2011 50th IEEE Conference on Decision and Control and
European Control Conference (CDC-ECC)
Orlando, FL, USA, December 12-15, 2011

Information Flow Based Connectivity Maintenance of A Multi-agent System During Formation Control

[†]Z. Kan, [†]A. P. Dani, [‡]J. M. Shea, and [†,‡]W. E. Dixon
[†]Department of Mechanical and Aerospace Engineering,
[‡]Department of Electrical and Computer Engineering,
University of Florida, Gainesville, FL 32611-6250, USA.
Email: kanzhen0322@ufl.edu, ashwin31@ufl.edu, jshea@ece.ufl.edu, wdixon@ufl.edu

Abstract— **Network connectivity is paramount for multi-agent systems with limited sensing and communication capabilities, since agents need to coordinate and communicate to make appropriate decisions in formation control. The goal in this work is to steer a group of agents to a desired configuration from any given initially connected graph in a decentralized manner, without partitioning the underlying network, and avoiding collision with other agents and moving obstacles. To maintain network connectivity, an information flow is proposed to specify the communication among agents. The underlying network graph is connected as long as all the information flows are maintained. Based on the approach of information flow, each agent is able to choose a short path to reach the desired agent in the information graph by dynamically building new communication links or breaking existing links. A navigation function formulation is used to maintain the information flow among agents, and guarantee the convergence of the system to the desired configuration.**

I. INTRODUCTION[1]

An individual agent operating to achieve a formation is required to coordinate its movements with other agents, requiring that the agents maintain connectivity through local sensing or network communications. Achieving a desired formation or simply remaining connected is a global network goal; however, gathering information about the global network structure leads to pathologies related to bandwidth consumption and latency propagation. The inherent challenge is to achieve global objectives such as connectivity (i.e., the group does not partition) and formation control while using local (decentralized) feedback.

Artificial potential field based methods have been widely used in formation control, where an attractive potential is placed at the goal configuration and a repulsive potential is used to prevent collisions among the agents and obstacles [1], [2]. A specific type of artificial potential, called a navigation function, achieves a unique minimum (c.f., [3]–[5]) and has been used in motion control for multi-agent systems (see e.g., [6]–[10]). A centralized navigation function control strategy is proposed in [11] to steer a group of mobile agents with limited sensing capabilities to achieve a desired formation. In [12], the problem in [11] is solved using a decentralized navigation function. However, controllers developed in these representative results are based on the assumption that the network is always connected (any agent has access to the states of the other agents) without regard to the communication constraints. The assumption of a connected graph is restrictive for a mobile network, where communication depends on the distance between agents, which can also be a function of the environment and available transmitting power. In practical applications, each agent has limited communication and sensing capabilities to determine required relative position and velocity information from other agents.

Motivated by the limited communication and sensing capabilities, our recent work in [13] developed a decentralized control scheme based on a navigation function formulation. This method ensures network connectivity and stabilizes a group of agents in a desired formation using only local information among immediate agents. However, the result in [13] requires that the initial graph is connected in a desired way so that no initial communication link is allowed to be broken during the motion. The constraints on the initial graph connections is also presented in the works, such as [14] and [9], where potential function is used to maintain the existing communication link. Based on the work in [13], the current result is motivated by the desire to design a decentralized controller for each agent to achieve a particular configuration (e.g., parallel, circular and helical formations) in the presence of limited communication and sensing capability *from any given initially connected graph*. Inspired by the work in [15] and [16], where a locally computable term is designed to provide a sufficient condition for connectedness of the underlying network, an information flow is proposed for the formation control in this work to specify the required communication among agents. Each information flow is realized by a series of communication links in the communication graph. An edge metric is designed to indicate the freedom of each agent to undergo arbitrary motion without disconnecting the communication link. Based on this edge metric, each agent is able to choose a short path to reach the desired agent in an information graph by dynamically building new communication links or breaking existing links to the agents within its communication zone. A navigation function based decentralized controller is then applied to achieve the desired configuration, while ensuring the underlying network graph

[1]This research is supported by National Science Foundation grant number CNS-0626863.

does not partition by maintaining all the information flow available to agents during the motion. The convergence is then shown by using Rantzer's Dual Lyapunov Theorem [17].

The proposed approach is unique in several ways. It is more general in the sense that any given initially connected graph can be reorganized to a pre-specified connected configuration without disconnecting the underlying network, while avoiding obstacles, and using only local information (i.e., the states of other agents or moving obstacles located within its sensing and communication zone). Existing control algorithms more rely on maintaining the connectivity of the whole communication graph. In this work, to maintain network connectivity, the main focus is to maintain the information flow among nodes. The information among agents in an information flow is available, as long as there exists at least one path in the communication graph connecting two particular nodes. Thus, to obtain the information from other agents via information flow, generally fewer communication links are required to be maintained compared to the efforts required to maintain connectivity of the whole communication graph. Finally, the developed navigation function guarantees that the agents will converge to the desired configuration and communication links can be formed or broken in a smooth manner without introducing discontinuity.

II. Problem Formulation

Consider a network composed of N agents in the workspace $\mathcal{F}$, where agent i moves according to

$$\dot{q}_i = u_i, \; i = 1, \cdots, N \tag{1}$$

where $q_i = [x_i \; y_i]^T \in \mathbb{R}^2$ denotes the position of agent i in a two dimensional (2D) plane, and $u_i \in \mathbb{R}^2$ denotes the velocity of agent i (i.e., the control input). The workspace $\mathcal{F}$ is assumed to be circular and bounded with radius R^2. Each agent in $\mathcal{F}$ is assumed by a point-mass with a limited communication and sensing capability encoded by a disk area around itself. For simplicity and without loss of generality, the following development is based on the assumption that the sensing zone is the same as the communication zone, both with radius R_c. Two moving agents can communicate with each other if the relative distance is less than the radius R_c. Moving obstacles can be sensed whenever they enter the sensing zone of the agent, and it is assumed that the maximal speed of the moving obstacle is less than that of the agent. All the agents are assumed to have equal actuation capabilities, and have real time knowledge of its own states.

The interaction of a multi-agent system is modeled by two graphs, the *communication graph* and the *information graph*. Since the agent can only communicate with other agents located within the communication zone to obtain the required information, the inter-communication among agents is modeled as a undirected communication graph $\mathcal{G}_c = (\mathcal{V}, \mathcal{E}_c(t))$, with $\mathcal{V}$ denoting the index set of all nodes and the set of edges $\mathcal{E}_c = \{ (i,j) \in \mathcal{V} \times \mathcal{V} | \, d_{ij} \leq R_c \}$, where node i and node j represents the agents located at a position q_i and q_j, and $d_{ij} \in \mathbb{R}^+$ is the distance between them, defined as $d_{ij} = \|q_i - q_j\|$. In the $\mathcal{G}_c$, the edge (i,j) denotes a bidirectional communication link between node i and j, which indicates that node i and j have access to the states of each other. The *communication neighborhood* of node i, $\mathcal{N}_i^C$, (i.e., all the agents within the communication zone of agent i), is given by $\mathcal{N}_i^C = \{ j, \; j \neq i | \, j \in \mathcal{V}, (i,j) \in \mathcal{E}_c \}$.

To indicate which agents need to exchange information during the motion, an information graph is designed as $\mathcal{G}_I = (\mathcal{V}, \mathcal{E}_I)$, with the edge set defined as $\mathcal{E}_I = \{ (i,j) | \, j \in \mathcal{N}_i^I \}$, where $\mathcal{N}_i^I$ denotes the *information neighborhood* of node i which indicates the set of nodes that node i is required to communicate with to achieve the desired configuration. The desired configuration is determined in advance and specified by the desired relative position and orientation between two nodes. The desired position of a node i, denoted by q_{di}, is defined as

$$q_{di} = \left\{ q_i | \, \|q_i - q_j - c_{ij}\|^2 = 0, \; j \in \mathcal{N}_i^I, \right\}, \tag{2}$$

where $c_{ij} \in \mathbb{R}^2$ represents the desired relative position and orientation of node i with a particular node $j \in \mathcal{N}_i^I$. Since node i is designed to achieve the desired relative pose to a node $j \in \mathcal{N}_i^I$ according to (2), it is necessary for node i to always has access to the states of node j during the motion. Therefore, the communication between node i and j is necessary. Such necessary communication is represented by the information graph $\mathcal{G}_I$ with the edge $(i,j) \in \mathcal{E}_I$, named an *information flow* I_{ij}, denoting the required communication between node i and j. Note that the information flow is also bidirectional (i.e., if node i requires information from node j, then node j also requires data from node i).

The differences between a communication graph $\mathcal{G}_c$ and an information graph $\mathcal{G}_I$ is that the neighborhood $\mathcal{N}_i^C$ in the $\mathcal{G}_c$ is a time varying set, since other nodes may enter or leave the communication region of node i at any time instant, while the neighborhood $\mathcal{N}_i^I$ in the $\mathcal{G}_I$ is a static set which is specified by the desired configuration in advance. In other words, the $\mathcal{G}_c(t)$ is a dynamic graph during the motion while the $\mathcal{G}_I$ is a fixed topology representing the desired configuration. The information flow is given in advance and can be considered as a underlying design requirement about the connectivity of nodes, where the two nodes may or may not be connected directly in the graph $\mathcal{G}_c$, while communication graph reflects the status of the connectivity in real time, where the edge in the graph $\mathcal{G}_c$ indicates that two nodes are located with a distance less than R_c.

A *collision region* is defined for each agent i as a small disk area with radius $\delta_1 < R_c$ around the agent i, such that any other agent $j \in \mathcal{V}$, or an obstacle $k \in \mathcal{M}$ inside this region is considered as a potential collision with agent i, where $\mathcal{M}$ denotes the set of moving obstacles. To ensure connectivity, an *escape region* for each agent i is defined as the outer ring of the communication area with radius r,

[2] A sphere world can be extended to a star world by a translated scaling map as shown in [3], since they are both topological discs.

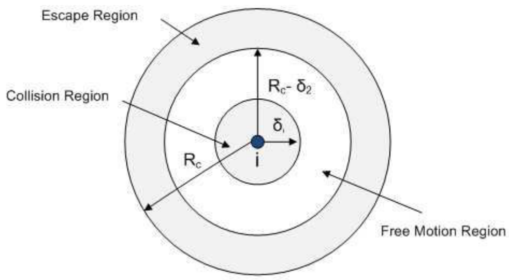

Fig. 1. Collision and escape regions for node i.

$R_c - \delta_2 < r < R_c$, where $\delta_2 \in \mathbb{R}$ is a predetermined buffer distance. Edges formed with a node $j \in \mathcal{N}_i^C$ in the escape region are in danger of breaking. Agent i moves with the constraint of avoiding collision with other agents and moving obstacles located in the collision region, and preventing the breakage of the communication link between agents located in the escape region. The collision and escape regions for agent i are shown in Fig. 1. The region in the sensing zone apart from collision and escape regions can be considered as the free motion region, since agent i can move freely to perform its own task.

To relax the assumption that the initial graph is connected in a desired edge neighborhood in the work [13], the goal in this paper is to develop a decentralized controller u_i that will (i) ensure network connectivity during the motion, and (ii) enable the system to stabilize to a desired configuration from a connected initial graph, and (iii) avoid collisions among agents as well as other moving obstacles during the motion. To achieve this goal, the subsequent development is based on the following assumptions.

Assumption 1: The initial graph $\mathcal{G}$ is connected and the initial positions do not coincide with some unstable equilibria (i.e., saddle points).

Assumption 2: The desired formation is specified in advance and is valid, which implies that the desired configuration is connected and will not lead to a collision or the desired configuration will not lead to a partitioned graph, i.e., $\delta_1 < \|c_{ij}\| < R_c - \delta_2$.

III. Connectivity on Information Flow

A framework of maintaining the network connectivity in graph $\mathcal{G}_c$ using information flow is proposed in this section. Since the initial graph is a connected graph by Assumption 1, there must exist a series of communication links in the graph $\mathcal{G}_c$ connecting node i and j, where $(i, j) \in \mathcal{E}_I$. As long as the path is always maintained (i.e., I_{ij} is maintained), node i and j are able to exchange information to reach the desired pose. If the information flow I_{ij} in the graph $\mathcal{G}_I$ is always realizable by a connected path in $\mathcal{G}_c$ for $\forall i$, the graph $\mathcal{G}_c$ will also be connected. Hence, the key to maintain network connectivity is to ensure that any information flow I_{ij} can be realized by a series of connected communication links and maintained in $\mathcal{G}_c$. However, in a connected graph $\mathcal{G}_c$, the information flow I_{ij} can be realized through several different paths. In this work, we are not only interested in maintaining the information flow I_{ij}, but also want to find a short path to connecting i and j.

To reduce the notational complexity, the path length of the information flow I_{ij} is assumed to be at most two, which means node i and node j are connected by at most one mutual neighbor in the graph $\mathcal{G}_c$[3]. The mutual node is called the *relay node*, since it is used to pass the information between node i and j. To indicate the freedom of motion that each agent can take without disconnecting the communication link, inspired by the work of [15], [16], a locally measurable term referred as edge robustness, δ_{mn}, is defined as

$$\delta_{mn} = \frac{1}{2}(R_c - d_{mn}) \tag{3}$$

for any two immediate nodes m and n in the communication graph $\mathcal{G}_c$ (i.e., $(m, n) \in \mathcal{E}_c$). The edge robustness δ_{mn} is used to measure the robustness of the edge (m, n), since node m and n will remain connected with each other, unless both of them are displaced by a distance of δ_{mn} at most. Therefore, a greater term δ_{mn} indicates more freedom of motion. Due to the motion of the nodes, some node may enter the communication zone of both node i and j at some time instant for an information flow I_{ij}. In this case, there may be a multiple choice of the relay node to connect node i.and j. Using (3), the length of the two-edge path l_{ij}, is represented as $l_{ij} = d_{ir} + d_{rj} = 2R_c - 2(\delta_{ir} + \delta_{rj})$, where δ_{ir} and δ_{rj} are the robustness of each communication link (i, r) and (r, j) computed from (3) respectively. Finding the shortest path for I_{ij} (i.e., minimizing l_{ij}) is equal to maximizing the addition of δ_{ir} and δ_{rj}, since R_c is a constant. Now the path robustness is defined as $\Delta_{I_{ij}} = \delta_{ir} + \delta_{rj}$, and we are seeking to minimize the time delay in communication by choosing the shortest path, and thereby maximizing the path robustness. Based on the analysis above, the relay node is determined by

$$r = \arg \max_{r \in \mathcal{N}_i^C \cap \mathcal{N}_j^C} \Delta_{I_{ij}}, \tag{4}$$

where the maximum taken over the intersection of communication neighbors, $\mathcal{N}_i^C \cap \mathcal{N}_j^C$, aims to find a node providing the shortest path connecting node i and j.

Remark 1: Motivation for choosing the addition of edge robustness as the path robustness, instead of choosing the minimum of the edge robustness (e.g., [15] and [16]) as the path robustness, is to avoid introducing discontinuity in the control algorithm. However, using the addition of edge robustness may overestimate individual edge robustness, since the agents can have different freedom of motion. As a first attempt to solve the formation problem with arbitrary initial graph, the main focus here is to ensure the connectivity of information flow. The future research will try to achieve a balance between taking into account the freedom of motion for each agent and achieving a group objective, e.g., formation control.

[3] The objective of this paper is not restricted to the assumed path length constraint and can be extended to a longer path length (e.g., the technique developed in [18]).

IV. Control Design

Consider a decentralized navigation function candidate $\varphi_i : \mathcal{F} \to [0,1]$ for node i as

$$\varphi_i = \frac{\gamma_i}{(\gamma_i^\alpha + \beta_i)^{1/\alpha}}, \tag{5}$$

where $\alpha \in \mathbb{R}^+$ is a tuning parameter, $\gamma_i : \mathbb{R}^2 \to \mathbb{R}^+$ is the goal function, and $\beta_i : \mathbb{R}^2 \to [0,1]$ is a constraint function.

The goal function γ_i in (5) drives the system to a desired configuration, specified in terms of the desired relative pose with respect to the information neighbor $j \in \mathcal{N}_i^I$. The goal function γ is designed as

$$\gamma_i = \sum\nolimits_{j \in \mathcal{N}_i^I} \|q_i - q_j - c_{ij}\|^2 . \tag{6}$$

The gradient and Hessian matrix of γ_i are given as

$$\nabla_{q_i}\gamma_i = \sum\nolimits_{j \in \mathcal{N}_i^I} 2(q_i - q_j - c_{ij}) \tag{7}$$

and

$$\nabla_{q_i}^2 \gamma_i = 2I_2\zeta_i, \tag{8}$$

where I_2 is the identity matrix in $\mathbb{R}^{2\times 2}$ and $\zeta_i \in \mathbb{R}^+$ denote the number of information neighbors in the set $\mathcal{N}_i^I$. Since the Hessian matrix of γ_i (8) is always positive definite, the goal function (6) has an unique minimum and the minimum is reached only when $\nabla_{q_i}\gamma_i = 0$, which implies that q_i and q_j achieves desired relative pose from (7).

The constraint function β_i in (5) is designed for node i as

$$\beta_i = B_{i0} \prod\nolimits_{j \in \mathcal{N}_i^I} b_{ij}^r \prod\nolimits_{k \in \mathcal{N}_i^C \cup \mathcal{M}} B_{ik}, \tag{9}$$

to ensure connectivity of every information flow I_{ij}, and collision avoidance with workspace boundary, adjacent nodes and moving obstacles at each time instant. In (9), $b_{ij}^r \triangleq b(q_i, q_r) : \mathbb{R}^2 \to [0,1]$ ensures connectivity of an information flow I_{ij} (i.e., guarantees that the relay node r will always connected to node i), and is designed as

$$b_{ij}^r = \begin{cases} 1 & d_{ir} \le R_c - \delta_2 \\ -\frac{1}{\delta_2^2}(d_{ir} + 2\delta_2 - R_c)^2 + \frac{2}{\delta_2}(d_{ir} + 2\delta_2 - R_c) & R_c - \delta_2 < d_{ir} < R_c \\ 0 & d_{ir} \ge R_c. \end{cases} \tag{10}$$

Note that node i is aware of δ_{rj} and $\mathcal{N}_j^C$ in (4) through communication with node j. Thus, the node r can be determined locally from (4). Also in (9), $B_{ik} \triangleq B(q_i, q_k) : \mathbb{R}^2 \to [0,1]$, for point $k \in \mathcal{N}_i^C \cup \mathcal{M}$, ensures that node i is repulsed from all nodes or moving obstacles located within its sensing zone to prevent a collision, and is designed as

$$B_{ik} = \begin{cases} -\frac{1}{\delta_1^2}d_{ik}^2 + \frac{2}{\delta_1}d_{ik} & d_{ik} < \delta_1 \\ 1 & d_{ik} \ge \delta_1. \end{cases} \tag{11}$$

Similarly, the function B_{i0} in (9) is used to model the potential collision of node i with the workspace boundary, where the positive scalar $B_{i0} \in \mathbb{R}$ is designed similar to B_{ik} with the replacement of d_{ik} by d_{i0}, where $d_{i0} \in \mathbb{R}^+$ is the relative distance of the node i to the workspace boundary defined as $d_{i0} = R - \|q_i\|$.

Based on the definition of the navigation function candidate, the decentralized controller for each node is designed as

$$u_i = -K_i \nabla_{q_i}\varphi_i, \tag{12}$$

where K_i is a positive gain, and $\nabla_{q_i}\varphi_i$ is the gradient of φ_i with respect to q_i, given as

$$\nabla_{q_i}\varphi_i = \frac{\alpha\beta_i\nabla_{q_i}\gamma_i - \gamma_i\nabla_{q_i}\beta_i}{\alpha(\gamma_i^\alpha + \beta_i)^{\frac{1}{\alpha}+1}}. \tag{13}$$

In (10) and (11), b_{ij}^r and B_{ik} are both designed to be continuous and differentiable functions in $(0, R_c)$, with b_{ij}^r achieving the minimum when the communication link (i, r) is about to be broken (e.g., $d_{ir} = R_c$) and B_{ik} achieves the minimum when nodes i and k are about to collide. The constraint function only takes effect whenever the node i has the potential to break an existing communication link or collide with other nodes. The gradient of b_{ij}^r and B_{ik} are the zero vector in the free motion region, (i.e., the interval of $(\delta_1, R_c - \delta_2)$), as shown in the Fig. 1, which indicates that node i is only driven by its goal function (6) to form the desired relative pose with node $j \in \mathcal{N}_i^I$ from (12) and (13). If node i dynamically builds new communication links or breaks existing links to the agents within the free motion region, the controller is still continuous from (13), since $\nabla_{q_i}\beta_i = 0$ and $\beta_i = 1$ in the free motion region. In contrast with the discontinuity introduced in the switching topology in current literature (see e.g., [19]), this highlighted feature enables the smooth switch between node i and other connected nodes.

A. Connectivity and Convergence Analysis

The previous development indicates that the graph $\mathcal{G}_c$ is connected if the information flow I_{ij} is maintained in $\mathcal{G}_c$. The following proof indicates that the controller in (12) can guarantee connectivity of information flow I_{ij} in $\mathcal{G}_c$.

Proposition 1: For any information flow I_{ij} with node r as the relay node, the control law (12) will guarantee that I_{ij} is maintained all the time, that is node i and j are connected in a communication path in $\mathcal{G}_c$ with length at most two.

Proof: Assuming that an information flow I_{ij} is realized in communication graph $\mathcal{G}_c$ by a path from node i to node j through a mutual node r. From the definition of relay node, it is known that $r \in \mathcal{N}_i^C \cap \mathcal{N}_j^C$, which means node r is located in the communication zone of both i and j. To show that the edge (i, r) is maintained under the control law (12), consider node i located at a point $q_0 \in \mathcal{F}$ that causes $b_{ij}^r = 0$, which indicates that node i is about to disconnect with node r. Since $b_{ij}^r = 0$, $\beta_i = 0$ from (9), the navigation function achieves its maximum value from (5). Since φ_i is maximized at q_0, no open set of initial conditions can be attracted to q_0 under the negated gradient control law designed in (12). Therefore, the communication link between node i and r is maintained by the controller (12). Following the same procedure, the edge (r, j) can be maintained by a similar control applied to node j. Due to the motion of the nodes, some other node k may provide a shorter path connecting node i and j than node r from some time instant. When this occurs, it is reasonable to create a new path from node i to node j through the node k to maintain the information flow I_{ij}. The relay node k can be determined according to (4), and to not introduce any discontinuity, node k can be switched from node r in the free motion region of both node i and j. Following the analysis above, the connectivity of the new path can also be guaranteed. ■

B. Convergence Analysis

Our previous work in [13] proven that the proposed φ_i in (5) is a qualified navigation function, which guarantees convergence of the system to the desired configuration. In this section, instead of using Morse Theory to show the convergence as in [13], a new tool, Rantzer's Dual Lyapunov Theorem [17], is used to show the convergence of the system to the desired configuration. From [13], the control law (12) ensures that almost all initial conditions are either brought to a saddle point or to the unique minimum q_{di} on a compact connected manifold with boundary, as long as the tuning parameter α in (5) satisfies that $\alpha > \max\{1, \Gamma(\varepsilon)\}$, where $\Gamma(\varepsilon)$ is developed in [13]. The following development uses Rantzer's Dual Lyapunov Theorem to show that the undesired critical points (i.e., saddle points) are all measure zero, and the system can only converge to the unique minimum q_{di}. For the bounded workspace in this work, a variation of Rantzer's Dual Lyapunov Theorem is stated as [20]:

Theorem 1: Given $\dot{x}(t) = f(x(t))$, where $f \in C^1(S, \mathbb{R}^n)$, $f(0) = 0$, and S is an open, bounded subset of $\mathbb{R}^n$, and positively invariant, and suppose $x^* = 0 \in S$ is a stable equilibrium point. Furthermore, suppose there exists a function $\rho \in C^1(S - \{0\}, \mathbb{R})$ such that $\rho(x)f(x)/\|x\|$ is integrable on $\{x \in S : \|x\| \geq 1\}$ and

$$[\nabla \cdot (f\rho)] > 0 \text{ for almost all } x \in S. \tag{14}$$

Then, for almost all initial states $x(0) \in S$, the trajectory $x(t)$ exists for $t \in [0, \infty)$ and tends to zero as $t \to \infty$[4].

Theorem 1 requires $x^* = 0 \in S$ to be a stable equilibrium point. From (2) and (6), the goal function evaluated at the desired point is $\gamma_i|_{q_{di}} = 0$, and $\nabla_{q_i}\gamma_i|_{q_{di}} = 0$ from (7), which can be used to conclude that $\nabla_{q_i}\varphi_i|_{q_{di}} = 0$ from (13). Thus, the desired point q_{di} in the workspace $\mathcal{F}$ is a critical point of φ_i. Using the facts that $\gamma_i|_{q_{di}} = 0$ and $\nabla_{q_i}\gamma_i|_{q_{di}} = 0$ and the Hessian of γ_i is $\nabla^2_{q_i}\gamma_i = 2\zeta_i I_2$ from (8), the Hessian of φ_i evaluated at q_{di} is given by $\nabla^2_{q_i}\varphi_i\big|_{q_{di}} = 2\beta_i^{-\frac{1}{\alpha}} I_2 \zeta_i$. The constraint function $\beta_i > 0$ at the desired configuration by Assumption 2, and ζ_i is a positive number. Hence, the Hessian of φ_i evaluated at that point is positive definite. The navigation function φ_i is minimized at q_{di}.

Proposition 2: The closed-loop kinematics of system (1) with the controller (12) are given by $\dot{\mathbf{q}} = f(\mathbf{q})$, where $\mathbf{q}$ denotes the stacked states of each node, $\mathbf{q} = \left[q_1^T \cdots q_N^T\right]^T$ and $f(\mathbf{q}) = \left[f_1^T \cdots f_N^T\right]$ with $f_i^T = -K_i \nabla_{q_i}\varphi_i$ for $\forall i \in \mathcal{N}$. Consider the system $\dot{\mathbf{q}} = f(\mathbf{q})$ for $\forall i \in \mathcal{N}$, and a density function as $\rho = -\varphi$, where $\varphi = \sum_{i=1}^N \varphi_i$ in Theorem 1. If there exists an $\varepsilon' > 0$ such that (14) is satisfied, as long as $\alpha > \max\{1, \Gamma(\varepsilon), \varepsilon'\}$ at any saddle points, where α is a running parameter in the navigation function (5), the undesired critical points are sets of measure zero from Theorem 1.

Proof: Note that ρ is defined for all points in the workspace other than the desired equilibrium q_{di}, and each φ_i is C^2 and takes a value in $[0, 1]$. Thus both the function φ and its gradient are bounded functions in the workspace, which indicates that the integrability condition in Theorem 1 is fulfilled. From the divergence criterion,

$$\nabla \cdot (f\rho) = (\nabla\rho)^T f + \rho\nabla \cdot (f),$$

and from the definition of a critical point, $\nabla_{q_i}\varphi_i = 0$. Hence, $f_i^T = -K_i\nabla_{q_i}\varphi_i = 0$ for $\forall i \in \mathcal{N}$, which indicates that $f = 0$, and $\nabla \cdot (f\rho)$ can be simplified as

$$\nabla \cdot (f\rho) = \varphi \sum_{i=1}^N K_i \left(\frac{\partial^2\varphi_i}{\partial x_i^2} + \frac{\partial^2\varphi_i}{\partial y_i^2}\right). \tag{15}$$

Since φ are positive at undesired critical points from (5), and K_i is a positive gain, a sufficient condition for (15) to be strictly positive is $\frac{\partial^2\varphi_i}{\partial x_i^2} + \frac{\partial^2\varphi_i}{\partial y_i^2} > 0$. Using (13), $\frac{\partial^2\varphi_i}{\partial x_i^2}$ and $\frac{\partial^2\varphi_i}{\partial y_i^2}$ are computed as

$$\frac{\partial^2\varphi_i}{\partial x_i^2} = \frac{\left(\frac{\partial\beta_i}{\partial x_i}\frac{\partial\gamma_i}{\partial x_i} + \beta_i\frac{\partial^2\gamma_i}{\partial x_i^2} - \frac{1}{\alpha}\frac{\partial\beta_i}{\partial x_i}\frac{\partial\gamma_i}{\partial x_i} - \frac{\gamma_i}{\alpha}\frac{\partial^2\beta_i}{\partial x_i^2}\right)}{(\gamma_i^\alpha + \beta_i)^{\frac{1}{\alpha}+1}} \tag{16}$$

$$\frac{\partial^2\varphi_i}{\partial y_i^2} = \frac{\left(\frac{\partial\beta_i}{\partial y_i}\frac{\partial\gamma_i}{\partial y_i} + \beta\frac{\partial^2\gamma_i}{\partial y_i^2} - \frac{1}{\alpha}\frac{\partial\beta_i}{\partial y_i}\frac{\partial\gamma_i}{\partial y_i} - \frac{\gamma_i}{\alpha}\frac{\partial^2\beta_i}{\partial y_i^2}\right)}{(\gamma_i^\alpha + \beta_i)^{\frac{1}{\alpha}+1}}. \tag{17}$$

Observing that $\frac{\partial^2\varphi_i}{\partial x_i^2}$ and $\frac{\partial^2\varphi_i}{\partial y_i^2}$ has similar structure, it suffices to show that $\frac{\partial^2\varphi_i}{\partial x_i^2} > 0$ for $\forall i \in \mathcal{N}$, since the same results can be derived for $\frac{\partial^2\varphi_i}{\partial y_i^2}$. Since γ_i and β_i are positive from (6) and (9), and can not be zero simultaneously from Assumption 2, the positivity of (16) can be proven by showing that the numerator of the left side of (16) is positive. Using the fact that $\frac{\partial\beta_i}{\partial x_i} = \frac{\alpha\beta_i}{\gamma}\frac{\partial\gamma_i}{\partial x_i}$ at a critical point, the following expression can be obtained from (16) as

$$C_1\alpha^2 + C_2\alpha + C_3 > 0. \tag{18}$$

where, $C_1 = \frac{\beta_i}{\gamma_i}\left(\frac{\partial\gamma_i}{\partial x_i}\right)^2$, $C_2 = \frac{\beta_i}{\gamma_i}\left(\frac{\gamma_i\partial^2\gamma_i}{\partial x_i^2} - \left(\frac{\partial\gamma_i}{\partial x_i}\right)^2\right)$ and $C_3 = -\frac{\gamma_i\partial^2\beta_i}{\partial x_i^2}$. Note that $\beta_i = 0$ indicates φ_i achieves its maximum from (5). However, since the set of initial conditions is open, and no open set of initial conditions can be attracted to the maxima of φ_i along the negative gradient motion $-K_i\nabla_{q_i}\varphi_i$ [14], then $\beta_i \neq 0$. In addition, γ_i is evaluated at the undesired critical points (i.e., except the q_{di}), so $\gamma_i \neq 0$ and $\frac{\partial\gamma_i}{\partial x_i} \neq 0$ from (6) and (7). To satisfy the condition in (18), two cases are considered for

$$C_1\alpha^2 + C_2\alpha + C_3 = 0. \tag{19}$$

[4]For a function $f : \mathbb{R}^n \to \mathbb{R}^n$, the notation of divergence is defined as $\nabla \cdot f = \frac{\partial f_1}{\partial x_1} + \cdots + \frac{\partial f_n}{\partial x_n}$.

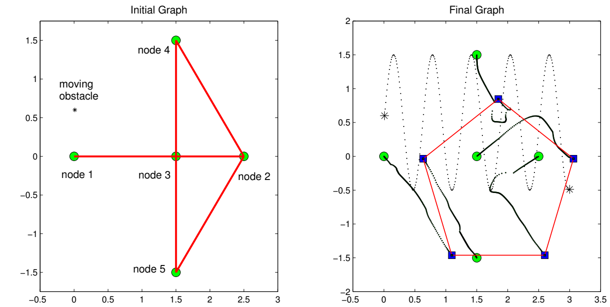

Fig. 2. The plot of initial and final graph, and the trajectory evolution of each node.

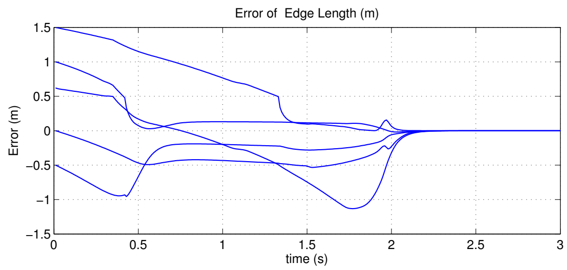

Fig. 3. The error plot of the inter-agent distance during the evolution.

Case 1: there are no solution of α for (19). Since $\frac{\beta_i}{\gamma_i}\left(\frac{\partial \gamma_i}{\partial x_i}\right)^2 >$ 0, which means α can be arbitrary value. Note that α is a positive gain in (5). Hence, as long as $\alpha > 0$, the condition in (18) is valid in Case 1. Case 2: there are solution of α for (19), and assume the two solutions are S_1 and S_2. In this case, the condition in (18) is satisfied as long as $\alpha > \max\{S_1,\ S_2,\ 0\}$. Combining Case 1 and Case 2, we can finally conclude that if $\alpha > \max\{1, \Gamma(\varepsilon), \varepsilon'\}$, where ε' is defined as $\varepsilon' = \max\{S_1,\ S_2,\ 0\}$, all saddle points are measure zero, and the system will only converge to the desired configuration. ■

V. Simulation

A group of 5 nodes with kinematics given in (1) are distributed in a workspace of $R = 100\ m$. Each node is assumed to have a limited communication and sensing zone of $R_c = 2\ m$ and $\delta_1 = \delta_2 = 0.4\ m$. The desired configuration is characterized as a regular pentagon. The initially connected graph and the trajectory evolution for each node are shown in Fig. 2 respectively. In Fig. 2, the '*' represents a moving obstacle, following a trajectory of $\sin(10t)$ from an initial position $(0, 0.5)$ and the solid lines indicate the communication links between connected nodes. As shown in the plot of Fig. 3, the system converges to the desired configuration.

VI. Conclusion

The proposed decentralized controller guarantees that a multi-agent system with limited sensing and communication capabilities will converge to the desired configuration from any given initially connected graph without disconnecting the underlying network graph, as well as prevent the collision among agents and moving obstacles during the evolution. Since the individual edge robustness can be overestimated when using the addition of edge robustness as an indicator, future work will include improving this result by taking into account the freedom of motion for each agent.

References

[1] R. Sepulchre, D. A. Paley, and N. E. Leonard, "Stabilization of planar collective motion: All-to-all communication," *IEEE Trans. Autom. Control*, vol. 52, no. 5, pp. 811 –824, May 2007.

[2] R. Sepulchre, D. Paley, and N. Leonard, "Stabilization of planar collective motion with limited communication," *IEEE Trans. Autom. Control*, vol. 53, no. 3, pp. 706 –719, Apr 2008.

[3] E. Rimon and D. Koditschek, "Exact robot navigation using artificial potential functions," *IEEE Trans. Robot. Autom.*, vol. 8, no. 5, pp. 501–518, Oct 1992.

[4] D. E. Koditschek and E. Rimon, "Robot navigation functions on manifolds with boundary," *Adv. Appl. Math.*, vol. 11, pp. 412–442, Dec 1990.

[5] S. Loizou and K. Kyriakopoulos, "Closed loop navigation for multiple holonomic vehicles," in *Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, vol. 3, 2002, pp. 2861 – 2866.

[6] D. Dimarogonas and K. Kyriakopoulos, "Decentralized stabilization and collision avoidance of multiple air vehicles with limited sensing capabilities," in *Proc. Am. Control Conf.*, June 2005, pp. 4667 – 4672 vol. 7.

[7] J. Chen, W. E. Dixon, M. Dawson, and M. McIntyre, "Homography-based visual servo tracking control of a wheeled mobile robot," *IEEE Trans. Robot.*, vol. 22, no. 2, pp. 406–415, 2006.

[8] D. Dimarogonas, M. Zavlanos, S. Loizou, and K. Kyriakopoulos, "Decentralized motion control of multiple holonomic agents under input constraints," in *Proc. IEEE Conf. Decis. Control*, vol. 4, Dec. 2003, pp. 3390 – 3395.

[9] D. Dimarogonas and K. Johansson, "Decentralized connectivity maintenance in mobile networks with bounded inputs," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2008, pp. 1507–1512.

[10] M. Zavlanos and G. Pappas, "Distributed connectivity control of mobile networks," *IEEE Trans. Robot.*, vol. 24, no. 6, pp. 1416–1428, Dec. 2008.

[11] H. Tanner and A. Kumar, "Towards decentralization of multi-robot navigation functions," in *Proc. IEEE Int. Conf. Robot. Autom.*, April 2005, pp. 4132 – 4137.

[12] M. De Gennaro and A. Jadbabaie, "Formation control for a cooperative multi-agent system using decentralized navigation functions," in *Proc. Am. Control Conf.*, June 2006, pp. 1346–1351.

[13] Z. Kan, A. Dani, J. M. Shea, and W. E. Dixon, "Ensuring network connectivity during formation control using a decentralized navigation function," in *Proc. IEEE Mil. Commun. Conf.*, San Jose, CA, 2010, pp. 954–959.

[14] D. Dimarogonas and K. Johansson, "Bounded control of network connectivity in multi-agent systems," *Control Theory Applications, IET*, vol. 4, no. 8, pp. 1330 –1338, Aug. 2010.

[15] D. Spanos and R. Murray, "Robust connectivity of networked vehicles," in *Proc. IEEE Conf. Decis. Control*, vol. 3, Dec. 2004, pp. 2893–2898.

[16] ——, "Motion planning with wireless network constraints," in *Proc. Am. Control Conf.*, Jun. 2005, pp. 87 – 92.

[17] A. Rantzer, "A dual to lyapunov's stability theorem," *Systems & Control Letters*, vol. 42, no. 3, pp. 161 – 168, 2001. [Online]. Available: http://www.sciencedirect.com/science/article/B6V4X-42BSJB9-1/2/afc4d7b165898322d9410de642da3f56

[18] X. Tang and S. Chanson, "The minimal cost distribution tree problem for recursive expiration-based consistency management," *IEEE T. Parall. Distr.*, vol. 15, no. 3, pp. 214 – 227, Mar. 2004.

[19] H. G. Tanner, A. Jadbabaie, and G. J. Pappas, "Flocking in fixed and switching networks," *IEEE Trans. Autom. Control*, vol. 52, no. 5, pp. 863–868, May 2007.

[20] D. Dimarogonas and K. Johansson, "Analysis of robot navigation schemes using Rantzer dual Lyapunov theorem," in *Proc. Am. Control Conf.*, June 2008, pp. 201 –206.