_{ji} - R_{m_2} c_{ji}, \quad i_1, i_2. \quad (42)$$

Interchanging the subscripts $i$ and $j$ in (41), and then expressing the equations for $j = i_1$ and $j = i_2$ separately, we can write:

$$\sum_{i \in I_{m_1}} q_{i_1 i} = -d_{i_1 m_1} + \sum_{i \in I_{m_1}} c_{i_1 i},$$

$$\sum_{i \in I_{m_2}} q_{i_2 i} = -d_{i_2 m_2} + \sum_{i \in I_{m_2}} c_{i_2 i}. \quad (43)$$
Analogously, from (42), we obtain:
\[ \sum_{i \in I_{m2}} q_{i1} = -d_{i1m1} + R_{m2} \sum_{i \in I_{m2}} c_{i1} \]
\[ \sum_{i \in I_{m2}} q_{i2} = -d_{i2m2} + R_{m2} \sum_{i \in I_{m2}} c_{i2}. \]  
(44)

Now, by substituting (43) in (37), we have:
\[ d_{i1m1}^T \sum_{i \in I_{m1}} c_{i1i} + d_{i2m1}^T \sum_{i \in I_{m1}} c_{i2i} = 0, \]
and using in (45) that, from (40), \( d_{i1m1} = -d_{i2m1}, \) gives:
\[ d_{i1m1}^T ( \sum_{i \in I_{m1}} c_{i1i} - c_{i2i}) = d_{i1m1}^T c_{i112} = d_{i2m1}^T c_{i112} = 0. \]  
(46)

Observe that (46) indicates that \( d_{i1m1} \) and \( d_{i2m1} \) are parallel to \( c_{i112}. \) We can then write:
\[ d_{i1m1} - d_{i2m1} = k_{12} c_{i112}, \]
and, substituting (38) in (47):
\[ d_{i1m2} - d_{i2m2} = -k_{12} c_{i112}, \]  
(48)

for some scalar \( k_{12}. \) We now define \( d_{i1m}' = d_{i1m}/N_m, \) \( i = i_1, i_2, \) \( m = m_1, m_2. \) Notice that subtracting the two equations in (43), we have:
\[ q_{i1} = -d_{i1m}' + d_{i2m} + c_{i112}. \]  
(49)

Then, substituting (47) yields:
\[ q_{i1} = (1 - (k_{12}/N_{m1}))(c_{i112}). \]  
(50)

On the other hand, subtraction of the equations in (44) gives:
\[ q_{i1} = -d_{i1m}' + d_{i2m} + R_{m2} c_{i112}, \text{ i.e.,} \]
\[ q_{i1} = d_{i1m}' - d_{i2m} = R_{m2} c_{i112}. \]  
(51)

Substituting (48) and (50) in the left-hand side of (51) yields:
\[ (1 - \kappa)c_{i112} = R_{m2} c_{i112}, \]
\[ \kappa = k_{12}[(1/N_{m1}) + (1/N_{m2})]. \]
(52)

where \( \kappa = k_{12}/[1/N_{m1} + 1/N_{m2}]. \) As multiplying by \( R_{m2} \) does not modify the norm of \( c_{i112}, \) and disregarding the case \( \kappa = 2, \) that can be seen to correspond to a degenerate configuration in which \( q_{i1} = c_{i112} = (N_{m1} - N_{m2})/(N_{m1} + N_{m2}), \) see (50), we clearly have that (52) can hold only if \( k_{12} = 0 \) and \( R_{m2} = I_2 = R_{m1}. \) Therefore, from (47), \( d_{i1m1}' = d_{i2m1}. \) Then, due to (40), these two vectors are null. This implies \( d_{i1m1} = 0 \) \( \forall i \in I_{m1} \) and, hence, substituting (22) in (4), we see that \( \gamma_{m1} = 0. \) Moreover, from (38), \( d_{i1m2} = d_{i2m2} = 0. \]

In search of global convergence guarantees, we formulate the following assumptions regarding the formation graph:
\[ A2) \ \text{card}(I_{m} \cap I_{n}) = 2 \ \forall \{m, n\} \in \mathcal{E}_{C_2}, \]  
\[ I_{m} \cap I_{n} = \emptyset \text{ otherwise (i.e., every intersection set of two maximal cliques of } \mathcal{G}_f \text{ either contains exactly two agents, or is empty).} \]
\[ A3) \ I_{m} \cap I_{n} \cap I_{r} = \emptyset, m \neq n, m \neq r, n \neq r, m, n, r \in 1, \ldots, M \text{ (i.e., the intersection sets between maximal cliques of } \mathcal{G}_f \text{ are mutually disjoint).} \]
\[ A4) \ C_2(\mathcal{G}_f) \text{ is a tree.} \]

Note that we replace \( A1 \) by the stronger condition \( A4. \) Clearly, Theorem 1 and Corollary 1 hold if \( A4 \) does. We enunciate next our global stability result.

**Theorem 2.** Suppose \( A2-A4 \) are satisfied. Then, the multijagent system under the control laws (24), for single-integrator kinematics, or (25), for unicycle kinematics, converges globally to the desired configuration, and the attained formation is static.

**Proof.** We build on the development presented for Theorem 1, using the same Lyapunov candidate function \( V = \gamma. \) We proceed by examining the possible stable equilibriums of the system, and showing that they only include the case \( V = 0. \) From Corollary 1, a stable equilibrium is characterized, for the two kinematic models considered, by the condition \( d_{i1} = 0, \ i = 1, \ldots, N. \) Let us assume this condition is satisfied. Then, the rest of the proof relies on applying Lemma 1 to pairs of nodes in \( C_2(\mathcal{G}_f), \) i.e., pairs of maximal cliques in \( \mathcal{G}_f. \) Clearly, the assumption that all \( d_{i1} = 0, \ A2, \) and \( A3 \) together imply that conditions L1, L2 and L3 of Lemma 1 are always satisfied for any pair of adjacent nodes in \( C_2(\mathcal{G}_f). \) Thus, to see if Lemma 1 is applicable to a given couple of nodes, we will only need to check if L4 is satisfied. Consider then a given leaf node \( l \) in \( C_2(\mathcal{G}_f), \) which is a tree (A4), and its adjacent node \( a. \) Denote \( I_l \cap I_a = \{r_1, r_2\}. \) Being a leaf node, and due to \( A2, \) all the agents in \( l \) except \( r_1 \) and \( r_2 \) belong to maximal clique \( l \) only, and thus their partial desired vectors satisfy, from (23), \( d_{i1} = d_{i2} = 0, i \in I_l, i \neq r_1, i \neq r_2. \) Then, clearly, L4 holds and Lemma 1 can be applied to the pair \( l \) (in the role of \( m_1 \) and \( a \) (in the role of \( m_2 \)). This way, by extension, it is ensured that for every clique \( m \) that is a leaf node of \( C_2(\mathcal{G}_f), \) \( \gamma_m = 0 \) and \( d_{i1m} = 0 \) \( \forall i \in I_m. \)

Let us define \( C_2(\mathcal{G}_f) \) as the induced subgraph of \( C_2(\mathcal{G}_f) \) containing all its nodes except the leaves. Clearly, \( C_2(\mathcal{G}_f) \) is also a tree. Let us consider any one of its leaf nodes, and denote it as \( l'. \) We have:
1) For every agent \( i \) belonging only to \( l', \) (i.e., \( C_1 = \{l'\} \)), from (23), \( d_{i1r} = d_{i2r} = 0. \)
2) Notice \( l' \) is adjacent to one or multiple leaves of \( C_2(\mathcal{G}_f). \) As we just showed, all the partial desired motion vectors corresponding to leaf nodes of \( C_2(\mathcal{G}_f) \) are null. Then, for the agents \( i \) shared by \( l' \) and a leaf of \( C_2(\mathcal{G}_f), d_{i1r} = d_{i2r}. \) Since all \( d_1 \) are assumed null, we have \( d_{i1r} = 0. \)
3) Being a leaf node of \( C_2(\mathcal{G}_f), l' \) is adjacent to exactly one node, which we denote as \( a', \) that is not a leaf node of \( C_2(\mathcal{G}_f). \) These two maximal cliques share two agents; let us denote \( I_{l'} \cap I_{a'} = \{r_1, r_2\}. \)

As, clearly, points 1), 2) and 3) encompass all the agents in clique \( l', \) we have \( d_{i1r} = 0 \) \( \forall i \in I_{l'}, i \neq r_1, i \neq r_2. \) Thus, we can apply Lemma 1 to \( l' \) (in the role of \( m_1 \) and \( a', \) in the role of \( m_2 \)), since L4 holds for this pair of cliques. In consequence, for every node \( m \) that is a leaf of \( C_2(\mathcal{G}_f), \) \( \gamma_m = 0 \) and \( d_{i1m} = 0 \) \( \forall i \in I_m. \) i.e., the same result shown above for the leaves of \( C_2(\mathcal{G}_f). \)

It is then clear that we can consider subsequent induced tree subgraphs of \( C_2(\mathcal{G}_f) \) and apply the reasoning above recursively, until reaching a trivial case (a final, irreducible tree with either one or two nodes). As a result, we have that \( \gamma_m = 0 \) for all the nodes in \( C_2(\mathcal{G}_f), \) i.e., for all the \( M \) maximal cliques. We
can conclude, then, that if $d_i = 0$, \( i = 1, \ldots, N \), i.e., if $\dot{V} = 0$ for all time (Corollary 1), it holds that $\gamma_m = 0$, $m = 1, \ldots, M$, i.e., $V = 0$. The converse is also true since $\dot{V} = \gamma = 0$, see (4), (6), implies $d_i = 0$, \( i = 1, \ldots, N \) (23). Hence, $\dot{V} = 0$ for all time $\iff V = 0$, i.e., the multiagent system converges globally to the desired formation. In addition, from Corollary 1, the configuration the agents reach is static. 

\[ \Box \]

V. DISCUSSION OF VALID FORMATION GRAPH TOPOLOGIES

We analyze in this section the characteristics of $G_f$ arising from the introduced topological assumptions. Let us start by considering A1. A formation graph satisfying this assumption is illustrated in Fig. 3. Firstly, we note that A1 specifies a class of graphs that are rigid in two dimensions. To see this, observe first that a clique is a rigid graph. Assume the intersection of every pair of maximal cliques of $G_f$ corresponding to adjacent nodes of $C_2(G_f)$ contains exactly two nodes. Notice then that, due to A1, $G_f$ can be constructed, starting from one of its maximal cliques, by applying successive edge-attachment operations, as defined in [17], to incorporate all the other maximal cliques. These operations consist in merging an edge of each of two given graphs into a single edge of a new graph that is a fusion of the two. In [17], it was shown that such edge-attachment procedures generate a rigid graph for two input graphs that are rigid. Thus, clearly, $G_f$ is rigid in two dimensions. If there are adjacent nodes in $C_2(G_f)$ associated with maximal cliques of $G_f$ which share more than two nodes, one can always eliminate some of the edges to obtain a subgraph of $G_f$ for which the relevant maximal clique intersections are two-node, and thus the reasoning above also applies. Note that not all rigid graphs satisfy A1.

As shown in the previous section, global convergence to the desired formation is guaranteed for any formation graph whose topology conforms with A2-A4. Clearly, this augmented set of assumptions also implies the graph is rigid in two dimensions. The class of rigid graphs satisfying A2-A4 is illustrated with four exemplary topologies, containing maximal cliques of up to six agents, in Fig. 4. Observe that the specification resulting from these assumptions provides flexibility, as it allows structures that are made up from maximal cliques of different sizes, and can be extended to arbitrary numbers of nodes. For instance, the chained structure in bottom-left of the figure can be prolonged to contain any number of four-node maximal cliques, and a more general topology with heterogeneous maximal cliques, such as the example depicted in the center, is also arbitrarily extendable. Observe that, regardless of the total number of nodes in $G_f$, any given agent only has to interact with (i.e., measure the relative position of) a small number of neighbors, which indicates the distributed and partial information-based nature of our controller. We require a denser (i.e., with more edges) formation graph than distance-based formation controllers, which are valid for general rigid graphs, but let us note that as the number of agents grows, the minimum number of edges we need is in the same order as the number of edges in a minimally rigid graph.

VI. SIMULATIONS

In this section, the effectiveness of our controller is illustrated in simulation. In our tests, we considered that a sensing or communication infrastructure in the team of agents allowed each of them to measure the relative positions of its neighbors in $G_f$, as commented in Section II. We first present results from an example where the desired formation was composed of twelve unicycle agents arranged in two concentric circles. The formation graph $G_f$ consisted of five maximal cliques, each containing four agents, with the chained structure depicted in bottom-left of Fig. 4. Figure 5 displays the paths followed by the agents using our proposed controller, showing how they reach the formation from an arbitrary initial configuration. The control law was computed for each agent in a local reference frame aligned with its heading. Observe that the final group shape has arbitrary translation and rotation in the workspace. Notice as well that the final headings of the agents are arbitrary. It would be straightforward to control these headings, if desired, by making the agents rotate in place once the formation has been attained. We also display in the same figure the linear and angular velocities of the agents and the evolution of the angles, expressed in a common reference frame, of the rotation matrices $R_m$ for the five maximal cliques. It can be observed that the angles converge to a common value as the formation is achieved. The vanishing global and partial cost functions are also depicted.

We also illustrate a second example where a group of forty agents was considered. This time, the agents obeyed the single-integrator kinematic model. The simulation results for this example are displayed in Fig. 6. The geometry of the rigid desired formation and the edges of the formation graph $G_f$, which consisted of eighteen maximal cliques of sizes ranging from three to six agents, are shown. Notice that this graph also belongs to the class defined by assumptions A2-A4, for which our controller guarantees global formation stabilization. Since single-integrator agents do not have a defined heading, we computed the control law considering for each agent an arbitrarily oriented local reference frame. The paths followed by the agents when using our proposed controller illustrate their successful convergence to a pattern having the same shape and size as the desired one. We also display the norms of the instantaneous velocity vectors $u_i$, the cost functions, and the angles of the rotation matrices for each of the maximal cliques. These angles, expressed in a common reference frame,
converge to a common value as the rigid desired formation is attained.

VII. CONCLUSION

We have presented a distributed control method to stabilize a set of mobile agents to a rigid formation. To alleviate the need for the agents to rely on centralized sensing or shared reference systems, we have proposed a coordinate-free approach which can be implemented using only partial relative position information measured locally. Our controller can be used on unicycle-type platforms and has been shown to be globally stable for a class of rigid formation graphs. Possible directions for future work include addressing a similar distributed stabilization problem in 3D space, where rigidity-related graph conditions are in general more complex to characterize. In addition, it can be interesting to study the case where the formation graph is considered to change dynamically as the agents move.

ACKNOWLEDGMENT

The authors wish to thank María L. Rodríguez-Arévalo for helpful discussions regarding graph theory.

REFERENCES


