Proceedings of the
44th IEEE Conference on Decision and Control, and
the European Control Conference 2005
Seville, Spain, December 12-15, 2005

Controlling Connectivity of Dynamic Graphs

Michael M. Zavlanos and George J. Pappas

Abstract— **The control of mobile networks of multiple agents raises fundamental and novel problems in controlling the structure of the resulting dynamic graphs. In this paper, we consider the problem of controlling a network of agents so that the resulting motion always preserves various *connectivity* properties. In particular, we consider preserving k-hop connectivity, where agents are allowed to move while maintaining connections to agents that are no more than k-hops away. The connectivity constraint is translated to constrains on individual agent motion by considering the dynamics of the adjacency matrix and related constructs from algebraic graph theory. As special cases, we obtain motion constraints that can preserve the exact structure of the initial dynamic graph, or may simply preserve the usual notion connectivity while the structure of the graph changes over time. We conclude by illustrating various interesting problems that can be achieved while preserving connectivity constraints.**

I. Introduction

Controlling dynamic graphs has recently emerged as a fundamental problem in the area of systems and control theory. Apart from the intellectual challenges associated with it, other motivations come from the area of controlling formations of ground or aerial vehicles with applications in air traffic control, satellite clustering, automatic highways, mobile robotics and mobile sensor networks. One of the main goals in this area is to achieve a coordinated objective while using only relative information concerning positions and velocities. The objective investigated in this paper is that of maintaining various notions of graph connectivity.

Dynamic graphs have not apparently been studied only in the framework proposed in this paper. In [1], a measure of local connectedness of a network, is introduced. This approach is distributed in the sense that this measure depends on neighbor-to-neighbor communication only. Motivated by a class of problems associated with control of distributed dynamic systems is also [2], where the authors consider a controllability framework for state-dependent dynamic graphs. In [3], the problem of finding the graph that corresponds to the maximum second smallest eigenvalue of its Laplacian is investigated. The authors propose a method that searches the graph space towards the direction that maximizes the second smallest eigenvalue of the graph Laplacian, and prove local convergence of their method. The second smallest eigenvalue of the graph Laplacian has also emerged as an important parameter in many system and control problems defined over networks [4], [5], [6]. In fact, in recent works, such as [5], it has been observed that this eigenvalue is a measure of stability and robustness of the networked dynamic system.

This work is partially supported by NSF ITR 0324977, and ARO MURI DAAD 19-02-01-0383.

M. M. Zavlanos and G. J. Pappas are with GRASP Laboratory, Department of Electrical and Systems Engineering, University of Pennsylvania, Philadelphia, PA 19104, USA {zavlanos,pappasg}@grasp.upenn.edu

Other research issues which are closely related to the problems discussed in this paper are formation stabilization [7], [8], [9], [10], consensus seeking by autonomous agents [5], [6], [11], [12], [13], and coverage tasks [14]. The goal in formation stabilization is convergence of the agents to a common velocity. Various approaches have been studied, such as, control laws that involve graph Laplacians for the fixed (or switched) associated neighborhood graphs [8] or Lyapunov function methods such as [9], where the notion of "formation feedback" as a means to regulate agent motion in order to satisfy the global formation constraints, was also introduced. Formation stabilization can also be viewed as a consensus problem. Necessary and sufficient conditions for consensus are investigated in [5], [6], [13]. Consensus can also, under certain conditions, be achieved in the case of switching communication graphs [6], [13].

Motivated by the importance of connectivity in mobile sensor networks as well as the connectivity assumption often made in formation stabilization or consensus problems, in this paper, we consider graph connectivity as our primary objective. Under the assumption that the initial graph is connected, we introduce the notion of k-hop connectivity, and based upon this notion, we develop a centralized control framework that guarantees graph connectivity for all time. The idea is to model connectivity as an invariance problem and transform it into a set of constraints on the control variables. Then, using optimization techniques we are able to compute solutions when the problem is feasible. As a model for connectivity, we use the adjacency matrix of a graph and its dynamics, instead of the Laplacian eigenvalues, since it provides more information about the graph structure. Hence, we consider the problem of designing controllers for the individual agents, so that the resulting graphs remain connected for all time.

The rest of this paper is organized as follows. In Section II we develop a general framework for our problem. In Section III, we relate our framework to the case of k-hop connectivity. We provide graph theoretic and algebraic characterizations for this property and prove that they are equivalent. In Section IV, we deal with the technical issues of our approach. We provide the dynamics of the various quantities that we introduce and propose a solution to the problem of maintaining k-hop connectivity. Finally, in Section V, we state and verify through computer simulations, various connectivity tasks that illustrate the setting we have developed.

II. PROBLEM FORMULATION

A. Graph Theoretic Formulation

Consider n nodes in an m-dimensional space $\mathbb{R}^m$. We denote by $x_i(t) \in \mathbb{R}^m$ the coordinates of the i-th node at time t, where by convention, x_i is considered a $m \times 1$ column vector, and by $\mathbf{x}(t) = [x_1^T(t) \dots x_n^T(t)]^T$ the $mn \times 1$ vector resulting from stacking the coordinates of the nodes into a single vector. Suppose that the dynamics of the i-th node, for all $i \in \{1, 2, \dots, n\}$, are given by, $\dot{x}_i(t) = f_i(\mathbf{x}(t), u_i(t))$ where $u_i(t)$ is the control vector taking values in some set $U \subseteq \mathbb{R}^p$. In vector notation, the system dynamics are given by,

$$\dot{\mathbf{x}}(t) = F(\mathbf{x}(t), \mathbf{u}(t)) \tag{1}$$

where $\dot{\mathbf{x}}(t) = [\dot{x}_1^T(t) \dots \dot{x}_n^T(t)]^T$ and $\mathbf{u}(t) = [u_1^T(t) \dots u_n^T(t)]^T$ are $mn \times 1$ and $pn \times 1$ vectors respectively.

The network of agents described by system (1), gives rise to a *dynamic graph* $\mathcal{G}(t) = \big(\mathcal{V}, \mathcal{E}(t)\big)$, where $\mathcal{V} = \{x_1(t), \dots, x_n(t)\}$ denotes the vertex set of the graph, and $\mathcal{E}(t)$ denotes the time varying edge set, where edges represent pairwise proximity, sensing, or communication relations between the nodes. For example, two distinct vertices $x_i(t)$ and $x_j(t)$ in $\mathcal{G}(t)$ could be connected by an edge if their pairwise distance is within some threshold value related to their sensing capabilities.

Since we have control over node (or vertex) dynamics, the question that naturally arises is whether we can control the motion of the agents, so that $\mathcal{G}(t)$ satisfies a graph theoretic property of interest for all time $t \geq 0$. In particular, in this paper we are interested in whether we can constrain the motion of all agents so that the graph $\mathcal{G}(t)$ *always* lies in some desired set $\mathcal{C}$ of graphs, such as the set of connected graphs. More formally, in this paper, we will address the following problem.

Problem 1 (Graph Theoretic Formulation): Let $\mathcal{C}$ be a desired set of graphs. Given $\mathcal{C}$, determine control constraints $U^*(\mathbf{x}(t))$ so that if $\mathcal{G}(0) \in \mathcal{C}$ and $\mathbf{u}(t) \in U^*(\mathbf{x}(t))$ then $\mathcal{G}(t) \in \mathcal{C}$ for all $t \geq 0$.

In other words, we would like the set $\mathcal{C}$ to be an invariant of motion for system (1). We will achieve this goal by choosing an equivalent formulation using the algebraic representation of the dynamic graph $\mathcal{G}(t)$.

B. Algebraic Formulation

The structure of any graph can be equivalently represented using the adjacency matrix.

Definition 2.1 (Adjacency Matrix): Given a graph $\mathcal{G}$ with vertices $\mathcal{V} = \{v_1, \dots, v_n\}$ and edges in the set $\mathcal{E}$, we define the adjacency matrix of $\mathcal{G}$ to be the matrix $A = (a_{ij})$ such that $a_{ij} = 1$ if $(v_i, v_j) \in \mathcal{E}$ and $a_{ij} = 0$ otherwise. Since we do not allow self-loops, for every $i \in \{1, 2, \dots, n\}$ we define $a_{ii} = 0$.

Note that if A is an adjacency matrix of a graph, then $A = A^T$. In our setting, the dynamic graph $\mathcal{G}(t)$ is a time varying graph (because of the time varying edge set $\mathcal{E}(t)$). This implies that we will be dealing with a time varying adjacency matrix. Let, $A(\mathbf{x}(t)) = \big(a_{ij}(\mathbf{x}(t))\big)$ denote the adjacency matrix corresponding to the graph $\mathcal{G}(t)$, where the entries $a_{ij}(\mathbf{x}(t))$ are functions of $\mathbf{x}(t)$, such that the structure of $A(\mathbf{x}(t))$ is consistent with Definition 2.1[1].

In order to translate Problem 1 in state-space, we must consider algebraic characterizations of the set $\mathcal{C}$ of desired graphs. Let $\mathcal{A}_\mathcal{C}$ denote the set of all adjacency matrices A whose corresponding graphs belong to the desired set $\mathcal{C}$. We will assume that $\mathcal{A}_\mathcal{C}$ can be captured by a mapping characterizing the property of interest.

Definition 2.2: There exists a function $p(\cdot)$ such that the set $\mathcal{A}_\mathcal{C}$ can be defined as $\mathcal{A}_\mathcal{C} = \big\{A \mid p(A) = 0\big\}$, where $p(\cdot)$ might also depend on the initial conditions $A(\mathbf{x}(0))$.

Therefore $A_1, A_2 \in \mathcal{A}_\mathcal{C}$ if and only if $p(A_1) = p(A_2)$. Let,

$$\mathcal{X}_\mathcal{C} = \big\{\mathbf{x}(t) \ \mid \ p\big(A(\mathbf{x}(t))\big) = 0\big\}$$

where now $\mathcal{X}_\mathcal{C}$ is the set of all states $\mathbf{x}(t)$ whose corresponding graphs $\mathcal{G}(t)$ belong to the desired set $\mathcal{C}$. Clearly, $A(\mathbf{x}(t)) \in \mathcal{A}_\mathcal{C}$ if and only if $\mathbf{x}(t) \in \mathcal{X}_\mathcal{C}$. Thus, given a set of graphs $\mathcal{C}$, the mapping $p(\cdot)$ enables us to consider the following algebraic reformulation of Problem 1.

Problem 2 (Algebraic Formulation): Consider the desired graph set $\mathcal{C}$ and let, $\mathcal{X}_\mathcal{C} = \big\{\mathbf{x}(t) \mid p\big(A(\mathbf{x}(t))\big) = 0\big\}$ be the corresponding state-space. Given $\mathcal{X}_\mathcal{C}$, determine control constraints $U^*(\mathbf{x}(t))$ so that if $\mathbf{x}(0) \in \mathcal{X}_\mathcal{C}$ and $\mathbf{u}(t) \in U^*(\mathbf{x}(t))$ then $p(A(\mathbf{x}(t))) = p(A(\mathbf{x}(0)))$ or equivalently $\mathbf{x}(t) \in \mathcal{X}_\mathcal{C}$, for all $t \geq 0$.

In this general framework, the only assumption we impose on the function $p(\cdot)$, besides that it is appropriately chosen so that both the graph theoretic and algebraic formulations are equivalent, is that the resulting state-space $\mathcal{X}_\mathcal{C}$ is connected[2]. Problem 2 requires that we determine constraints for the evolution of system (1) so that a desired state-space $\mathcal{X}_\mathcal{C}$ remains invariant for all time. The latter, connectedness, assumption is necessary for the invariance of $\mathcal{X}_\mathcal{C}$ to be meaningful.

The algebraic reformulation of the main goal of this paper is much more amenable to control theoretic analysis. The main challenge is finding such functions capturing desired graph properties, and rendering them invariant by appropriately constraining the motion of the nodes. As long as the set $U \cap U^*(\mathbf{x}(t))$ is nonempty for all $t \geq 0$, we can guarantee that by choosing $\mathbf{u}(t) \in U \cap U^*(\mathbf{x}(t))$ the dynamic graph $\mathcal{G}(t)$ will always belong in $\mathcal{C}$. We are therefore transforming a constraint on graphs ($\mathcal{G}(t) \in \mathcal{C}$) into a set of constraints on the control inputs $\mathbf{u}$.

However, our approach poses two main challenges that we must address. First, we need to find appropriate representations $p(\cdot)$ of the graph properties of interest, and second we should be able to compute the dynamics of these functions which are necessary for the desired invariance properties. In

[1] In the following sections, we will explicitly define the functions we will be using.

[2] The proof of this condition, which due to space limitations we omit, is based on a particular choice of the function $p(\cdot)$, which we define later.

the rest of this paper we will focus on the connectivity property of a graph. We will propose a representation function $p(\cdot)$ and then show that we can actually compute an input set $U^*(\mathbf{x}(t))$ such that if $\mathbf{u}(t) \in U^*(\mathbf{x}(t))$ the network of nodes will remain connected for all time.

III. Modeling of Connectivity

A. Graph Theoretic Model for Connectivity

Let $\mathcal{G}(\mathbf{x})$ be a dynamic graph on n nodes[3], as described in Section II. We say that two nodes i and j in $\mathcal{G}(\mathbf{x})$ are connected by a *path* of length r if there exists a sequence of $r+1$ distinct nodes starting with i and ending with j such that consecutive nodes are adjacent. Let l_{ij} denote the length of the minimum length path from node i to node j. We define the *k-hop neighborhood* of node i corresponding to the graph $\mathcal{G}(\mathbf{x})$, to be the set,

$$\mathcal{N}_k^{(i)}(\mathbf{x}) = \{j \mid l_{ij} \leq k\}$$

and denote the collection of all k-hop neighborhoods corresponding to the graph $\mathcal{G}(\mathbf{x})$, by $\mathcal{N}_k(\mathbf{x}) = \{\mathcal{N}_k^{(1)}(\mathbf{x}), \ldots, \mathcal{N}_k^{(n)}(\mathbf{x})\}$. Consider a reference graph $\mathcal{G}(\mathbf{x}_0)$ at $\mathbf{x}_0$, and denote by $\mathcal{N}_k(\mathbf{x}_0)$ the set of k-hop neighborhoods corresponding to that graph. Let $\mathcal{R}_k(\mathbf{x}_0)$ be the set of all graphs $\mathcal{G}(\mathbf{x})$ that share the same k-hop neighborhood set with $\mathcal{G}(\mathbf{x}_0)$, i.e.,

$$\mathcal{R}_k(\mathbf{x}_0) = \{\mathcal{G}(\mathbf{x}) \mid \mathcal{N}_k(\mathbf{x}) = \mathcal{N}_k(\mathbf{x}_0)\} \tag{2}$$

and denote by $\mathcal{C}$ the set of all connected graphs. In the rest of this paper we will be interested in the graphs belonging to the set $\mathcal{C}_k(\mathbf{x}_0) = \mathcal{C} \cap \mathcal{R}_k(\mathbf{x}_0)$.

Definition 3.1: We say that a graph $\mathcal{G}(\mathbf{x})$ is *k-hop connected* with respect to the k-hop neighborhood $\mathcal{N}_k(\mathbf{x}_0)$ if and only if $\mathcal{G}(\mathbf{x}) \in \mathcal{C}_k(\mathbf{x}_0)$.

We will call the property associated with k-hop connected graphs, *k-hop connectivity*. It is clear that for $k = 1$, $\mathcal{C}_k(\mathbf{x}_0) = \{\mathcal{G}(\mathbf{x}_0)\}$ if $\mathcal{G}(\mathbf{x}_0)$ is connected and $\mathcal{C}_k(\mathbf{x}_0) = \emptyset$ otherwise. On the other hand, for $k = n-1$, $\mathcal{C}_k(\mathbf{x}_0) = \mathcal{C}$. Observe that in this case, for every $i \in \mathcal{V}$, and every possible configuration $\mathbf{x}$, $\mathcal{N}_{n-1}^{(i)}(\mathbf{x}) \cup \{i\} = \mathcal{V}$. Hence, the condition in (2) is an identity which implies that $\mathcal{R}_k(\mathbf{x}_0)$ contains all possible graphs. Taking intersection with $\mathcal{C}$ results in the set of connected graphs $\mathcal{C}$.

B. Algebraic Model for Connectivity

One of the challenges of the setting introduced in Section II is to come up with an appropriate function representation for the connectivity property of graphs. The following two graph theoretic results will provide some insight into this direction.

Theorem 3.2 ([15]): Let A be the adjacency matrix of a graph $\mathcal{G}(A)$ with vertices $\{v_1, \ldots, v_n\}$. Then, the (i,j)-th entry of A^k is the number of paths of length k from v_i to v_j.

Theorem 3.3 (Connectivity): Let A be the adjacency matrix of a graph $\mathcal{G}(A)$ with vertices $\{v_1, \ldots, v_n\}$. Then, $\mathcal{G}(A)$ is connected if and only if there exists an integer k such that all the entries of the matrix $C_k(A) = I + A + A^2 + \cdots + A^k$ are non-zero.

We call the matrix $C_k(\mathbf{x}) = I + A(\mathbf{x}) + A^2(\mathbf{x}) + \cdots + A^k(\mathbf{x})$ the *k-connectivity matrix* of the graph $\mathcal{G}(\mathbf{x})$. By Theorem 3.3 it is clear that $C_k(\mathbf{x})$ captures the connectivity property of a graph. Let $\hat{u}(\cdot)$ be a continuous approximation to the step function defined as,

$$\hat{u}(y) = \lim_{\substack{w \to \infty \\ \varepsilon \to 0}} \sigma_w(y - \varepsilon) \to \begin{cases} 1 & \text{if } y > 0 \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

where $\sigma_w(y) = \frac{1}{1+e^{-wy}}$ is the sigmoid function, and define the matrix $H_k(\mathbf{x}) = \left(h_{ij}^{(k)}(\mathbf{x})\right)$ such that,

$$H_k(\mathbf{x}) = \hat{u}(C_k(\mathbf{x})) \tag{4}$$

where the step function $\hat{u}(\cdot)$ is applied to every entry of $C_k(\mathbf{x})$. Note that the (i,j)-th entry of the matrix $C_k(\mathbf{x})$ is just the number of paths of length at most k from node i to node j. Hence, the (i,j)-th entry of the matrix $H_k(\mathbf{x})$ simply denotes whether there *exists* a path of length at most k from node i to node j. $h_{ij}^{(k)}(\mathbf{x}) = 1$ implies that such a path exists, thought $h_{ij}^{(k)}(\mathbf{x}) = 0$ implies that such a path does not exist.

Let $\mathbf{x}_0 = [x_{1,0}^T \ldots x_{n,0}^T]^T$ denote a reference configuration of the nodes in the workspace as before, and define the set,

$$\mathcal{X}_k(\mathbf{x}_0) = \{\mathbf{x} \mid H_k(\mathbf{x}) = H_k(\mathbf{x}_0), \;\; H_{n-1}(\mathbf{x}_0) = 1_{n \times n}\}$$

The following proposition, which due to space limitations we state without proof, actually converts the graph theoretic problem of identifying the set of k-hop connected graphs $\mathcal{C}_k(\mathbf{x}_0)$ into an algebraic problem of specifying the set $\mathcal{X}_k(\mathbf{x}_0)$.

Proposition 3.4: $\mathcal{G}(\mathbf{x}) \in \mathcal{C}_k(\mathbf{x}_0)$ if and only if $\mathbf{x} \in \mathcal{X}_k(\mathbf{x}_0)$.

Hence, we conclude that equation (4) is indeed an appropriate representation for k-hop connectivity. The rest of this paper will be devoted in determining control constraints $U^*(\mathbf{x}(t))$ such that if $\mathbf{u}(t) \in U \cap U^*(\mathbf{x}(t))$, then $\mathbf{x}(t) \in \mathcal{X}_k(\mathbf{x}_0)$ for all $t \geq 0$.

IV. Maintaining Connectivity

A. Modeling and Dynamics of the Adjacency Matrix

Let, $d_{ij}(\mathbf{x}) = \|x_i - x_j\|_2$ denote the *Euclidean distance* between two nodes i and j. We say that nodes i and j are connected to each other by an edge in the graph $\mathcal{G}(\mathbf{x})$ if and only if $d_{ij}(\mathbf{x}) \leq \delta$, where δ is some specified threshold. Hence, we may define the (i,j)-th entry of the adjacency matrix $A(\mathbf{x})$ to be (Figure 1),

$$a_{ij}(\mathbf{x}) = \hat{u}(\delta - d_{ij}(\mathbf{x})) \tag{5}$$

where $\hat{u}(\cdot)$ is a continuous approximation to the step function given by equation (3). Obviously, $a_{ij}(\mathbf{x}) = 1$ if and only if $d_{ij}(\mathbf{x}) \leq \delta$, and equation (5) is consistent with Definition 2.1 of an adjacency matrix. Moreover, since $d_{ij}(\mathbf{x}) = d_{ji}(\mathbf{x})$ we also have that $a_{ij}(\mathbf{x}) = a_{ji}(\mathbf{x})$ and hence $A(\mathbf{x})$ is a symmetric matrix.

[3] We write $\mathcal{G}(\mathbf{x})$ instead of $\mathcal{G}(t)$ to emphasize the dependence of $\mathcal{G}(t)$ on the state $\mathbf{x}(t)$.

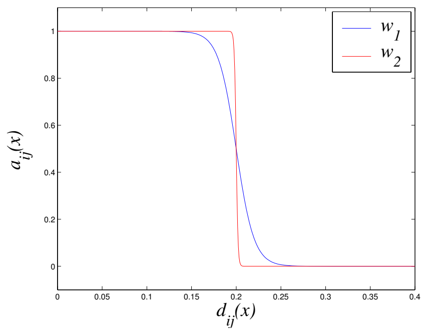

Fig. 1. Plot of the function $a_{ij}(\mathbf{x}) = \hat{u}(\delta - d_{ij}(\mathbf{x}))$ for parameter values $w_1 = 10^2$, $w_2 = 10^3$ and threshold $\delta = 0.2$.

Computing, the dynamics of the adjacency matrix is straightforward. Using the notation introduced in Section II, let $\nabla_{\mathbf{x}} a_{ij}(\mathbf{x})$ be the $mn \times 1$ column vector denoting the gradient of $a_{ij}(\mathbf{x})$ with respect to $\mathbf{x}$ and define the $n \times mn^2$ matrix $\nabla_{\mathbf{x}} A(\mathbf{x})$, with block structure,

$$\nabla_{\mathbf{x}} A(\mathbf{x}) = \big(\nabla_{\mathbf{x}} a_{ij}(\mathbf{x})^T\big) \tag{6}$$

We then have,

$$\dot{A}(\mathbf{x}) = \big(\dot{a}_{ij}(\mathbf{x})\big) = \big(\nabla_{\mathbf{x}} a_{ij}(\mathbf{x})^T \dot{\mathbf{x}}\big) = \nabla_{\mathbf{x}} A(\mathbf{x})\big(I_n \otimes \dot{\mathbf{x}}\big) \tag{7}$$

where I_n denotes the n-dimensional identity matrix and $\otimes$ denotes the *Kronecker* product[4].

B. Dynamics of the k-Connectivity Matrix

Using properties of *Kronecker* products[5], observe that,

$$\begin{aligned} \big(I_n \otimes \dot{\mathbf{x}}\big)A^k(\mathbf{x}) &= \big(I_n \otimes \dot{\mathbf{x}}\big)\big(A^k(\mathbf{x}) \otimes 1_{(1\times1)}\big) \\ &= \dots \\ &= \big(A^k(\mathbf{x}) \otimes I_{mn}\big)\big(I_n \otimes \dot{\mathbf{x}}\big) \end{aligned} \tag{8}$$

Differentiating $C_k(\mathbf{x})$ with respect to time and using equations (7) and (8) we get,

$$\begin{aligned} \dot{C}_k(\mathbf{x}) &= \dot{A}(\mathbf{x}) + \dot{A}(\mathbf{x})A(\mathbf{x}) + A(\mathbf{x})\dot{A}(\mathbf{x}) + \dots + \\ &\quad + A^{k-1}(\mathbf{x})\dot{A}(\mathbf{x}) \\ &= \dots \\ &= \left\{\sum_{i=0}^{k-1} C_{(k-1)-i}(\mathbf{x})\nabla_{\mathbf{x}} A(\mathbf{x})\big(A^i(\mathbf{x}) \otimes I_{mn}\big)\right\} \cdot \\ &\quad \cdot \big(I_n \otimes \dot{\mathbf{x}}\big) \end{aligned} \tag{9}$$

Equations (6) and (7) can also be applied to the connectivity matrix, yielding respectively $\nabla_{\mathbf{x}} C_k(\mathbf{x}) = \big(\nabla_{\mathbf{x}} c_{ij}^{(k)}(\mathbf{x})^T\big)$ and $\dot{C}_k(\mathbf{x}) = \nabla_{\mathbf{x}} C_k(\mathbf{x})\big(I_n \otimes \dot{\mathbf{x}}\big)$.

Hence, equation (9) can be rewritten as,

$$\begin{aligned} \nabla_{\mathbf{x}} C_k(\mathbf{x})\big(I_n \otimes \dot{\mathbf{x}}\big) &= \left\{\sum_{i=0}^{k-1} C_{(k-1)-i}(\mathbf{x})\nabla_{\mathbf{x}} A(\mathbf{x}) \cdot \right. \\ &\quad \left. \cdot \big(A^i(\mathbf{x}) \otimes I_{mn}\big)\right\}\big(I_n \otimes \dot{\mathbf{x}}\big) \end{aligned} \tag{10}$$

or equivalently,

$$\nabla_{\mathbf{x}} C_k(\mathbf{x}) = \sum_{i=0}^{k-1} C_{(k-1)-i}(\mathbf{x})\nabla_{\mathbf{x}} A(\mathbf{x})\big(A^i(\mathbf{x}) \otimes I_{mn}\big) \tag{11}$$

since equation (10) should hold for all $\dot{\mathbf{x}}$.

Equations (7) and (9) provide the dynamics of the adjacency matrix and the $C_k(\mathbf{x})$ matrix respectively, in terms of the dynamics $\dot{\mathbf{x}}$ of the nodes. Finally, equation (11) provides $\nabla_{\mathbf{x}} C_k(\mathbf{x})$ in terms of $\nabla_{\mathbf{x}} A(\mathbf{x})$. This relation is very useful when we need to compute the gradients $\nabla_{\mathbf{x}} c_{ij}^{(k)}(\mathbf{x})$.

C. Maintaining Connectivity

Let $\mathbf{x}_0 = [x_{1,0}^T \dots x_{n,0}^T]^T$ be the initial configuration of the nodes in the workspace. Obviously, $\mathbf{x}_0 \in \mathcal{X}_k(\mathbf{x}_0)$. Our goal is to derive constraints on the control variables $\mathbf{u}$ so that $\mathcal{X}_k(\mathbf{x}_0)$ is an invariant set. For all $t \geq 0$, define the system of differential inequalities,

$$\begin{cases} \dot{h}_{ij}^{(k)}(\mathbf{x}(t)) \geq 0 & \text{if } h_{ij}^{(k)}(\mathbf{x}_0) = 1 \\ \dot{h}_{ij}^{(k)}(\mathbf{x}(t)) \leq 0 & \text{if } h_{ij}^{(k)}(\mathbf{x}_0) = 0 \end{cases} \tag{12}$$

Then, obviously, for all $t \geq 0$, any configuration $\mathbf{x}(t) = [x_1^T(t) \dots x_n^T(t)]^T$ of the nodes satisfying the system (12) will belong in $\mathcal{X}_k(\mathbf{x}_0)$.

Since $C_k(\mathbf{x})$ is a symmetric matrix, so is $H_k(\mathbf{x})$. Hence, the matrix differential inequalities (12), actually reduce to a set of $\frac{n(n-1)}{2}$ differential inequalities corresponding to the upper triangular part of $H_k(\mathbf{x})$ (not including the diagonal entries). Thus, using also the fact that $\dot{h}_{ij}(\mathbf{x}) = \nabla_{\mathbf{x}} h_{ij}(\mathbf{x})^T \dot{\mathbf{x}} = \hat{u}'(c_{ij}(\mathbf{x}))\nabla_{\mathbf{x}} c_{ij}(\mathbf{x})^T \dot{\mathbf{x}}$, equation (12) can be rewritten as[6],

$$\begin{cases} \hat{u}'(c_{ij}(\mathbf{x}))\nabla_{\mathbf{x}} c_{ij}(\mathbf{x})^T \dot{\mathbf{x}} \geq 0 & \text{if } h_{ij}^0 = 1 \\ \hat{u}'(c_{ij}(\mathbf{x}))\nabla_{\mathbf{x}} c_{ij}(\mathbf{x})^T \dot{\mathbf{x}} \leq 0 & \text{if } h_{ij}^0 = 0 \end{cases} \quad \text{for every } i < j \tag{13}$$

where $\nabla_{\mathbf{x}} c_{ij}(\mathbf{x})^T$ is given by equation (11) and for notational simplicity we have dropped the index k. In matrix notation, (13) can be rewritten as,

$$G(\mathbf{x})\dot{\mathbf{x}} \geq 0 \tag{14}$$

where $G(\mathbf{x})$ is the $\frac{n(n-1)}{2} \times mn$ dimensional matrix given by,

$$G(\mathbf{x}) = \begin{pmatrix} (-1)^{1-h_{12}^0}\hat{u}'(c_{12}(\mathbf{x}))\nabla_{\mathbf{x}} c_{12}(\mathbf{x})^T \\ (-1)^{1-h_{13}^0}\hat{u}'(c_{13}(\mathbf{x}))\nabla_{\mathbf{x}} c_{13}(\mathbf{x})^T \\ \vdots \\ (-1)^{1-h_{(n-1)n}^0}\hat{u}'(c_{(n-1)n}(\mathbf{x})) \cdot \\ \cdot \nabla_{\mathbf{x}} c_{(n-1)n}(\mathbf{x})^T \end{pmatrix} \tag{15}$$

[4]Let $A = (a_{ij})$, $B = (b_{ij})$ be $n \times m$ and $p \times q$ matrices respectively. Their Kronecker product, denoted by $A \otimes B$, is the $np \times mq$ matrix with the block structure: $A \otimes B = (a_{ij}B)$. (see [16])

[5]Let A, B, C and D be matrices of appropriate dimensions. A property of the Kronecker product that will be of particular interest to us is: $(A \otimes B)(C \otimes D) = AC \otimes BD$. (see [16])

[6]for simplicity, we make use of the notation $\hat{u}'(y)$ to denote the derivative $\frac{d\hat{u}(y)}{dy}$

Suppose that the dynamics of the nodes in the graph are given by equation (1). Then, combining equations (1) and (14) the system dynamics become,

$$\begin{cases} G(\mathbf{x})F(\mathbf{x},\mathbf{u}) \geq 0 \\ \dot{\mathbf{x}} = F(\mathbf{x},\mathbf{u}) \end{cases} \tag{16}$$

For every configuration $\mathbf{x}$ of the nodes, the inequality $G(\mathbf{x})F(\mathbf{x},\mathbf{u}) \geq 0$ defines a set $U^*(\mathbf{x})$ of valid control inputs. Hence, in order to guarantee invariance of the set $\mathcal{X}_k(\mathbf{x}_0)$ we need to pick inputs $\mathbf{u}$ from the set $U \cap U^*(\mathbf{x})$. As long as this set is non-empty we can guarantee that by choosing $\mathbf{u} \in U \cap U^*(\mathbf{x})$, the graph $\mathcal{G}(\mathbf{x}(t))$ will always be k-hop connected.

Remark: Clearly, graph connectivity is a problem whose complexity grows exponentially with the number of nodes. The combinatorial nature of the problem is captured in the structure of the k-connectivity matrix. In our setting, k-hop connectivity serves as a tradeoff between the computationally expensive $(n-1)$-hop connectivity, where in order to guarantee connectivity, we have to account for all combinations of all possible path lengths between all pairs of nodes, and the computationally inexpensive 1-hop connectivity (keeping the same neighbors), where we only consider single edge paths between nodes.

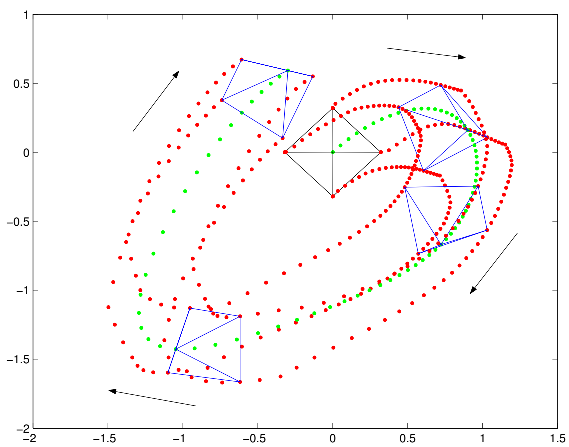

Fig. 2. One Leader, Four Followers / 1-hop connectivity (Keep the same neighbors).

V. Connectivity Tasks

The model we developed was based on the assumption that we have control of all nodes in the workspace. However, we will show that it also performs well in the *leaders-followers* case, and in particular when we have no control over the leaders. Assume that we have n nodes in the plane and that their dynamics are given by $\dot{\mathbf{x}} = \mathbf{u}$, where notation is according to the one introduced in Section II. The system of constraints (14) becomes $G(\mathbf{x})\mathbf{u} \geq 0$.

Let $L \subset \{1, 2, \dots, n\}$ denote the set of nodes corresponding to the leaders. We will assume that the dynamics of every leader $i \in L$ are of the form $u_i = f_i(x_i)$. Then the system of constraints becomes,

$$\begin{cases} G(\mathbf{x})\mathbf{u} \geq 0 \\ u_i = f_i(x_i) \quad \text{for every } i \in L \end{cases}$$

Hence, the problem becomes to find solutions that satisfy these constraints. Since solutions, if they exist, might not be unique, we may also choose to minimize a cost function. In particular, we will be interested in minimizing the *energy* given to the system. Thus, for every configuration $\mathbf{x}$, we will be solving the quadratic program,

$$\begin{array}{ll} \min\limits_{\mathbf{u}} & \|\mathbf{u}\|^2 \\ \text{s.t.} & G(\mathbf{x})\mathbf{u} \geq 0 \\ & u_i = f_i(x_i) \quad \forall i \in L \end{array}$$

The cost function is obviously not unique. Different cost functions will result in different solutions. The one we use in our setting gave some nice results which we now illustrate. In the following tasks, the initial graph configuration is denoted with black color, and the subsequent graphs with blue. The leaders are denoted with green and the followers with red[7].

A. One Leader, Four Followers

Let $L = \{1\} \subset \{1, \dots, 5\}$ correspond to the set of leaders, with dynamics given by,

$$u_1 = \begin{bmatrix} 1 + x_{1,2}\left(1 - \frac{3}{2}x_{1,2}\right) - x_{1,1}^3 \\ 1 - x_{1,1}\left(1 + x_{1,1}^2\right) - \frac{3}{2}x_{1,2}^2 \end{bmatrix}$$

and let the initial configuration of the nodes in the plane be: $\mathbf{x}_0 = \begin{bmatrix} 0 & 0 & 0.32 & 0 & 0 & 0.32 & -0.32 & 0 & 0 & -0.32 \end{bmatrix}^T$. We illustrate our results for connectivity threshold $\delta = 0.5$ and for k-hop connectivity values, $k = 1$ (Figures 2) and $k = 4$ (Figures 3). Comparing the respective figures, we may observe that in all cases, our model generates a control vector $\mathbf{u}$ such that the graph always satisfies the constraints we impose. Observe that for $k = 1$ (keep the same neighbors), edges are not allowed between nodes that were not initially connected by an edge. This is consistent with the definition of k-hop connectivity introduced in Section III.

B. Two Leaders, Three Followers (Cell Coverage Task)

Let $L = \{1, 2\} \subset \{1, \dots, 5\}$ correspond to the set of leaders, with dynamics given by the artificial potential functions [17],

$$u_i = -K\nabla_{x_i}\varphi_i(x_i) \quad \forall i \in L$$

where $\varphi_i(x_i) = \left(\frac{\gamma_{d,i}^r}{1+\gamma_{d,i}^r}\right)^{\frac{1}{r}}$ is the potential function, $\gamma_{d,i} = \|x_i - x_{d,i}\|^2$ is the distance of leader i to its destination $x_{d,i}$ and $K, r > 0$ are a gain and a parameter respectively.

Let the initial configuration of the nodes in the plane be: $\mathbf{x}_0 = \begin{bmatrix} 1 & 3 & 2 & 3 & 1.5 & 3.5 & 1.33 & 2.5 & 1.67 & 2.5 \end{bmatrix}^T$

[7]For animations of the connectivity tasks illustrated in this paper, we refer the reader to the web address: http://www.seas.upenn.edu/~zavlanos/

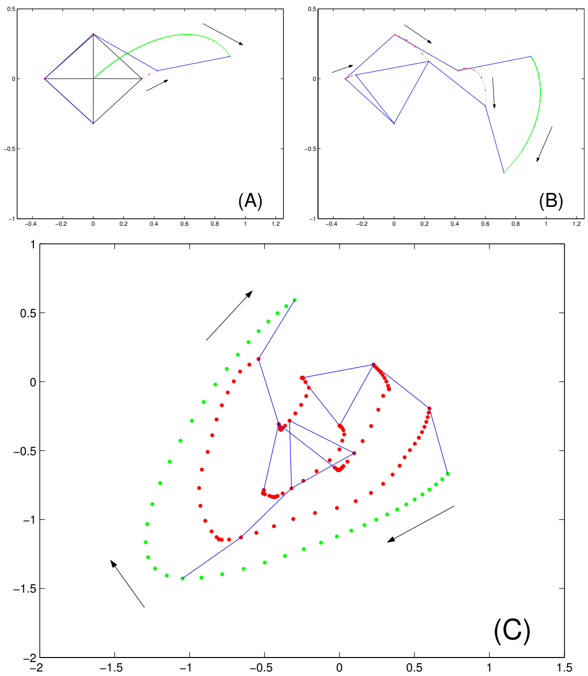

Fig. 3. One Leader, Four Followers / 4-hop connectivity (Stay Connected).

and the destinations of the leaders be: $x_{d,1} = \begin{bmatrix} 0 & 0 \end{bmatrix}^T$ and $x_{d,2} = \begin{bmatrix} 3 & 0 \end{bmatrix}^T$ respectively (Figure 4).

We require from the leaders to reach their destinations while the graph remains connected (i.e., $k = 4$). This is an example of a *cell coverage* task. We may observe that the task is accomplished in this case as well. (the connectivity threshold for this task is $\epsilon = 0.7571$)

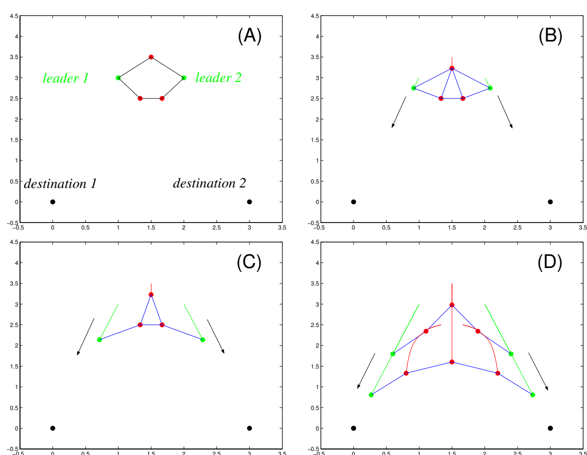

Fig. 4. Two Leaders, Three Followers / 4-hop connectivity (Cell coverage task).

VI. Conclusions

In this paper, we considered the problem of controlling the structure of dynamic graphs so that the resulting motion always preserves various connectivity properties. In particular, we introduced the notion of k-hop connectivity and developed a centralized control framework that guarantees maintenance of this property. The idea was to model connectivity as an invariance problem and transform it into a set of constraints on the control variables. Then, by minimizing an appropriate cost function, we were able to compute control laws for various connectivity tasks that illustrate the applicability of our method. We also showed that the notion of k-hop connectivity serves as a tradeoff between computational complexity and the size of the reachable set of graph configurations. We believe that this work points to a new direction in systems and control theory on the interface with algebraic and combinatorial graph theory.

VII. Acknowledgements

The authors gratefully acknowledge the contribution of Nicolò Giorgetti (University of Siena, Italy) in writing this paper.

References

[1] Demetri P. Spanos and Richard M. Murray. *Robust Connectivity of Networked Vehicles*, Proceedings of the IEEE Conference on Decision and Control, December 2004.

[2] Mehran Mesbahi. *On State-dependent Dynamic Graphs and their Controllability Properties*, IEEE Transactions on Automatic Control (50) 3: 387- 392, 2005.

[3] Yoonsoo Kim and Mehran Mesbahi. *On Maximizing the Second Smallest Eigenvalue of a State-dependent Graph Laplacian*, IEEE Transactions on Automatic Control (to appear).

[4] A. Jadbabaie, J. Lin and A. S. Morse. *Coordination of Groups of Mobile Autonomous Agents using Nearest Neighbor Rules*, IEEE Transactions on Automatic Control, (48) 6: 988-1001, 2003.

[5] R. Olfati-Saber and R. M. Murray. *Consensus Protocols for Networks of Dynamic Agents*, Proceedings of the American Control Conference, June 2003.

[6] R. Olfati-Saber and R. M. Murray. *Agreement Problems in Networks with Directed Graphs and Switching Topology*, Proceedings of the IEEE Conference on Decision and Control, December 2003.

[7] J. P. Desai, J. P. Ostrowski and V. Kumar. *Modeling and Control of Formations of Nonholonomic Mobile Robots*, IEEE Transactions on Robotics and Automation, vo. 17, no. 6, pp. 905-908, 2001

[8] H. Tanner, A Jadbabaie and G. Pappas. *Flocking in Fixed and Switching Networks*, IEEE Transactions on Automatic Control, April 2005. Submitted.

[9] P. Ögren, M. Egerstadt and X. Hu. *A Control Lyapunov Function Approach to Multi-agent Coordination*, IEEE Transactions on Robotics and Automation, vol. 18, no. 5, October 2002.

[10] J. S. Caughman, G. Lafferriere, J. J. P. Veerman, A. Williams. *Decentralized Control of Vehicle Formations*, Systems and Control Letters, 54, 899-910, 2005.

[11] G. Ayres de Castro and F. Paganini. *Convex Synthesis of Controllers for Consensus*, Proceedings of the American Control Conference, pages 4933-4938, June 2004.

[12] L. Moreau. *Leaderless Coordination via Bidirectional and Unidirectional Time-dependent Communication*. Proceedings of the IEEE Conference on Decision and Control, pages 3070-3075, December 2003.

[13] W. Ren and R. Beard. *Consensus of Information under Dynamically changing Interaction Topologies*, Proceedings of the American Control Conference, pages 4939-4944, June 2004.

[14] J. Cortes, S. Martinez and F. Bullo. *Coordinated Deploymeht of Mobile Sensing Networks with Limited-range Interactions*, Proceedings of the IEEE Conference on Decision and Control, pp. 1944-1949, December 2004.

[15] Norman Biggs, *Algebraic Graph Theory*, Cambridge University Press, 1993.

[16] Alexander Graham, *Kronecker Products and Matrix Calculus with Applications*, Halsted Press, John Wiley and Sons, NY, 1981.

[17] E. Rimon and D. E. Koditschek, *Exact Robot Navigation using Artificial Potential Functions*, IEEE Transactions on Robotics and Automation, vol. 8, no. 5, pp. 501-518, 1992.