2008 American Control Conference
Westin Seattle Hotel, Seattle, Washington, USA
June 11-13, 2008

Decentralized Formation Control via the Edge Laplacian

Daniel Zelazo, Amirreza Rahmani, Jasmine Sandhu, and Mehran Mesbahi†

Abstract— **Formation keeping strategies for groups of interconnected agents have recently been of great research interest in the systems community. In this work, we explore the utility of the edge variant of the graph Laplacian in the synthesis of formation keeping control laws. Along the way, we show a general duality relation between networked dynamic systems with measurement restrictions and those with control constraints. In both cases, it is shown that the closed loop error dynamics for the formation reduces to the edge agreement problem- which in turn- can be fully characterized via the spanning trees of the underlying interconnection topology.**

I. Introduction

Distributed dynamic systems are collections of dynamical units that interact over an information exchange network. Such systems are ubiquitous in diverse areas of science and engineering. Examples include physiological systems and gene networks [1], large scale energy systems, and multiple space, air, and land vehicles [2], [3], [4], [5]. The control and dynamical systems community is actively trying to formalize these systems and lay out a foundation for their analysis and synthesis [9], [6], [7], [8]. As a result, a distinct area of research that lies at the intersection of systems theory and graph theory has emerged. A basic yet fundamental class of problems that lies at this intersection is the Laplacian dynamics, also known as the agreement or consensus protocol [10], [11], [12], [13].

In our recent work [14], an edge variant of the agreement problem and its system-theoretic significance have been examined. The main results of [14] provided a better understanding of the role that certain subgraphs, e.g., cycles and spanning trees, play in the dynamics of the original agreement problem. We have also pointed out the ramification of the edge Laplacian in the parameterization of reduced order models via the spanning trees of the underlying network. The present paper demonstrates the utility of the edge Laplacian in the context of formation control. In this direction, we consider the formation keeping problem for a group of interconnected single integrator agents. Our main result indicates a duality between networked systems with measurement restrictions and those with control constraints. In both cases by synthesizing a controller with specific properties, we show that the closed loop error dynamics for the formation reduces to an edge agreement problem that in turn can be fully characterized via the spanning trees of the underlying interconnection topology.

†This work was supported by the National Science Foundation grants CMS-0301753 and ECS-0501606. D. Zelazo, A. Rahmani, and M. Mesbahi are with the Department of Aeronautics and Astronautics, University of Washington, Seattle, WA 98195-2400. Emails: $\{$*dlz* + *rahmani*$\}$@*u.washington.edu* and *mesbahi@aa.washington.edu*. J. Sandhu is with Lincoln Laboratory, MIT, Cambridge, MA; Email: *jasmine@ll.mit.edu*.

$$E = \begin{bmatrix} 1 & 1 & 1 \\ -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -1 \end{bmatrix}$$

(a)

$$E = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 \\ -1 & 0 & 0 & 1 & 0 & -1 \\ 0 & -1 & 0 & -1 & 1 & 0 \\ 0 & 0 & -1 & 0 & -1 & 1 \end{bmatrix}$$

(b)

Fig. 1. Incidence matrices for two simple graphs

Finally, we provide a "node" interpretation of the closed loop system and show that in special cases, the formation keeping problem reduces to the agreement problem.

II. Preliminaries and Notations

Since our approach to formation keeping employs ideas from graph theory, we will provide a few relevant notions that are used throughout the paper.

A. Graphs and their algebraic representation

An undirected (simple) graph $\mathcal{G}$ is specified by a vertex set $\mathcal{V}$ and an edge set $\mathcal{E}$ whose elements characterize the incidence relation between distinct pairs of $\mathcal{V}$. Two vertices i and j are called *adjacent* (or neighbors) when $\{i, j\} \in \mathcal{E}$; we denote this by writing $i \sim j$. The cardinalities of the vertex and edge sets of $\mathcal{G}$ will be denoted by $|\mathcal{G}|$ and $\|\mathcal{G}\|$, respectively. A *subgraph* of a graph $\mathcal{G}$ is a graph whose vertex and edge sets are subsets of those of $\mathcal{G}$. An *orientation* of an undirected graph $\mathcal{G}$ is the assignment of directions to its edges, i.e., an edge e_k is an ordered pair (i, j) such that i and j are, respectively, the initial and the terminal nodes of e_k.

Graphs admit a set of convenient matrix representations. For example, the $|\mathcal{G}| \times \|\mathcal{G}\|$ incidence matrix $E(\mathcal{G})$ for an oriented graph $\mathcal{G}$ is a $\{0, \pm 1\}$-matrix with rows and columns indexed by vertices and edges of $\mathcal{G}$, respectively, such that

$$[E(\mathcal{G})]_{ik} = \begin{cases} +1 & \text{if } i \text{ is the initial node of edge } e_k \\ -1 & \text{if } i \text{ is the terminal node of edge } e_k \\ \ \ 0 & \text{otherwise} \end{cases}$$

Figure 1 depicts an example of two oriented graphs and their respective incidence matrices. From the definition of the incidence matrix it follows that the null space of its transpose, $\mathcal{N}(E(\mathcal{G})^T)$, contains $\mathbf{span}\,\{\mathbf{1}\}$, where $\mathbf{1}$ is the vector with all entries equal to one. The rank of the incidence matrix depends only on $|\mathcal{G}|$ and the number of its connected components [16].

Theorem 2.1: Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ be a graph with c connected components. Then $\mathbf{rank}\, E(\mathcal{G}) = |\mathcal{G}| - c$.

978-1-4244-2079-7/08/$25.00 ©2008 AACC.

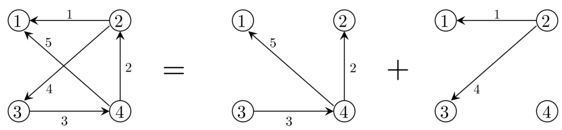


Fig. 2. A graph can be represented (not necessarily in a unique way) as a tree and edges that complete its cycles

The degree of a vertex is the cardinality of the set of vertices adjacent to it. A graph is complete if all possible pairs of vertices are adjacent, or equivalently, if the degree of all vertices is $|\mathcal{G}|-1$. A sequence of $r+1$ distinct and consecutively adjacent vertices, starting from vertex i and ending at vertex j, is called a path of length r (form i to j); when $i=j$, we call this path a *cycle*. We call a graph *connected* if there exists a path between any pair of vertices. A connected graph without cycles is referred to as a *tree*. Equivalently, a tree is a connected graph on $|\mathcal{G}|$ vertices with $|\mathcal{G}|-1$ edges. Figure 1(a) shows an example of a tree, while Figure 1(b) is a connected graph containing cycles.

The Laplacian of $\mathcal{G}$,

$$L_n(\mathcal{G}) := E(\mathcal{G})\, E(\mathcal{G})^T, \tag{1}$$

is a rank deficient positive semi-definite matrix [1].

A graph is connected if and only if the spectrum of its Laplacian has one zero element [16].

For notational convenience, in subsequent sections we will not include "$\mathcal{G}$" in the description of graphs and their associated matrices.

B. Role of cycles and trees in the edge Laplacian

The edge Laplacian of an arbitrary oriented graph $\mathcal{G}$ is defined as

$$L_e(\mathcal{G}) := E(\mathcal{G})^T\, E(\mathcal{G}). \tag{2}$$

The edge Laplacian is a real $\|\mathcal{G}\| \times \|\mathcal{G}\|$ symmetric matrix. It follows from (2) that the nonzero eigenvalues of the edge Laplacian are equal to the square of the nonzero singular values of $E(\mathcal{G})$. We note that adding an edge to the graph $\mathcal{G}$ increases the sum of the eigenvalues of $L_e(\mathcal{G})$ by two [14].

Theorem 2.2 ([14]): Given a connected directed graph $\mathcal{G}$ with incidence matrix $E(\mathcal{G})$, and edge Laplacian $L_e(\mathcal{G})$, the null space of $L_e(\mathcal{G})$ and $E(\mathcal{G})$ are equivalent. In particular, both null spaces characterize the cycles of the graph $\mathcal{G}$.

We will assume that all graphs under consideration are connected and hence contain a spanning tree. The edges that are not in the given spanning tree must complete the cycles in the graph. Using an appropriate permutation of the edge ordering, we can therefore express the incidence matrix as

$$E = \begin{bmatrix} E_\tau & E_c \end{bmatrix} \tag{3}$$

where E_τ represents a given spanning tree and E_c represents the remaining edges not in the tree (the cycle edges).

[1]Subscripts n and e are used to distinguish between operations on the nodes and edges of the graph respectively

The incidence relation, in turn, can be expressed as

$$E = [\, E_\tau \quad E_c \,] = E_\tau [\, I \quad T\,] = E_\tau\, R, \tag{4}$$

where

$$R = [\, I \quad T\,] \tag{5}$$

and

$$T = (E_\tau^T\, E_\tau)^{-1} E_\tau^T\, E_c \tag{6}$$

captures the relation between the cycle states and tree states [14], [18].

C. Edge agreement and its reduced order representation

The edge agreement problem is a variant of the well-studied agreement dynamics on the node states [14]. The agreement dynamics is defined as

$$\dot{x}_n(t) \;\; = \;\; -L_n(\mathcal{G}) x_n(t) \tag{7}$$

where $x_n(t) \in \mathbb{R}^{|\mathcal{G}|}$ denotes the state vector of the nodes in the network.

We define the edge states, $x_e(t) \in \mathbb{R}^{\|\mathcal{G}\|}$, as the difference between the states of two nodes incident to an edge,

$$x_e(t) = E^T x_n(t);$$

hence the edge agreement dynamics can be derived as

$$\dot{x}_e(t) = -L_e(\mathcal{G})\, x_e(t). \tag{8}$$

As indicated by (4), the incidence relation of the cycle edges are captured by the spanning tree of the graph. We can partition the edge state vector into the states corresponding to the tree edges, $x_\tau(t)$, and those of the cycle edges, $x_c(t)$, as

$$\begin{aligned} x_e(t) &= \begin{bmatrix} x_\tau^T(t) & x_c^T(t) \end{bmatrix}^T \\ &= R^T x_\tau(t). \end{aligned} \tag{9}$$

Using this observation and the definition of the edge Laplacian (2), we can parameterize reduced order representations of the edge agreement as follows.

Theorem 2.3 ([14]): The system described by (8) is equivalent to the reduced order system described by

$$\begin{aligned} \dot{x}_\tau(t) &= -E_\tau^T E_\tau R R^T x_\tau(t) \\ &= -L_e(\mathcal{G}_\tau) R R^T x_\tau(t), \end{aligned} \tag{10}$$

where $L_e(\mathcal{G}_\tau)$ is the edge Laplacian associated with a given spanning tree $\mathcal{G}_\tau$, $x_\tau(t)$ is the tree states, and the matrix R is defined as in (5).

III. Decentralized Laws for Formation Control

Formation keeping is one of the common control objectives in multi-agent systems. Representative scenarios include formation keeping problems for unmanned aerial vehicles, coordinated robots, and multi-satellite systems. A formation for a set agents can often be described by the relative states (e.g., positions) of the agents, in addition to the inertial position of the formation- as measured from an arbitrary point in the formation. For the remainder of our presentation, we will assume that a target formation is specified only by the relative positions. In such a setting, we proceed to consider two dynamic systems associated with a given formation, with dual interpretations of each other, and examine how the edge Laplacian can be used to synthesize decentralized stabilizing control laws for formation keeping.

A. Admissible formations

We first note that in general, the transformation from node to edge states expressed by

$$E^T : x_n(t) \rightarrow x_e(t) \tag{11}$$

is neither injective nor surjective. Fixing one of the nodes at the origin, however, eliminates the ambiguity in the node states knowing the edge states; in this case, relative states are sufficient to specify the state of the system and the transformation (11) becomes injective. In this direction, the transformation from the modified node states,

$$\bar{x}_n(t) = x_n(t) - x_n^{(i)}(t)\,\mathbf{1},$$

using a node anchored at the origin $x_n^{(i)}$, to the set of tree edge states $x_\tau(t)$ is bijective; hence the corresponding transformation (11) becomes invertible.

We also note that cycle states in the network should satisfy the geometric constraint imposed by the tree states. In other words, cycle edge states, $x_c(t)$, should satisfy the relation

$$x_c(t) = T^T x_\tau(t), \tag{12}$$

where the matrix T is defined as in (6).

Definition 3.1: A reference formation $z_r(t)$ is *admissible* if it satisfies the cycle constraint (12).

As an example, consider the graph in Figure 2. The relative formation

$$z_r = \begin{bmatrix} 1 & 2 & -1 & 1 & 3 \end{bmatrix}^T$$

is not admissible, since

$$T^T \begin{bmatrix} 1 \\ 2 \\ -1 \end{bmatrix} = \begin{bmatrix} -1 \\ 3 \end{bmatrix} \neq \begin{bmatrix} 1 \\ 3 \end{bmatrix}.$$

In the meantime, the transformation from node states with a node anchored at index i, i.e.,

$$\bar{x}_n = x_n - x_n(i)\,\mathbf{1},$$

to the set of tree edge states, $x_\tau(t)$, is bijective, and hence invertible.

B. Measurement-restricted systems

Consider a system of N agents modeled by single integrator dynamics

$$\begin{aligned} \dot{x}(t) &= u(t) && (13) \\ y(t) &= E^T x(t), && (14) \end{aligned}$$

where $x(t)$ is the collection of all N agent states, and $u(t)$ is the collection of the agents' controls. We assume that each agent has available to it a relative measurement for use in its control law. The specific measurement available to agent i is determined by the underlying interconnection topology of the system. We denote this topology by the incidence matrix E described in §II-A.

Given an admissible reference formation $z_r(t)$ (which for now is assumed to be constant), we can define the formation error signal as

$$e(t) = z_r(t) - E^T x(t). \tag{15}$$

Differentiating (15) leads to the formation error dynamics,

$$\begin{aligned} \dot{e}(t) &= -E^T u(t) && (16) \\ y(t) &= e(t). && (17) \end{aligned}$$

Proposition 3.2: The error dynamics (16) is controllable if and only if the incidence matrix E corresponds to a spanning tree. On the other hand, (16) is uncontrollable if there are cycles in the underlying network.

Proof: We note that the controllability matrix for (16) is $\mathcal{C} = -E^T$. From the discussion in §II-A, one has

$$\textbf{rank}\, E^T \leq N - 1.$$

If the incidence matrix E corresponds to a spanning tree, then $\mathcal{C}$ has full column rank and (16) is controllable. On the other hand, if E has p independent cycles, then (16) has p uncontrollable modes. ■

Let us first consider the case where the underlying topology corresponds to a spanning tree. We can define a feedback law of the form

$$u(t) = E_\tau e(t). \tag{18}$$

The closed-loop error dynamics then becomes

$$\begin{aligned} \dot{e}(t) &= -E_\tau^T E_\tau e(t) \\ &= -L_e(\mathcal{G}_\tau) e(t). && (19) \end{aligned}$$

From the properties of the edge Laplacian, it is evident that the formation error approaches the origin in the steady state from an arbitrary initial condition. Furthermore, the control law is decentralized, in the sense that the control law for each agent only relies on the formation error pertaining to its neighbors.

Next consider the scenario when the underlying topology contains cycles. Using the results from [14], we can thereby partition the error states into the error on the tree edges and the cycle edges as

$$e(t) = \begin{bmatrix} e_\tau(t)^T & e_c(t)^T \end{bmatrix}^T. \tag{20}$$

Proposition 3.3: The formation error dynamics described in (16) is equivalent to the following reduced order model,

$$\begin{aligned} \dot{e}_\tau(t) &= -E_\tau^T u(t) && (21) \\ e(t) &= R^T e_\tau(t). && (22) \end{aligned}$$

Proof: The complete error dynamics can be written as a linear combination of the error trajectory over a spanning tree. More specifically,

$$e(t) = R^T e_\tau(t) \tag{23}$$

where R is defined in (4). Differentiating (23) with respect to the dynamics in (16) and left-multiplying by the left-inverse of R^T gives (21). ∎

Proposition 3.3 shows that even when there are cycles in the network, one can always describe the formation error dynamics by an equivalent system on the underlying spanning tree. Using this result, we can then synthesize the feedback law

$$u(t) \quad = \quad E_\tau e_\tau(t), \tag{24}$$

for formation keeping. In this venue, the closed-loop formation error dynamics becomes

$$\begin{aligned} \dot{e}_\tau(t) &= -L_e(\mathcal{G}_\tau)e_\tau(t) && (25)\\ e(t) &= R^T e_\tau(t). && (26) \end{aligned}$$

Thus, the proposed control law is effectively ignoring the redundant information in the cycle structure of the network. The steady-state formation error still approaches the origin as $-L_e(\mathcal{G}_\tau)$ is Hurwitz.

C. Control-restricted systems

We now provide a dual interpretation of the system described in §III-B. Consider as a motivating example a network of nodes coupled physically, as in a piezoelectric network. In this setting, we envision that we have control over the edges on the network, resulting in the expansion or contraction of the physical lengths between the nodes. The system can thus be modeled as,

$$\begin{aligned} \dot{x}(t) &= E\,u(t), && (27)\\ y(t) &= x(t), && (28) \end{aligned}$$

where $x(t)$ represents the node states and the matrix E is the incidence matrix for the graph representing the interconnection topology. As in our discussion in §III-B, we can define the formation error signal to be (15). Differentiating (15) with respect to the dynamics in (27) leads to the error dynamics,

$$\begin{aligned} \dot{e}(t) &= -E^T E u(t) = -L_e(\mathcal{G})u(t) && (29)\\ y(t) &= e(t). && (30) \end{aligned}$$

Proposition 3.4: The error dynamics (29) is controllable if and only if the incidence matrix E corresponds to a spanning tree. On the other hand, (29) is uncontrollable if there are cycles in the underlying network.

Proof: The proof follows directly from the properties of the edge Laplacian. ∎

Let us now consider the case where $\mathcal{G}$ is a tree. Define a feedback law of the form,

$$u(t) \quad = \quad e(t). \tag{31}$$

Implementing this control law results in a closed-loop error dynamics identical to the earlier model (19). When there are cycles present in the network, we can again use the results from [14] to obtain a reduced order representation of the error dynamics.

Proposition 3.5: The formation error dynamics described in (29) is equivalent to the following reduced order model:

$$\begin{aligned} \dot{e}_\tau(t) &= -L_e(\mathcal{G}_\tau)Ru(t) && (32)\\ y(t) &= R^T e_\tau(t) && (33) \end{aligned}$$

where R is defined as in (5).

Proof: This follows from Theorem 2.3. ∎

Using the feedback law defined in (31) leads to the closed-loop system,

$$\begin{aligned} \dot{e}_\tau(t) &= -L_e(\mathcal{G}_\tau)RR^T e_\tau(t) && (34)\\ e(t) &= R^T e_\tau(t). && (35) \end{aligned}$$

To show that $e_\tau(t)$ converges to the origin for the system described in (34), we use the following result.

Theorem 3.6: The edge Laplacian for a graph with cycles, $L_e(\mathcal{G})$, is similar to the matrix

$$\begin{bmatrix} L_e(\mathcal{G}_\tau)RR^T & 0 \\ 0 & \mathbf{0} \end{bmatrix},$$

where $\mathcal{G}_\tau$ is a spanning tree subgraph of $\mathcal{G}$, the matrix R is defined in (5), and the block-matrix of zeros is square with dimension equal to the number of independent cycles in the graph.

Proof: We define a transformation matrix as

$$S \quad = \quad \begin{bmatrix} R^T & V \end{bmatrix}, \tag{36}$$

where the matrix V has columns corresponding to the orthonormal basis of the kernel of $L_e(\mathcal{G})$. In the meantime, as shown by Theorem 2.2, the columns of V span the cycle space of the graph as well. In fact, the matrix V is non-singular; its inverse is

$$S^{-1} \quad = \quad \begin{bmatrix} (RR^T)^{-1}R \\ V^T \end{bmatrix}. \tag{37}$$

Applying the transformation (36), we have

$$\begin{aligned} S^{-1}L_e(\mathcal{G})S &= S^{-1}R^T L_e(\mathcal{G}_\tau)RS \\ &= \begin{bmatrix} L_e(\mathcal{G}_\tau)RR^T & 0 \\ 0 & \mathbf{0} \end{bmatrix}. \end{aligned}$$

∎

Theorem 3.6 shows that the eigenvalues of $L_e(\mathcal{G}_\tau)RR^T$ corresponds to the non-zero eigenvalues of $L_e(\mathcal{G})$ where $\mathcal{G}$ is the complete graph. Since the non-zero eigenvalues are positive, it follows that $e_\tau(t)$ governed by (34) converges to the origin.

Recall that in §III-B, we showed that the control can effectively neglect the measurements associated with the cycle states. To continue with our duality interpretation, we observe that the formation error can be driven to the origin by utilizing the controls only on the tree edges, and neglecting the controls on the cycle edges.

We can partition the control into the tree and cycle components as,

$$u(t) \quad = \quad \begin{bmatrix} u_\tau(t)^T & u_c(t)^T \end{bmatrix}^T. \tag{38}$$

For the reduced order representation of (32), we now apply the control law,

$$u(t) = \begin{bmatrix} I \\ \mathbf{0} \end{bmatrix} e_\tau(t). \tag{39}$$

This control now results in the closed-loop system,

$$\dot{e}_\tau(t) = -L_e(\mathcal{G}_\tau)e_\tau(t) \tag{40}$$
$$e(t) = R^T e_\tau(t). \tag{41}$$

Clearly, the formation error converges to the origin, as the matrix $-L_e(\mathcal{G}_\tau)$ is Hurwitz.

D. Improved performance controllers

In the previous sections, we showed a duality between two different systems associated with a networked formation. In this direction, we examined how a simple decentralized control law results in identical closed-loop formation error dynamics. This dynamics corresponds to the edge Laplacian agreement protocol described in (8). At the same time, we observed that only the measurement and control over a spanning tree is necessary for the formation keeping. However, as shown in [14], cycles will improve the convergence rate of the edge agreement protocol. Therefore, it would be more desirable to have cycles in the topology, with a complete graph corresponding to the optimal interconnection.[2] Theorem 3.6 now suggests a way to synthesize a control law that imitates the performance of a complete graph while using the controls and measurements of a spanning tree.

Theorem 3.7: Consider the reduced order formation error dynamics described by (21). Define the feedback control law

$$u(t) = E_\tau \tilde{R}\tilde{R}^T e_\tau(t) \tag{42}$$

with $\tilde{R}$ being the usual matrix R (5), associated with the incidence matrix E_c of the complement graph of $\mathcal{G}_\tau$. Then the closed-loop error dynamics corresponds to an edge agreement protocol for a complete graph.

Proof: The closed-loop error dynamics using the control law (42) is

$$\dot{e}_\tau(t) = -E_\tau^T E_\tau \tilde{R}\tilde{R}^T e_\tau(t).$$

Using Theorem 3.6, we can see that the eigenvalues of the matrix $E_t^T E_t \tilde{R}\tilde{R}^T$ are the same as the non-zero eigenvalues of the edge Laplacian for the complete graph. ■

E. Node representation of formation problem

Consider the system of (13), given the interconnection topology, apply the feedback controller of the form

$$u(t) = Ee(t).$$

Then the closed loop system in terms of the node states assumes the form

$$\dot{x}(t) = -L(\mathcal{G})x(t) + Ez_r(t). \tag{43}$$

When $z_r = 0$, the dynamics (43) reduces to the well-known consensus problem over the node states.

[2]This restricted notion of optimality does not take into account the cost of extra edges in the network.

On the other hand, when the improved performance controller is used, the closed loop system in terms of the node states becomes

$$\begin{aligned} \dot{x}(t) &= -E_t\tilde{R}\tilde{R}^T E_t x(t) + E_\tau \tilde{R}\tilde{R}^T z_\tau(t) \\ &= -L(\mathcal{G}_c)x(t) + E_c\tilde{R}^T z_\tau(t) \end{aligned} \tag{44}$$

where the subscript c denotes the complete graph, and z_τ corresponds to the desired formation defined on the tree edges. The formation keeping problem can therefore be seen as a controlled agreement problem [15]. We note that the condition on admissible formations corresponds to the controllability of this system.

F. Double integrator model

We conclude the main technical part of the paper by pointing out how our analysis for formation consisting of agents with first order dynamics can be extended to those with double integrator dynamics. In this direction, consider the relative state dynamics of a network of double integrators associated with a spanning tree $\mathcal{G}_\tau$. Next, construct the error dynamics that corresponds to a given edge reference signal, $\zeta_r(t) := [\, z_r(t)^T \ \dot{z}_r(t)^T \,]^T$; hence

$$\ddot{e}(t) = -E_\tau^T \ddot{x}(t) = -E_\tau u(t).$$

Define now the state feedback controller

$$u(t) = K\,[\, e(t)^T \ \dot{e}(t)^T \,]^T$$

and set $K = [\, E_\tau \ E_\tau \,]$. The resulting closed loop system is then

$$\begin{bmatrix} \dot{e}(t) \\ \ddot{e}(t) \end{bmatrix} = \begin{bmatrix} 0 & I \\ -L_e(\mathcal{G}_\tau) & -L_e(\mathcal{G}_\tau) \end{bmatrix} \begin{bmatrix} e(t) \\ \dot{e}(t) \end{bmatrix}.$$

We note that for the above state matrix A_{cl}, the characteristic equation is $\det A_{cl} = \det(\lambda^2 I + (\lambda+1)L_e(\mathcal{G}_\tau)) = 0$. Since $\lambda = -1$ does not satisfy this equation it is not an eigenvalue of A_{cl}. The eigenvalues of A_{cl} thus satisfy

$$\det(\lambda^2/(\lambda+1)\, I + L_e(\mathcal{G}_\tau)) = 0.$$

Denoting the eigenvalues of $-L_e(\mathcal{G}_\tau)$ by μ, one has that for each i,

$$\mu_i = \lambda_i^2/(\lambda_i + 1),$$

and hence

$$\lambda_i = \left(\mu_i \pm \sqrt{\mu_i^2 + 4\mu_i}\right)/2.$$

Since $-L_e(\mathcal{G}_\tau)$ is negative definite, $\mu_i < 0$ for all i; thus A_{cl} is Hurwitz guaranteeing that $\{e(t), \dot{e}(t)\} \to 0$ as $t \to \infty$. The closed loop system that corresponds to the relative state dynamics with $\zeta(t) := [\, z(t)^T \ \dot{z}(t)^T \,]^T$ is now

$$\dot{\zeta}(t) = \mathcal{L}_e(\mathcal{G}_\tau)\,\zeta(t) + \mathcal{F}_e(\mathcal{G}_\tau)\,\zeta_r(t) \tag{45}$$

where

$$\begin{aligned} \mathcal{L}_e(\mathcal{G}_\tau) &:= \begin{bmatrix} 0 & I \\ -L_e(\mathcal{G}_\tau) & -L_e(\mathcal{G}_\tau) \end{bmatrix}, \\ \mathcal{F}_e(\mathcal{G}_\tau) &:= \begin{bmatrix} 0 & 0 \\ L_e(\mathcal{G}_\tau) & L_e(\mathcal{G}_\tau) \end{bmatrix}. \end{aligned}$$

If we convert the edge dynamics back to the node dynamics, the closed loop system assumes the form

$$\dot{\eta}(t) = \mathcal{L}(\mathcal{G}_\tau)\, \eta(t) + \mathcal{E}(\mathcal{G}_\tau)\, \zeta_r(t), \tag{46}$$

where $\eta(t) := [\, x(t)^T \ \dot{x}(t)^T \,]^T$,

$$\mathcal{L}(\mathcal{G}_\tau) = \begin{bmatrix} 0 & I \\ -L(\mathcal{G}_\tau) & -L(\mathcal{G}_\tau) \end{bmatrix},\ \mathcal{E} = \begin{bmatrix} 0 & 0 \\ E_\tau & E_\tau \end{bmatrix},$$

and $L(\mathcal{G}_\tau)$ is the graph Laplacian of $\mathcal{G}_\tau$. Once again, we note a duality correspondence between the node and edge dynamics of (45) and (46).

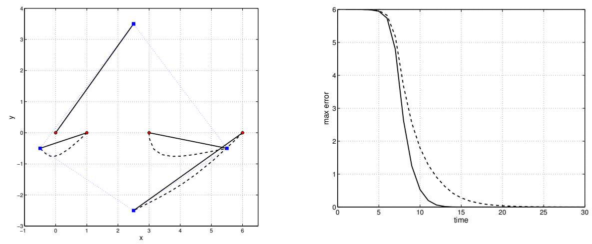
IV. Simulations

Here we use the results described in §III-D to drive 4 nodes with a star interconnection topology to a diamond-shaped final relative formation. We assume the agents have first order dynamics of the form (13). We developed two controllers, one that uses the relative feedback over the described star topology as in (24), while the other one is based on the improved performance controller of (42).

Figure 3(a) depicts the trajectories of the nodes for each controller. The agents all start aligned on the x-axis and move to their final relative position of a diamond (cubes). The trajectories of the agents using the improved controller are more direct. As we expected this will improve the performance by driving the RMS error to zero faster than the simple topology feedback controller. Figure 3(b) shows the maximum formation error for both controllers and how the improved control feedback law has a better performance. We should emphasize that the improved controller is not optimal in terms of the global performance and is only the optimal provided the limited knowledge (relative positions) and the required local interaction topology.

V. Conclusions

In this paper we applied results from [14] to the problem of decentralized formation control for a group of agents. Our results highlighted a duality between networked systems that are measurement-restricted and those that are control-restricted. In both cases, the form of the proposed control laws are such that the closed-loop error dynamics reduces to an edge agreement problem. Using reduced order representations of the edge Laplacian then led to decentralized controllers that require only measurements and control over the tree states. Furthermore, we show the performance achievable on a complete network, in terms of the convergence rate of the formation error, can be emulated by using only a spanning tree measurement topology.



(a) node trajectories (b) maximum relative error

Fig. 3. Controller performance for diamond formation of a star topology controller (dashed) and improved controller (solid)

References

[1] C. Fall, E. Marland, J. Wagner, and J. Tyson (Editors). *Computational Cell Biology*, Springer, 2005.

[2] T. Balch and R. C. Arkin. Behavior-based formation control for multi-robot teams, *IEEE Transactions on Robotics and Automation*, 14: 926-939, 1998.

[3] M. Mesbahi and F. Y. Hadaegh. Formation flying control of multiple spacecraft via graphs, matrix inequalities, and switching, *AIAA Journal of Guidance, Control, and Dynamics*, (24) 2: 369-377, 2001.

[4] G. Walsh, H. Ye, and L. Bushnell. Stability analysis of networked control systems, *American Control Conference*, June 1999.

[5] P. K. C. Wang and F. Y. Hadaegh. Coordination and control of multiple microspacecraft moving in formation, *Journal of the Astronautical Sciences*, 44: 315-355, 1996.

[6] B-D. Chen and S. Lall. Dissipation inequalities for distributed systems on graphs, *IEEE Conference on Decision and Control*, Dec. 2003.

[7] J. Cortes and F. Bullo. Coordination and geometric optimization via distributed dynamical systems, *SIAM Journal of Control and Optimization*, (45) 5: 1543-1574, 2005.

[8] R. D'Andrea and G. E. Dullerud. Distributed control design for spatially interconnected systems. *IEEE Transactions on Automatic Control*, (48) 9: 1478-1495, 2003.

[9] D. P. Bertsekas and J. N. Tsitsiklis. *Parallel and Distributed Computation*, Prentice Hall, 1989.

[10] Y. Hatano and M. Mesbahi. Agreement over random networks. *IEEE Transactions on Automatic Control*, (50) 11: 1867-1872, 2005.

[11] L. Moreau. Stability of multiagent systems with time-dependent communication links, *IEEE Transactions on Automatic Control*, (50) 2: 169 - 182, 2005.

[12] R. Olfati-Saber and R. M. Murray. Consensus problems in networks of agents with switching topology and time-delays, *IEEE Transactions on Automatic Control*, (49) 9: 1520-1533, 2004.

[13] L. Xiao and S. Boyd. Fast linear iterations for distributed averaging. *Systems and Control Letters*, (53): 65-78, 2004.

[14] D. Zelazo, A. Rahmani, and M. Mesbahi. Agreement via the Edge Laplacian. *IEEE Conference on Decision and Control*, Dec. 2007.

[15] A. Rahmani and M. Mesbahi. On the controlled Agreement Problem. *American Control Conference,*, June 2006. 2007.

[16] C. Godsil and G. Royle. *Algebraic Graph Theory*, Springer, 2001.

[17] R. A. Horn and C. Johnson. *Matrix Analysis*. Cambridge University Press, 1985.

[18] J. Sandhu, M. Mesbahi and T. Tsukamaki. Relative Sensing Networks: Observability, Estimation, and the Control Structure, *IEEE Conference on Decision and Control*, 2005.

[19] J. Sandhu, M. Mesbahi, and T. Tsukamaki. Cuts and flows in relative sensing and control of spatially distributed systems, *American Control Conference*, 2005.

[20] M. Fiedler. Algebraic connectivity of graphs, *Czechoslovak Mathematical Journal*, 23 (98): 298-305, 1973.

[21] J. A. Fax and R. M. Murray. Information flow and cooperative control of vehicle formations, *IEEE Transactions on Automatic Control*, (49) 9: 1465- 1476, 2004.

[22] 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.