2010 IEEE International Symposium on Intelligent Control
Part of 2010 IEEE Multi-Conference on Systems and Control
Yokohama, Japan, September 8-10, 2010

Vision Based Connectivity Maintenance of a Network with Switching Topology

Z. Kan, S. Subramanian, J.M. Shea, and W. E. Dixon

Abstract— **Networks of cooperating agents are investigated for formation and coverage control, target tracking, flocking, and consensus applications. Within these applications, agents are required to coordinate to make appropriate decisions and achieve desired goals; hence, the ability to maintain connectivity with other agents is paramount. In this paper, we propose a two level control framework for connectivity maintenance and cooperation of multi-agent systems. Each agent is equipped with an omnidirectional camera and wireless communication capabilities. Image feedback is the primary method to maintain connectivity among agents with wireless communication that is only used to broadcast information when a specific topology change occurs. All agents in the team are categorized as clusterheads or regular nodes. A high level graph is composed of all clusterheads and the motion of the clusterheads is controlled to maintain existing connections among them. A low level graph composed of all regular nodes is controlled to maintain connectivity with respect its specific clusterhead.**

I. INTRODUCTION

Multi-agent cooperative control problems arise from applications such as formation and coverage control, target tracking, flocking, consensus and mobile sensor networks. In most of these applications, agents need to coordinate and communicate to take appropriate decisions to fulfill a pre-specified goal. Two moving agents can communicate with each other if they remain inside a specific distance. As agents move to perform some mission objective, ensuring that the group of agents remain connected (i.e., the group does not partition) can be challenging.

A. Previous Work

Motivated by the practical need to keep agents in a single group, previous literature has focused on the network connectivity maintenance problem (e.g., [1]–[5]). An artificial potential field is one approach that has been widely used in path planning for multi-agent systems to ensure network connectivity and collision avoidance. For example, Zavlanos et al. [6] modeled connectivity as an imaginary obstacle and use artificial potential fields to avoid collision. This centralized feedback control approach uses information from the Laplacian matrix to ensure connectivity. That is, the method exploits the fact that is the second smallest eigenvalue of the Laplacian matrix is positive (i.e., *algebraic connectivity*) then the connectivity of the underlying network graph is ensured [1], [5], [7]. However, global information of the underlying graph is required to compute the graph Laplacian. A distributed control strategy was developed in [8], where connectivity control is performed in the discrete space of graphs to verify link deletions with respect to connectivity, and motion control is performed in the continuous configuration space using a potential field. In [9], a neighbor control law based on potential fields, is designed to achieve velocity alignment and network connectivity among different network topologies. In [10], a repulsive potential is used for a collision avoidance objective, and an attractive potential field is used to drive agents close to each other. Li et al. introduced an artificial potential based control strategy for connectivity guaranteed trajectory tracking of multi-agent systems with the feature that no initially connected graph is required [11].

Other research closely related to network connectivity maintenance includes [12], [13]. The idea of a communication backbone, which has been extensively used in many aspects of wireless networks was used in [12] along with artificial potential methods to ensure network connectivity while preforming distributed tasks for a multi-agent path planning problem. Two hop information is required to update the backbone at each instant. Efforts in [12] were extended in [13] to address the local minima escape problem while maintaining connectedness among mobile agents.

B. Contributions

A common feature in the aforementioned literature is the assumption of wireless communication among agents within either a centralized or decentralized manner. However, such connections may not be applicable in some dynamic, hostile, or tactical environments, and even when radio communication is possible the network bandwidth may be required for distributing other data. Although the connectivity maintenance problem is widely studied, few efforts use image feedback to maintain the connectivity of the underlying network. Motivated to reduce interagent radio communication, a network connectivity maintenance objective is considered in this paper that relies primarily on image feedback. Previous image-based distributed control strategies assume that robots are capable of communicating with their neighbors [14], [15] while other works [16], [17] focus on image-based formation control in a leader follower manner.

[1]This research is supported in part by the NSF CAREER award number 0547448, NSF award number 0901491, the Department of Energy, grant number DE-FG04-86NE37967 as part of the DOE University Research Program in Robotics (URPR), ASTREC, and NSF I/UCRC.

[2]Z. Kan and W. E. Dixon are with the Department of Mechanical and Aerospace Engineering, University of Florida, Gainesville, FL 32611-6250, USA. W. E. Dixon is also with the Department of Electrical and Computer Engineering, University of Florida, Gainesville, FL 32611-6250, USA. Email: {kanzhen0322, wdixon}@ufl.edu

[3]S. Subramanian and J.M. Shea are with the Department of Electrial and Computer Engineering, University of Florida, Gainesville, FL 32611-6250, USA. Email: sankrith@ufl.edu, jshea@ece.ufl.edu

This paper addresses the network connectivity maintenance problem by using image feedback under two basic assumptions. Each robot is equipped with an omnidirectional camera and associated image processing software such that it can measure the relative position and orientation of the other agents in its sensing area. Each agent is also equipped with some form of transceiver that can be used to broadcast information to local nodes. The goal is to use image feedback to maintain the connectivity of the network unless a clusterhead change is required. If the clusterhead changes and the network needs to reorganize the topology, only then is the wireless communication used to alert the nodes of the topology change.

Another distinguishing feature of the current work is the use of a two level control strategy to guarantee network connectivity for a multi-agent system with image feedback. A challenge to achieve this result lies in the fact that connectedness is a global quality of the network, but distributed motion planning must rely on locally available image information. Also, there is no communication between different nodes which means each agent has no path to obtain information outside its sensing region, which greatly increase the difficulty to design a decentralized controller for each agent to guarantee the whole graph remains connected. Specific methods are also developed to drive a group of agents from one connected configuration to another without disconnecting the network in process.

II. Problem Formulation

Consider a network composed of N agents, where agent i moves according to the following kinematics:

$$\dot{x}_i(t) = u_i(t),\ i \in \mathcal{V} = \{1, \ldots, N\} \tag{1}$$

where $x_i(t) \in \mathbb{R}^2$ denotes the position of agent i in a two dimensional (2D) plane at time t, and $\mathbf{x}(t) \in \mathbb{R}^{2N}$ denotes the stack position vectors of all agents. In (1), $u_i(t) \in \mathbb{R}^2$ denotes the velocity of agent i (i.e. the control input). The interaction of the group is modeled as a *dynamic graph*, in the sense that it evolves in time with its connectivity governed by the kinematics of the agents (1). This time varying property gives rise to the notion of a dynamic graph, $\mathcal{G}(t) = (\mathcal{V}, \varepsilon(t))$, in which the set of links $\varepsilon(t)$ is time varying and each component in $\mathcal{V}$ stands for the index of an agent. Given the assumption that each agent is equipped with an omnidirectional camera and wireless communication capabilities, two different graph models need to be specified: a communication graph and a visibility graph. Each graph is composed of different types of nodes: clusterheads and regular nodes, and the interaction between the nodes in each graph is modeled in a different way.

A. Communication Graph

Inter-agent communication is modeled in terms of a time-varying communication graph $\mathcal{G}_c$= $(\mathcal{V}, \varepsilon_c(t))$ with the index set of nodes $\mathcal{V}$ and set of edges

$$\varepsilon_c = \{(i,j) \in \mathcal{V} \times \mathcal{V} |\, \|x_{ij}\| \leq R_c\}\,. \tag{2}$$

In (2), each node is located at a position x_i, $\|x_{ij}\| \in \mathbb{R}^+$ is defined as

$$\|x_{ij}\| = \|x_i - x_j\|\,, \tag{3}$$

and R_c denotes the maximum communication radius. The communication graph $\mathcal{G}_c$ is an undirected graph in the sense that nodes i can influence node j and vice versa. An undirected communication link between nodes i and j is denoted by (i,j) when $\|x_{ij}\| \leq R_c$. The index set of neighbors of node i is denoted by $\mathcal{N}_i^c = \{j :\ j \neq i | j \in \mathcal{V}, (i,j) \in \varepsilon_c\}$.

B. Visibility Graph

Each agent is capable of sensing a disk area with the maximum radius $R_v \leq R_c$, so that any two agents are able to communicate with each other as long as they can see each other. The visibility graph is modeled as a directed time-varying graph $\mathcal{G}_v = (\mathcal{V}, \varepsilon_v(t))$ with the index set of nodes $\mathcal{V}$ and set of edges

$$\varepsilon_v = \{(i,j) \in \mathcal{V} \times \mathcal{V} |\, \|x_{ij}\| \leq R_v\}\,.$$

For the visibility graph, the edge (i,j) is directed indicating node i can influence node j, but not vice versa. The index set of neighbors of node i is denoted by $\mathcal{N}_i^v = \{j :\ j \neq i | j \in \mathcal{V}, (i,j) \in \varepsilon_v\}$. The subsequent development is based on the assumption that the distance between two nodes can be estimated from the image feedback (e.g., using methods as in [18]).

C. Connectivity Maintenance

Since $R_v \leq R_c$, a sufficient goal to ensure $\mathcal{G}_c$ remains connected is to ensure the visibility graph $\mathcal{G}_v$ remains connected. For simplicity, the following development is based on the assumption that $R_v = R_c = R$ without loss of generality. To understand connectivity for each graph, consider Fig 1. For the communication graph, if node i in Fig 1 is connected to node j and node j is connected to node k, then node i is also connected to node k through edge (i,j) and (j,k). Node i and node k may exchange information in $\mathcal{G}_c$, to achieve a desired cooperative motion. If Fig 1 is considered as a visibility graph, then although node j can be seen by node i and node k can be seen by node j, node i is not capable of sharing information with node j. The communication graph is considered connected if every node in $\mathcal{G}_c$ is reachable from every other node by a series of edges.

The goal in this paper is to develop a decentralized image-feedback controller (i.e., velocity input) for each agent so that $\mathcal{G}_c$ remains connected despite clusterhead shifts (i.e., when a clusterhead role shifts from one node to a regular node). The advantage is that the network maintenance is achieved without radio communication, except when a clusterhead shift needs to occur. When the topology changes due to a clusterhead shift, the new role of node is broadcast across the wireless network.

III. Control Strategy

Motivated by the idea of a communication backbone [12], [19], a two level network structure is proposed. The basic idea is to group all nodes into m subsets. Each subset

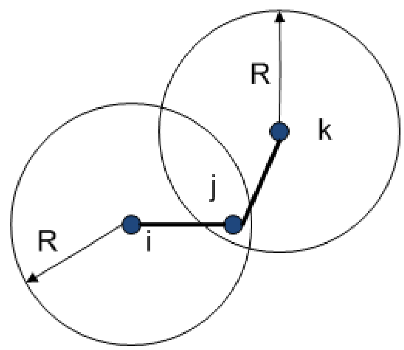

Fig. 1. Model of visibility graph

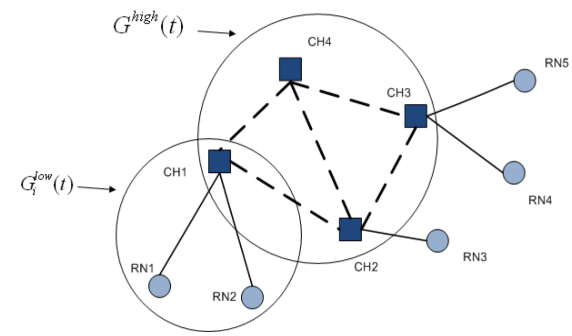

Fig. 2. Schematic topology of underlying network

contains a one (and only one) special node defined as *clusterhead,* where $\mathcal{V}_{CH}=\{1,\cdots,m\}$ denotes the index set of clusterheads, and the set of clusterheads forms a high level network graph, represented as $\mathcal{G}^{high}(t)$. Specifically, the high level network subgraph is composed of clusterheads only, which is a small subset of the group, providing a hierarchical organization of the original network. The high level network subgraph is defined as $\mathcal{G}^{high}(t) = (\mathcal{V}_{CH}, \varepsilon^{high})$, where $\varepsilon^{high}=\{(i,j) \in \mathcal{V}_{CH}\times\mathcal{V}_{CH}| \,\|x_{ij}\| \leq R\}$.

All the remaining nodes in each subgraph are defined as a *regular nodes,* where $\mathcal{V}_{RN}=\{m+1,\cdots,N\}$ denotes the index set of regular nodes. The m subsets form the low level network, represented as $\mathcal{G}_i^{low}(t)$. Specifically, the low level network subgraph is defined as $\mathcal{G}^{low}(t) = \left\{\mathcal{G}_1^{low},\cdots,\mathcal{G}_m^{low}\right\}$. Each $\mathcal{G}_i^{low}(t)$ forms a connected subgraph of $\mathcal{G}_v(t)$ and only one particular node is selected as a clusterhead in each $\mathcal{G}_i^{low}(t)$. Note that $\cap_{i=1}^{m}\mathcal{G}_i^{low}(t) = 0$,which means $\mathcal{G}_i^{low}(t)$ is mutually exclusive to each other, and $\cup_{i=1}^{m}\mathcal{G}_i^{low}(t) = \mathcal{G}_v(t)$. Since only local information can be obtained by vision sensors, we require that the selected clusterhead can be seen by all regular nodes in each $\mathcal{G}_i^{low}(t)$, and each regular node in $\mathcal{G}_i^{low}(t)$ moves under the constraint that it must stay connected to its clusterhead for all time. Hence, each low level network subgraph $\mathcal{G}_i^{low}(t)$ has a fixed topology. This two graph structure is depicted in Fig. 2, where CH stands for clusterhead and RN stands for regular node. As indicated in Fig. 2, {CH1, CH2, CH3, CH4} forms the high level network subgraph $\mathcal{G}^{high}(t)$, while {{CH1, RN1, RN2}, {CH2, RN3}, {CH3, RN5, RN4}, {CH4}} forms the low level network graph $\mathcal{G}^{low}(t)$.

The key to maintain the network connectivity is to maintain connectivity within each subset (i.e., ensure each $\mathcal{G}_i^{low}(t)$ is individually connected) and maintain connectivity of the $\mathcal{G}^{high}(t)$ graph. The graphs $\mathcal{G}_i^{low}(t)$ and $\mathcal{G}^{high}(t)$ are initially specified, but events can occur that require a clusterhead to change roles with a regular node in $\mathcal{G}_i^{low}(t)$. Information-driven methods such as those described in [20] and [21] can be used to dynamically select clusterheads for different tasks. The development in this paper simply assumes that some process determines the need for a clusterhead and regular node to change roles.

From a systems theory perspective, the underlying network graph dynamics are considered to have a transient and steady-state response. A steady-state topology is when the roles of the clusterheads and regular nodes remain constant. The control objective during steady-state is to ensure that all regular nodes move to maintain connectivity with the respective clusterhead within $\mathcal{G}_i^{low}(t)$ and ensure that all clusterheads maintain connectivity within $\mathcal{G}^{high}(t)$. In steady-state, no new edges are formed when one node enters other node's sensing radius. In other words, the underlying graph has a fixed topology in the sense that the edges of $\mathcal{G}_i^{low}(t)$ and $\mathcal{G}^{high}(t)$ do not change, but the relative position of the nodes within the graph can dynamically change. Connectivity during steady-state is maintained by image feedback alone.

A transient topology is when the overall graph switches from one connected configuration to another without disconnecting. The network can become transient when the role of a node is changed and new edges are created under some rule. To guarantee the connectivity during a transient stage, wireless communication has to be used to broadcast the new role of nodes to the neighbors. Once the new roles of the nodes has been broadcast, then all nodes resume steady-state where the nodes use only image feedback.

The topology will become transient due to changes in mission objectives or topology disturbances. For example, the role of RN1 in Fig. 2 may need to change to become a new clusterhead. When the topology undergoes a reconfiguration, a two step strategy is investigated. First, RN1 broadcasts its role-change through immediate neighbors to every node in the group. Radio communication is terminated when all nodes have been updated. Then, under image feedback, the nodes start to form a new connected G^{high} and G_i^{low}. Since no radio communication is allowed, each node only has local information within its sensing region. CH3 needs to move toward CH2 first, and, whenever an edge between CH1 and CH3 is created, it moves to CH1 to get close enough to RN1. Likewise, new edges are created for CH4 and CH2 once they can be seen by RN1. Although new edges are created among clusterheads, there is no edge created for regular nodes, even if some other nodes enter its sensing region. In other words, all regular nodes move with its respective clusterhead. As a result, each subgraph $\mathcal{G}_i^{low}$ can be represented as one single node, represented as a clusterhead. As long as the clusterheads are connected, the whole graph is connected. One benefit of this structure is that network with large number of nodes can be scaled down.

IV. Control Design

A. Potential Field

The goal in this section is to design distributed control laws $u_i(t)$ for all nodes to guarantee the connectivity of $\mathcal{G}_v(t)$. The set $\mathcal{N}_i^v(t)$ is time varying and dependent on the relative positions of the nodes. Nodes within distances less than R are interacting with each other through a potential force and a potential function is used for connectivity maintenance, as well as collision avoidance.

An attractive potential field is defined as $\varphi_{ij} : \mathbb{R}^2 \rightarrow \mathbb{R}^+$, which is a nonnegative function of the distance between nodes i and j, i.e. $\varphi_{ij} = \varphi_{ij}(\|x_{ij}\|)$. The purpose of the attractive force is to guarantee that node j will never leave the sensing zone of node i, if node j is initially located at a distance less than R from node i. The attractive potential field is to regulate distances between agents within the range $(0, R)$. Some properties are required to make φ_{ij} a qualified potential function:

1) $\varphi_{ij}(\|x_{ij}\|) \rightarrow \infty$ as $\|x_{ij}\| \rightarrow R$.

2) $\varphi_{ij}(\|x_{ij}\|)$ is C^1 for $\|x_{ij}\| \in (0, R)$ and $\frac{\partial \varphi_{ij}}{\partial \|x_{ij}\|} > 0$, if $\|x_{ij}\| \in (0, R)$.

3) $\varphi_{ij}(\|x_{ij}\|) = 0$ when $\|x_{ij}\| > R$.

A repulsive potential field is defined as $\psi_{ij} : \mathbb{R}^2 \rightarrow \mathbb{R}^+$, which is a differentiable (except at point $\|x_{ij}\| = 0$), nonnegative function of the distance between nodes i and j, i.e. $\psi_{ij} = \psi_{ij}(\|x_{ij}\|)$.The purpose of the repulsive force is to guarantee collisions avoidance between node i and node j as they get close to each other. Some properties are required to make ψ_{ij} a qualified potential function:

1) $\psi_{ij}(\|x_{ij}\|) \rightarrow \infty$ as $\|x_{ij}\| \rightarrow 0$.

2) $\psi_{ij}(\|x_{ij}\|) = 0$, and $\frac{\partial \psi_{ij}}{\partial x_i} = 0$, if $\|x_{ij}\| \geq R$.

3) $\frac{\partial \psi_{ij}}{\partial \|x_{ij}\|} < 0$, if $\|x_{ij}\| \in (0, R)$, and $\frac{\partial \psi_{ij}}{\partial \|x_{ij}\|} = 0$, if $\|x_{ij}\| \geq R$.

Property 2) guarantees that

$$\sum_{j \in \mathcal{N}_i^v(t)} \frac{\partial \psi_{ij}}{\partial x_i} = \sum_{j \neq i} \frac{\partial \psi_{ij}}{\partial x_i}.$$

The importance of this property is that, unlike an attractive force, the time varying set $\mathcal{N}_i^v(t)$ does not introduce any discontinuity to the system when one node enters the sensing zone of another node. Inspired by [1], [9], [10], a function $\varphi^*_{ij}(\|x_{ij}\|)$ is introduced to smooth the discontinuity when a new edge is formed. To capture the newly formed edge, a set $\mathcal{N}_i^*(t)$ is defined as $\mathcal{N}_i^*(t) = \{j \in \mathcal{V}, j \neq i | \, \|x_{ij}\| \leq R - \epsilon\}$, where $0 < \epsilon \ll R$. The set of edges is updated as: $\varepsilon_v(t) = \varepsilon_v(t^-) \cup \varepsilon_v^*(t)$, where $\varepsilon_v^*(t) = \{(i,j) | ((i,j) \notin \varepsilon_v(t^-)) \wedge (j \in \mathcal{N}_i^*(t)\}$. The function φ^*_{ij} is defined with following properties:

1) $\varphi^*_{ij} = \varphi_{ij}$ if $\|x_{ij}\| \leq R - \epsilon$.

2) $\varphi^*_{ij} = const$ and $\frac{\partial \varphi^*_{ij}}{\partial x_i} = 0$ if $\|x_{ij}\| \geq R$.

3) φ^*_{ij} is C^1 everywhere and the partial derivative $\frac{\partial \varphi^*_{ij}}{\partial \|x_{ij}\|} > 0$ for $R - \epsilon < \|x_{ij}\| < R$.

4) $\varphi^*_{ij}(R - \epsilon) = \varphi_{ij}(R - \epsilon)$ and $\frac{\partial \varphi^*_{ij}(R-\epsilon)}{\partial \|x_{ij}\|} = \frac{\partial \varphi_{ij}(R-\epsilon)}{\partial \|x_{ij}\|}$.

The function switches from φ^*_{ij} to φ_{ij} during a switching state, and a new edge is created whenever a node j is a distance less than $R - \epsilon$ with respect to node i. It seems that φ^*_{ij} is used to capture the potential among node i and nodes outside of its sensing zone, which is a violation of a decentralized approach. Actually, according to Property 2), φ^*_{ij} is carefully designed so that its partial derivative with respect to x_i is 0 when $\|x_{ij}\| \geq R$. The only element that contributes to the controller is $\frac{\partial \varphi^*_{ij}}{\partial \|x_{ij}\|}$, $\|x_{ij}\| \in (R - \epsilon, R)$. Although node j is in the sensing region of node i, no new edge is created. In addition, φ^*_{ij} is C^1 everywhere. Hence, the switch to φ_{ij} is sufficiently smooth when a node j enters the sensing zone of node i.

B. Controller for Steady State

In each subgraph G_i^{low}, a regular node is attracted by its clusterhead only and repelled by all the adjacent nodes. The total potential of regular node $i, i \in V_{RN}$, is:

$$U_i^r = \varphi_{ik} + \sum_{j \in \mathcal{N}_i^v(t)} \psi_{ij}, \tag{4}$$

where k, $k \in V_{CH}$, denotes the index of the corresponding clusterhead in G_i^{low}. The control law for a regular node is designed as

$$u_i^r(t) = -\frac{\partial U_i^r}{\partial x_i}. \tag{5}$$

The motion of a clusterhead is not affected by regular nodes, and a clusterhead only moves with the constraint to ensure connectivity and collision avoidance in G^{high}. The composite potential of clusterhead $i, i \in V_{CH}$, is given by:

$$U_i^c = \sum_{(i,j) \in \varepsilon^{high}} \varphi_{ij} + \sum_{(i,j) \in \varepsilon^{high}} \psi_{ij} + U_i^T, \tag{6}$$

where U_i^T denotes a task potential to model a required performance, which imposes an attractive potential on node i. The control law for the clusterheads is designed as

$$u_i^c(t) = -\frac{\partial U_i^c}{\partial x_i}. \tag{7}$$

C. Controller for Switching State

Collision avoidance and network connectivity must be maintained even when the topology undergoes a transition. As described in Section III, the motion of regular nodes is dictated by the motion of the parent clusterhead. The total potential and control law for regular node i is the same as (4) and (5) in steady state conditions. However, the potential for the clusterhead nodes change. Specifically, the composite potential of clusterhead $i \in V_{CH}$ is given by

$$U_i^c = \sum_{(i,j) \in E(t)} \varphi_{ij} + \sum_{(i,j) \notin E(t)} \varphi^*_{ij} + \sum_{j \neq i} \psi_{ij}, \tag{8}$$

where the set $E(t) \subset \varepsilon^{high}(t)$ denotes a time varying set of edges developed based on the switching strategy in Section

III. The goal of new set $E(t)$ is to guide clusterheads to form a new steady state. Note that there are two main differences between (6) and (8). First, there is no U_i^T in (8). The term U_i^T is designed to perform tasks in steady state. The goal of the switching state is to reshape the topology to a new steady topology. There is no need to keep U_i^T during the switching process. Secondly, the function φ_{ij}^* is used to take care of the discontinuity that is caused by new edge formation. Based on the developed composite potential, the control law for clusterheads is designed as

$$u_i^c(t) = -\frac{\partial U_i^c}{\partial x_i}. \tag{9}$$

An initial connected underlying graph is required to guarantee the connectivity for all the future time.

V. Connectivity Analysis

Proposition 1: For steady state, if the network graph $\mathcal{G}_v(t)$ is connected at $t = t_0$, then connectivity and collision avoidance is guaranteed with the controller proposed in (5) and (7) for $t > t_0$.

Proof: The topology of $\mathcal{G}_v(t)$ is static in steady state in the sense that new edges are not formed. In each subgraph $\mathcal{G}_i^{low}$, regular nodes move with respect to its clusterhead, and in subgraph $\mathcal{G}^{high}$, clusterheads move with the constraint that the connectivity is ensured. A Lyapunov candidate functional is designed as

$$V = \sum_{i\in\mathcal{V}_{RN}} U_i^r + \sum_{i\in\mathcal{V}_{CH}} U_i^c. \tag{10}$$

Based on (4) and (6), as an agent gets close to a collision or as the graph gets closer to partitioning, then $V(\mathbf{x}(t))$ approaches infinity. Taking time derivative of (10) and substituting for (5) and (7), yields

$$\begin{aligned} \dot{V} &= \sum_{i\in\mathcal{V}_{RN}} \frac{\partial U_i^r}{\partial x_i}\dot{x}_i + \sum_{i\in\mathcal{V}_{CH}} \frac{\partial U_i^c}{\partial x_i}\dot{x}_i \qquad (11) \\ &= -\sum_{i\in\mathcal{V}_{RN}} \left\|\frac{\partial U_i^r}{\partial x_i}\right\|^2 - \sum_{i\in\mathcal{V}_{CH}} \left\|\frac{\partial U_i^c}{\partial x_i}\right\|^2 \\ &\leq 0. \end{aligned}$$

The expressions in (10) and (11) imply that $V(\mathbf{x}(t)) \leq V(\mathbf{x}(t_0))$. Since the system is initially collision free and connected at t_0, then $V(\mathbf{x}(t_0)) < \infty$, and the graph is ensured to remain collision free and connected for all $t \geq t_0$ provided the graph topology remains in a steady state condition. ∎

Proposition 2: During the switching process, connectivity and collision avoidance of the network graph $\mathcal{G}(t)$ is guaranteed by the controller proposed in (5) and (9).

Proof: Proposition 1 indicates that connectivity is guaranteed in each $\mathcal{G}_i^{low}$. To show the graph $\mathcal{G}_v(t)$ is connected during a clusterhead switch, we only need to show that once any two clusterheads come into a distance less than or equal to $R-\epsilon$ for the first time, they will remain connected to each other, i.e. the distance between them is strictly less than R for all future time. To examine this scenario, a Lyapunov candidate functional is designed as:

$$V = \sum_{i\in\mathcal{V}_{CH}} \Big(\sum_{(i,j)\in E(t)} \varphi_{ij} + \sum_{(i,j)\notin E(t)} \varphi_{ij}^* + \sum_{i\neq j} \psi_{ij}\Big). \tag{12}$$

An attractive potential function φ_{ij} is a discontinuous function at the point $\|x_{ij}\| = R$ while the repulsive function ψ_{ij} is a differentiable function. Whenever a node j forms a new edge with node i, the function φ_{ij}^* is switched to φ_{ij} in a sufficiently smooth manner, so that V is continuously differentiable. Taking the time derivative of V and substituting (5) and (9) yields $\dot{V} \leq 0$, and hence, $V(\mathbf{x}(t)) \leq V(\mathbf{x}(t_0)) < \infty$. It is known that $V \to \infty$ when $\|x_{ij}\| \to R$ for at least one pair of nodes. Hence, all pairs of nodes that did not initially form an edge move so that new edges are formed so that the communication graph remains connected. ∎

VI. Simulation

Preliminary simulation results illustrate the performance of the proposed control strategy. A group of 7 nodes with kinematics given in (1) are distributed in the plane with an initially connected underlying graph. Assume that each node has a sensing zone of $R = 1$ m. When two agents are adjacent, a line is drawn between them to show the connectivity.

A group of 7 nodes evolved under the control law proposed in Section IV. In Fig 3, the rectangular nodes represent clusterheads and circles represent regular nodes. The dashed lines identify the link among clusterheads, while solid lines identity the link between a regular node and a clusterhead. At $t = 0$, an initial connected graph is generated. During time interval $t \in (0, 120)$, the group of nodes moves in steady state. The topology is maintained during node motion, as shown in Fig 3 at $t = 110$. One clusterhead is simulated with a task function to move along a designed trajectory, $P_y = 2\sin(0.2P_x)$, where P_y and P_x denotes the stack x and y coordinate vector respectively. In the first two subplots of Fig 3, all nodes move in a desired manner. To simulate the performance of a switching state, the topology changes at $t = 121$ in the sense that one regular node switches its role to a new clusterhead, while one clusterhead changes its role to a regular node. The new clusterhead is tasked with the objective to move along the desired trajectory, $P_y = -2\sin(0.2P_x)$, from its current position. The bottom two subplots of Fig 3 illustrate how these nodes move to reshape the topology to form a new steady state topology without disconnecting the group.

VII. Conclusion

A two stage control framework is proposed for connectivity maintenance and cooperation of a multi-agent system using image feedback. The idea is to group all nodes into two subgraphs, a high level network subgraph and several low level network subgraphs. The key to maintain the network connectivity is to ensure the connectivity of high level subgraph and the connectivity of each low level subgraph. A potential-field-based controller is used to guarantee the

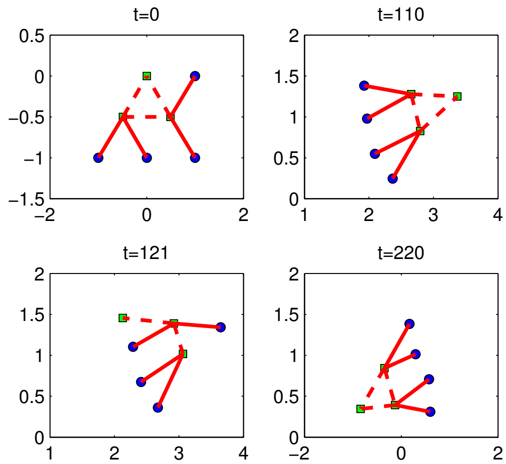

Fig. 3. Evolution of nodes in time.

connectivity, as well as collision avoidance. Future efforts will focus on more simulations with more nodes and more switching to examine the interplay of the nodes, including cases where multiple nodes shift from clusterheads at the same time.

REFERENCES

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

[2] Q. Hui, W. Haddad, and S. Bhat, "Semistability theory for differential inclusions with applications to consensus problems in dynamical networks with switching topology," in *American Control Conference*, June 2008, pp. 3981–3986.

[3] D. V. Dimarogonas and K. J. Kyriakopoulos, "On the rendezvous problem for multiple nonholonomic agents," *IEEE Transactions on Automatic Control*, vol. 52, no. 5, pp. 916–922, May 2007.

[4] D. Dimarogonas and K. Johansson, "Decentralized connectivity maintenance in mobile networks with bounded inputs," in *IEEE International Conference on Robotics and Automation*, May 2008, pp. 1507–1512.

[5] R. Olfati-Saber and R. Murray, "Consensus problems in networks of agents with switching topology and time-delays," *IEEE Transactions on Automatic Control*, vol. 49, no. 9, pp. 1520–1533, Sept. 2004.

[6] M. Zavlanos and G. Pappas, "Potential fields for maintaining connectivity of mobile networks," *IEEE Transactions on Robotics*, vol. 23, no. 4, pp. 812–816, Aug. 2007.

[7] A. Jadbabaie, J. Lin, and A. Morse, "Coordination of groups of mobile autonomous agents using nearest neighbor rules," *IEEE Transactions on Automatic Control*, vol. 48, no. 6, pp. 988–1001, June 2003.

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

[9] M. Zavlanos, A. Jadbabaie, and G. Pappas, "Flocking while preserving network connectivity," in *IEEE Conference on Decision and Control*, Dec. 2007, pp. 2919–2924.

[10] D. Dimarogonas and K. Kyriakopoulos, "Connectivity preserving distributed swarm aggregation for multiple kinematic agents," in *IEEE Conference on Decision and Control*, Dec. 2007, pp. 2913–2918.

[11] Q. Li and Z.-P. Jiang, "Decentralized control strategies for connectivity guaranteed tracking of multi-agent systems," in *World Congress on Intelligent Control and Automation*, June 2008, pp. 323–328.

[12] Z. Yao and K. Gupta, "Backbone-based connectivity control for mobile networks," in *IEEE International Conference on Robotics and Automation*, May 2009, pp. 1133–1139.

[13] ——, "Distributed strategies for local minima escape in motion planning for mobile networks," in *Second International Conference on Robot Communication and Coordination*, April 2009, pp. 1–7.

[14] Z. Zhu, K. Rajasekar, E. Riseman, and A. Hanson, "Panoramic virtual stereo vision of cooperative mobile robots for localizing 3d moving objects," in *IEEE Workshop on Omnidirectional Vision*, 2000, pp. 29–36.

[15] P. Rybski, S. Stoeter, M. Gini, D. Hougen, and N. Papanikolopoulos, "Performance of a distributed robotic system using shared communications channels," *IEEE Transactions on Robotics and Automation*, vol. 18, no. 5, pp. 713–727, Oct 2002.

[16] R. Vidal, O. Shakernia, and S. Sastry, "Formation control of nonholonomic mobile robots with omnidirectional visual servoing and motion segmentation," in *IEEE International Conference on Robotics and Automation*, vol. 1, Sept. 2003, pp. 584–589 vol.1.

[17] A. Das, R. Fierro, V. Kumar, J. Ostrowski, J. Spletzer, and C. Taylor, "A vision-based formation control framework," *IEEE Transactions on Robotics and Automation*, vol. 18, no. 5, pp. 813–825, Oct 2002.

[18] N. Gans, A. Dani, and W. Dixon, "Visual servoing to an arbitrary pose with respect to an object given a single known length," in *American Control Conference*, June 2008, pp. 1261–1267.

[19] L. Bao and J. J. Garcia-Luna-Aceves, "Topology management in ad hoc networks," pp. 129–140, 2003.

[20] F. Zhao, J. Shin, and J. Reich, "Information-driven dynamic sensor collaboration," *Signal Processing Magazine, IEEE*, vol. 19, no. 2, pp. 61–72, Mar 2002.

[21] F. Zhao, J. Liu, J. Liu, L. Guibas, and J. Reich, "Collaborative signal and information processing: an information-directed approach," *Proceedings of the IEEE*, vol. 91, no. 8, pp. 1199–1209, Aug. 2003.