Proceedings of the
47th IEEE Conference on Decision and Control
Cancun, Mexico, Dec. 9-11, 2008



# Persistent Formation Control of Multi-Robot Networks

Brian S. Smith, Jiuguang Wang, and Magnus B. Egerstedt

*Abstract*— **This paper presents a method for controlling formations of mobile robots. In particular, the problem of maintaining so-called "persistent formations" while moving the formation from one location to another is defined and investigated. A method for accomplishing such persistent formation motions is presented, and the method is demonstrated in simulation and with a prototype network of robots.**

## I. Introduction

This work is conducted in the contract of a National Aeronautics and Space Administration (NASA) project to implement a multi-robot system for research in Antarctica. In this project, a team of geologists at NASA should be able to use a mobile sensor network composed of mobile robots to take sensor readings across ice shelves to better understand the impacts of global climate change [1].

According to specifications, the network should be able to automatically deploy and distribute itself across an area of interest with a user-defined resolution, and to achieve specific, user-defined geometric relationships among the members of the network. This is generalized as the ability of the network to assemble and deploy *formations* (see [2], [3], [4], [5] for a representative sample). However, the environment in which the network is deployed is dynamic. The Antarctic ice sheets are being studied to better understand, in part, why they "break up" [6]. As such, the desired position of the formation may change as the system collects data. This change may occur due to the environment, such as a dangerous event on the ice shelf. This may also occur when sensor data from a different area, or repositioning of the sensors relative to a moving target, is requested.

Our previous work on this topic with a prototype network of mobile robots presents methods for the network assembly in a decentralized fashion [7], [8]. This work was based on models of formation specifications as weighted graphs in the spirit of [9], [10], [11], [12]. In these graphs, the vertices correspond to agents, and the edges correspond to *constraints*, i.e. geometric relationships between robots that are established and maintained. In particular, the method in [8] uses graph-based rules for assembling the formation as a sequence of graph operations. The result of these graph operations is a final network graph that is isomorphic to the formation specification graph, and the geometry of the mobile sensor network matches the geometry of the desired formation. In [7], [8], desired formations for the network are modeled as *minimally persistent graphs* [14]. Persistent graphs are directed graphs, and the direction of the edge corresponds to which agent is responsible for maintaining the constraint. *Minimally* persistent, in turn, ensures that as few such edges as possible are used.

In this paper, we address the issue of *formation motion*, i.e. moving the agents from one location to another while maintaining the formation. Specifically, we present control laws for moving minimally persistent formations. Previous work has presented control laws for a similar version of this problem, using control strategies with both holonomic and nonholonomic agents [13]. Due to the complexity of the control laws in [13], explicit proofs of stability were left to future work, even though they performed well in simulation. In this paper, by addressing only holonomic control strategies, we can make explicit claims as of the stability of the formation error.

In Section II the problem of moving minimally persistent formations is defined. Section III presents control strategies for moving persistent formations. Section IV describes how the robots determine their desired positions as dictated by their constraints, as well as the permissible error of the formation. Section V discusses simulation results and implementation results with actual robots. Finally, Section VI concludes the paper.

## II. Problem Definition

This section discusses the problem of maintaining a formation during motion. We introduce our model of the system, preliminary information about the initial state of the network, and the desired trajectory of the network. We also define a network graph that models the network's control topology.

### A. Network and Trajectory Modeling

We define $n \geq 2$ as the number of agents in the network, and $N = \{1, \ldots, n\}$ as a set of indices such that $i \in N$ is the index of agent $i$. We represent each agent over an interval of time $T = [0, \infty)$. Each agent is represented as a state $x_i : T \mapsto \mathbb{R}^2$ such that, $\forall i \in N$, $x_i(t)$ is the planar position of agent $i$ at time $t \in T$. We assume that the dynamics of each agent is given by a single integrator, i.e. for each $i \in N$, $u_i : T \mapsto \mathbb{R}^2$ is the control for agent $i$ such that $\dot{x}_i = u_i$. The state of the entire network is represented by a *network trajectory* $X : T \mapsto \mathbb{R}^{2n}$ such that $X(t) = \left[x_1(t)^T, \ldots, x_n(t)^T\right]^T$.

We define a *desired state* of each agent $i$ as $x_i^* : T \mapsto \mathbb{R}^2$ $\forall i \in N$ such that $x_i^*(t)$ is the desired state of agent $i$ at time $t$. Moreover, we assume that $x_i^*$ is continuously differentiable $\forall i \in N$. The desired state of the entire network is represented by a *desired network trajectory* $X^*(t) = \left[x_1^*(t)^T, \ldots, x_n^*(t)^T\right]^T$.

Brian S. Smith, Jiuguang Wang, and Magnus B. Egerstedt are with the School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA 30332-0250, USA {brian, magnus}@ece.gatech.edu, j.w@gatech.edu

We need $X^*$ to describe a trajectory that maintains a desired formation. Therefore, for each $i \in N$, we define a nominal *formation position* $p_i \in \mathbb{R}^2$. These positions define the "shape" of the formation in that they define the desired distances between all pairs of agents. For this problem, we assume that $p_i \neq p_j \ \forall (i,j) \in N \times N$. Each pair of positions $(p_i, p_j)$ defines $d_{ij} = \|p_i - p_j\|$, $\forall (i,j) \in N \times N$. To satisfy this formation, we require that $\|x_i^*(t) - x_j^*(t)\| = d_{ij}$ $\forall (i,j,t) \in N \times N \times T$. This prerequisite for $X^*$ based on the positions and the distances they define is sufficient to guarantee that $X^*(t)$ represents the same formation $\forall t \in T$ [11].

*The goal for the network is to (asymptotically) maintain formation while following the desired trajectory. In other words, we desire that* $\lim_{t\to\infty} X(t) - X^*(t) = 0$. To this end, we define the *network formation error* $\tilde{X} = X - X^*$, and we want $\lim_{t\to\infty} \tilde{X}(t) = 0$. We represent the *error* of each agent $i \in N$ by $\tilde{x}_i : T \mapsto \mathbb{R}^2$ such that $\tilde{x}_i = x_i - x_i^*$.

Initially, we assume that the network is close to defining the perfect, error-free formation, i.e. for some small $\epsilon > 0$, we assume that $\|\tilde{X}(0)\| \leq \epsilon$. Previous work shows that we can assemble such formations with an arbitrarily bounded error [8]. Thus, $\epsilon$ can be arbitrarily small.

### B. Formations Represented with Persistent Graphs

We represent the control topology as a graph $G = (V, E)$, such that $V = \{v_1, \ldots, v_n\}$ is the vertex set, and $E \subset V \times V$ is the edge set. Here, each vertex in $V$ is associated with its corresponding agent, i.e. $v_i$ is the index corresponding to agent $i$. We also assume that each edge is directed, in that the ordered pair $(v_i, v_j) \in E$ indicates that the control law of agent $i$ depends on the position of agent $j$. This control law should be designed to maintain a specified relationship between agents $i$ and $j$. Thus, we say that each edge $(v_i, v_j) \in E$ represents a *network constraint* of agent $i$ with agent $j$. The number of constraints of agent $i$ corresponds to the *out-degree* of vertex $v_i$, defined as $|\{(v_a, v_b) \in E : v_a = v_i\}|$.

For the purposes of this paper, we assume that $G$ is a *stably rigid, minimally persistent graph* [14], [15] as shown in Fig. 1(a). This implies several properties of $G$. First, $G$ is a connected, directed acyclic graph (DAG). Also, $G$ has a *leader* vertex $v_l \in V$ with an out-degree of zero and a *first-follower* vertex $v_f \in V$ with an out-degree of one, such that $(v_f, v_l) \in E$. Thus, we say that our network has a leader agent $l$ and a first-follower agent $f$. Each remaining vertex is a *follower* vertex, i.e. the follower vertices are all vertices in $V \setminus \{v_l, v_f\}$. Each follower vertex has an out-degree of two. Furthermore, each edge is in a directed path to the leader vertex. Finally, our assumptions imply that $G$ is minimally rigid [16] and *constraint consistent* [14]. We also assume that $\forall k \in N$ such that $\{(v_k, v_i), (v_k, v_j)\} \subset E$, $x_i^*(t)$, $x_j^*(t)$, and $x_k^*(t)$ are *not* collinear $\forall t \in T$. For developments in this paper, we define the indices of the agents such that 1 is the index of the leader, 2 is the index of the first-follower, and all other agent indices correspond to a reverse topological order such that $\forall (v_j, v_i) \in E$, $j > i$. This topological ordering is possible because, by our assumptions, $G$ is a DAG. Since all followers have an out-degree of two, this implies that, $\forall \{k \in N : k \geq 3\}$, agent $k$ is a follower, and $\exists (i,j)$ such that $\{(v_k, v_i), (v_k, v_j)\} \subset E$.

Due to the rigidity of the network, all inter-agent distances are preserved if the constraints of each agent are maintained, i.e. $\|x_i(t) - x_j(t)\| = d_{ij}$ $\forall (v_i, v_j) \in E$ $\forall t \in T \Leftrightarrow \|x_i(t) - x_j(t)\| = d_{ij}$ $\forall (i,j,t) \in N \times N \times T$. Therefore, the formation is maintained by only maintaining a subset of inter-agent distances (in this case, with $2n-3$ constraints). Qualitatively, the constraint consistency ensures that no agent can satisfy its constraints in a manner which forces another agent to violate a constraint (for a more rigorous treatment of persistence and constraint consistence, see [14]). Since the graph is acyclic, the edges of the graph can be assembled from an empty set by first adding the edge $(v_f, v_l)$ between the leader and first-follower, and then performing a sequence of *vertex addition operations*, in which a pair of edges $(v_k, v_i)$ and $(v_k, v_j)$ are added for each vertex in $V \setminus \{v_l, v_f\}$, as shown in [7], [8].

Since the network has a leader whose vertex has an out-degree of zero, this implies that the leader has no constraints; it is free to move regardless of its relationship to other agents. Similarly, the first-follower only has one constraint with the leader. Therefore, the first-follower can rotate around the leader. Rotating the first-follower around the leader while the other followers maintain their constraints causes the formation to rotate. This suggests a useful strategy for positioning and rotating the formation by coordinating the position and rotation of the leader and first-follower agents.

## III. Persistent Control Strategies

In this section, we define control strategies for maintaining persistent formations. We discuss two different control laws, each based on assumptions about the information available to each agent in terms of $X^*$ and its dynamics.

### A. Relative Formation Geometry

The desired trajectory $X^*$ defines the desired translation and rotation of the formation. Specifically, $x_1^*$ defines the location of the leader as a function of time, and $x_2^*$ defines the rotation of the first-follower as a function of time. Similarly, the geometry defined by $X^*$ defines the *relative position* of all agents. For each follower agent $k$, there exists $(i,j)$ such that $\{(v_k, v_i), (v_k, v_j)\} \subset E$ and a function $f_k$ such that $x_k^* = f_k(x_i^*, x_j^*)$. Section IV discusses the equations for defining $f_k$ for all followers.

We always assume that the leader and first-follower have access to their desired states. We define persistent control strategies for two cases. In the first case, we assume that $\forall k \in N$ such that $\{(v_k, v_i), (v_k, v_j)\} \subset E$, follower agent $k$ has access to $x_i^*$ and $x_j^*$, and can thus derive $x_k^*$. Here, the agents are controlled with knowledge of their desired states and their dynamics. In the second case, we assume that each follower agent $k$ can only estimate $x_k^*$ by the actual positions of the other agents as $f_k(x_i, x_j)$. Here, the agents are controlled by estimating their desired states and their dynamics.

### B. Control With Knowledge of the Desired States and Their Dynamics

We assume that the leader agent 1 and the first-follower agent 2 have access to $x_1^*$ and $x_2^*$. For our first scenario, we assume that, if $n \geq 3$, agents 1 and 2 communicate these to agent 3. This implies that agent 3 can calculate both $x_3^*$ and $\dot{x}_3^*$. In general, we assume that, for each agent $k \in N$ such that $\{(v_k, v_i), (v_k, v_j)\} \subset E$, agents $i$ and $j$ share $(x_i^*,\ \dot{x}_i^*,\ x_j^*,\ \dot{x}_j^*)$ with agent $k$. This could be accomplished by agents sharing these as explicit functions of time, or communicating these values as the formation is executed. Then, agent $k$ can derive $x_k^* = f_k\left(x_i^*, x_j^*\right)$ and $\dot{x}_k^* = \frac{\partial f_k}{\partial x_i^*}\left(x_i^*, x_j^*\right)\dot{x}_i^* + \frac{\partial f_k}{\partial x_i^*}\left(x_i^*, x_j^*\right)\dot{x}_j^*$. We can define each agent's control based on its desired state and the desired state's dynamics.

We define agent $i$'s control by $u_i = \dot{x}_i^* - K_i\tilde{x}_i$, where $-K_i \in \mathbb{R}^{2\times 2}$ is a Hurwitz matrix. This implies that the error dynamics are

$$\dot{\tilde{x}}_i = \dot{x}_i - \dot{x}_i^* = u_i - \dot{x}_i^* = \dot{x}_i^* - K_i\tilde{x}_i - \dot{x}_i^* = -K_i\tilde{x}_i.$$

The system $\dot{\tilde{x}}_i = -K_i\tilde{x}_i$ has a globally exponentially stable origin. Further, the network error is described by the system $\dot{\tilde{X}} = -K\tilde{X}$, where $-K \in \mathbb{R}^{2n\times 2n}$ is a Hurwitz matrix with $-K_1, \ldots, K_n$ on its diagonal. Therefore, the network error has a globally exponentially stable origin. Hence, $\lim_{t\to\infty}\tilde{X}(t) = 0$.

### C. Control Without Knowledge of the Desired States and Their Dynamics

Here, we consider the case of limited knowledge of $X^*$. We assume that the leader agent 1 has access to $x_1^*$, and the first-follower agent 2 has access to $x_2^*$. However, agents $k \in N$ such that $\{(v_k, v_i), (v_k, v_j)\} \subset E$ do not have access to either $x_i^*$, $x_j^*$, or $x_k^*$. However, we assume that agent $k$ has access to both $x_i$ and $x_j$, as in the case where agent $k$ can estimate the relative position of agents $i$ and $j$ through sensor information. For each agent $i$, we define an *estimation of its desired state* as $\hat{x}_i^*$. The follower agents estimate the desired state by $\hat{x}_k^* = f_k\left(x_i, x_j\right)$ and $\dot{\hat{x}}_k^* = \frac{\partial f_k}{\partial x_i}\left(x_i, x_j\right)\dot{x}_i + \frac{\partial f_k}{\partial x_j}\left(x_i, x_j\right)\dot{x}_j$. Since the leader and first-follower agents still have access to $x_1^*$ and $x_2^*$, we define $\hat{x}_1^* = x_1^*$ and $\hat{x}_2^* = x_2^*$. We also define an estimation of the desired state of the network as $\hat{X}^* = [\hat{x}_1^{*T}, \ldots, \hat{x}_n^{*T}]^T$. Similarly, we define an *estimation of the error* of agent $i$ as $\hat{x}_i = x_i - \hat{x}_i^*$, and the *network error estimation* as $\hat{X} = X - \hat{X}^*$.

We define agent $i$'s control law by $u_i = \dot{\hat{x}}_i^* - K_i\hat{x}_i$, where $-K_i \in \mathbb{R}^{2\times 2}$ is a Hurwitz matrix. This implies that $\dot{\hat{x}}_i = -K_i\dot{\hat{x}}_i\ \forall i \in N$, which has a globally stable origin. This implies that $\hat{X}$ has a globally exponentially stable origin and $\lim_{t\to\infty}\hat{X}(t) = 0$.

If, for each $k \in N$ such that $\{(v_k, v_i),\ (v_k, v_j)\} \subset E$, we assume that $\lim_{t\to\infty}\tilde{x}_i(t) = \lim_{t\to\infty}\tilde{x}_j(t) = 0$, then this implies that

$$\begin{aligned}
&\lim_{t\to\infty}\left(\hat{x}_k^*(t) - x_k^*(t)\right)\\
&= \lim_{t\to\infty}\left(f_k\left(x_i(t), x_j(t)\right) - f_k\left(x_i^*(t), x_j^*(t)\right)\right)\\
&= \lim_{t\to\infty}\left(f_k\left(x_i^*(t), x_j^*(t)\right) - f_k\left(x_i^*(t), x_j^*(t)\right)\right)\\
&= 0.
\end{aligned}$$

Also,

$$\begin{aligned}
\lim_{t\to\infty}\tilde{x}_k(t) &= \lim_{t\to\infty}\left(x_k(t) - x_k^*(t)\right)\\
&= \lim_{t\to\infty}\left(x_k(t) - \hat{x}_k^*(t) + \hat{x}_k^*(t) - x_k^*(t)\right)\\
&= \lim_{t\to\infty}\left(\hat{x}_k(t) + \hat{x}_k^*(t) - x_k^*(t)\right) = 0.
\end{aligned}$$

Note that, by our definition of the system, $\hat{x}_1^* = x_1^*$ and $\hat{x}_2^* = x_2^*$. This implies that $\lim_{t\to\infty}\tilde{x}_1(t) = \lim_{t\to\infty}\tilde{x}_2(t) = 0$. Then, by the topological properties of $G$, the topological ordering of vertices, and induction, this implies that, $\forall k \in N$ such that $k \geq 3$, $\exists(i,j)$ such that $k > i$, $k > j$, $\{(v_k, v_i), (v_k, v_j)\} \subset E$, and $\lim_{t\to\infty}\tilde{x}_i(t) = \lim_{t\to\infty}\tilde{x}_j(t) = 0$. Hence, $\lim_{t\to\infty}\tilde{X}(t) = 0$.

## IV. Circle-Circle Intersection Solutions

Here, we describe how each agent $k$ determines its desired location and dynamics. As discussed in Section II, for each pair $(v_i, v_j) \in E$, there is a constant $d_{ij}$ such that the formation is persistent if $\|x_i(t) - x_j(t)\| = d_{ij}\ \forall t \in T$. The leader and first-follower agents attempt to satisfy this by stabilizing to their desired states. By our assumptions, the error of the leader and first-follower exponentially stabilizes to zero as $t \to \infty$.

Each additional follower agent employs solutions to the circle-circle intersection problem to determine their desired states and their dynamics. Two circles whose centers are at $x_i(t) \in \mathbb{R}^2$ and $x_j(t) \in \mathbb{R}^2$ with radii of $r_i$ and $r_j$ respectively intersect at the point $f_k(x_i(t), x_j(t))$ defined by

$$\begin{aligned}
f_k(x_i(t), x_j(t)) = \frac{x_i(t) + x_j(t)}{2} + \frac{1}{2\|x_j(t) - x_i(t)\|^2}\cdot\\
\Bigg(\left(r_i^2 - r_j^2\right)\left(x_j(t) - x_i(t)\right)\\
+ Q\left(x_j(t) - x_i(t)\right)\cdot\\
\sqrt{\left(\left(r_i + r_j\right)^2 - \|x_j(t) - x_i(t)\|^2\right)}\cdot\\
\sqrt{\left(\|x_j(t) - x_i(t)\|^2 - \left(r_j - r_i\right)^2\right)}\Bigg)
\end{aligned}$$

where

$$Q_1 = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}, \quad Q_2 = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix},$$

and where $Q = Q_1$ or $Q_2$. The choice of $Q$ selects which intersection point is desired, since there are (typically) two solutions to the circle-circle intersection problem. For each agent $k$ such that $f_k$ is defined, the appropriate equation is used to define its desired state.

The radii $r_i$ and $r_j$ are defined by the desired network trajectory $X^*$. For each agent $k \in N$ such that $\{(v_k, v_i), (v_k, v_j)\} \subset E$, $r_i = d_{ki}$ and $r_j = d_{kj}$. Therefore, $x_k^*(t) = f_k(x_i^*(t), x_j^*(t))$. However, when operating on $x_i(t)$ and $x_j(t)$, there is a discontinuity when $x_i(t) = x_j(t)$, i.e. when the circles either do not intersect or completely overlap each other. There are complex solutions when $\|r_i + r_j\|^2 < \|x_j(t) - x_i(t)\|^2$ or $\|r_j - r_i\|^2 > \|x_j(t) - x_i(t)\|^2$. The first condition indicates that the centers of the circles are too far apart to intersect. The second condition implies that one circle is inside the other so that they do not intersect. Since we can assemble these formations with arbitrary initial error [8], we place an upper bound on our initial error $\epsilon$ such that we operate within a sufficiently tight neighborhood of $X^*$ such that these conditions do not occur.

## V. Results

In this section, we present simulation results for the control strategy using estimations of the desired states presented in Section III-C, as well as implementation results with the prototype network of mobile robots.

### A. Simulation Scenarios

First, we present a scenario of the NASA project. Using software based on previous work [7], [8], along with a Graphical User Interface (GUI) for entering formation positions, we assume that the geologists specify a hexagonal formation of $n = 7$ robots to track a moving terrain feature with their sensors. A minimally persistent formation is determined such that, for each edge $(v_i, v_j) \in E$, $d_{ij} = 10$ m. The software automatically configures the robots for assembling the formation. The robots then assemble this formation and begin collecting data. During the data collection, the terrain feature begins moving with a velocity of $[1, 1]^T$. The leader and first-follower begin moving in order to track this motion, defining a desired velocity of $\dot{x}_i^* = [1, 1]^T$, $\forall i \in N$. For each agent $i$, we assume that $-K_i = -I$.

The simulation results for this motion are shown for three different initial error assumptions. First, it is assumed that the initial error of each agent is bounded to within 2 m. Then, we perform two simulations to test the "robustness" of the control laws defined by $f_k$ when the conditions discussed in Section IV occur. In one simulation, it is assumed that all the agents are very close initially to the initial desired state of the leader, causing $x_i(0)$, $x_j(0)$ to be very close to each other $\forall (i, j) \in N \times N$. In the last simulation, we assume that the initial position of each agent is within 20 m of the initial desired leader state. These last two scenarios allow sufficient error to produce complex results for the circle-circle intersection solutions.

### B. Simulation Results

Fig. 1 shows the results when the error of each agent is initially bounded to 2 m. As the formation moves, the error decreases. Fig. 1(a) shows the trajectory. In this figure, dotted lines represent the desired trajectory, while solid lines represent the actual trajectory. Arrows between the states correspond to the edges of $G$. Fig. 1(b) shows the errors of each agent with respect to their desired states, which approach zero as $t \to \infty$. Fig. 1(c) shows the errors of each constraint of the network, which also approach zero as $t \to \infty$.

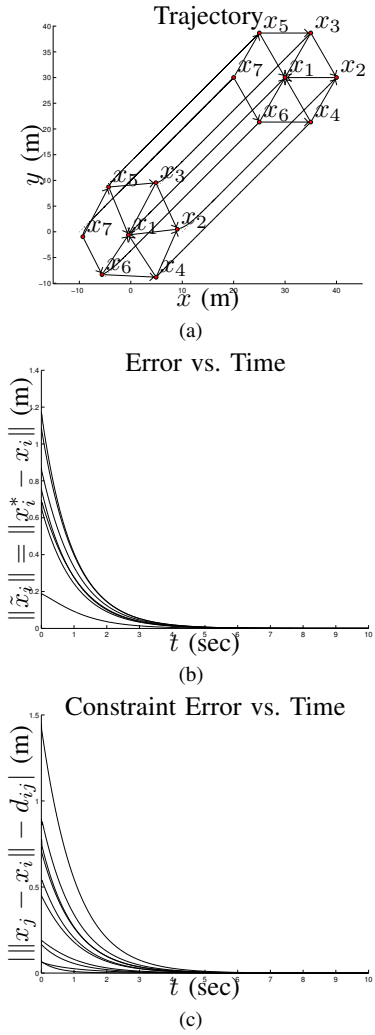


Fig. 1. The network trajectory, network error, and constraint errors. Here, we define an initial error bound of 2 m for each agent. As the agents move, the error in the formation stabilizes to zero as $t \to \infty$. Fig. 1(a) shows the formation trajectory. Fig. 1(b) shows the errors of each agent with respect to their desired states, and Fig. 1(c) shows the errors of each agent in satisfying their constraints. All errors stabilize to zero as $t \to \infty$.

In the next simulation, we force each agent's initial state at time $t = 0$ to be within $10^{-60}$ m of $x_1^*(0)$. This can produce complex results and large errors in the early portions of the formation trajectory. However, the errors still stabilize to zero as $t \to \infty$. Fig. 2 shows the corresponding results of this scenario.

To further test the robustness of the dynamics defined by $f_k$ when it produces complex results, we enlarge the bound on the initial error of each agent to 20 m. Fig. 3 shows the real portion of the results for this scenario. As seen in Fig. 3(b), this produces large initial errors, as well as complex results. However, using these dynamics, the errors

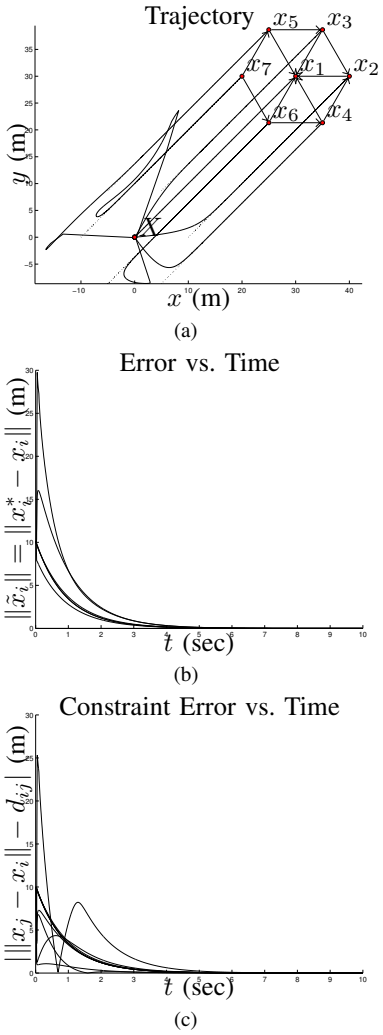


Fig. 2. The network trajectory, network error, and constraint errors. Here, the agents all begin initially within $10^{-60}$ m of $x_1^*(0)$. This can result in large errors early in the trajectory. However, the error still approaches zero as $t \to \infty$.

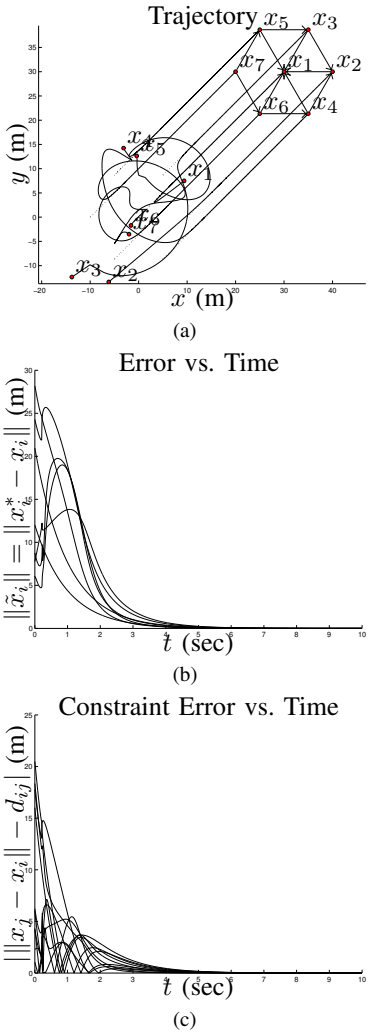


Fig. 3. The network trajectory, network error, and constraint errors. Here, the agents are all within 20 m of $x_1^*(0)$, producing complex results for $f_k$. Here, the real parts of the solution are plotted. However, the trajectory error still approaches zero as $t \to \infty$.

still stabilize to zero as $t \to \infty$.

### C. Implementation Results

For the pre-Antarctic stages of this project, we use a prototype network. The control strategy and equations presented in Sections III and IV are implemented with only the *relative* positions of the other robots. As such, these relative positions are the only sensor information available to the follower agents in the network. Only the leader and first-follower have access to the relative position of their desired trajectories.

For this demonstration, we choose $d_{ij} = 5$ m $\forall (v_i, v_j) \in E$. After the triangle is assembled, the robots begin moving while maintaining formation, as shown in Fig. 4. The robots maintain a triangle formation during and after the formation motion. The error is within the hardware limitations of the network, approximately 2 m for each robot. A similar formation motion is shown with a "diamond" formation in Fig. 5. Fig. 5(a) shows the formation before assembly. Fig. 5(b) shows the assembled formation. Fig. 5(c) shows the formation after a formation motion.

## VI. Conclusions and Future Work

This paper presents a method for maintaining persistent formations during motion of the formation. This allows our prototype network to move from one location to another while staying in a user-defined formation. Currently, range limitations in the prototype network's communications equipment prevent a large-scale demonstration using the full network and a large translation of the formation. Future work will address this issue, allowing the network to deploy, assemble, and move large formations longer distances using the methods presented here.

## VII. Acknowledgements

This was supported under a contract with NASA.

(a) $t = 0$ sec

(b) $t = 27$ sec

(c) $t = 71$ sec

Fig. 4. Triangle formation motion.

(a) $t = 0$ sec

(b) $t = 207$ sec

(c) $t = 281$ sec

Fig. 5. Diamond formation motion.

## REFERENCES


[1] S. Williams, A. Viguria, and A. M. Howard, "A robotic mobile sensor network for achieving scientific measurements in challenging environments," *Proceedings of the Earth Science Technology Conference*, 2008.

[2] J. Desai, J. Ostrowski, and V. Kumar, "Control of formations for multiple robots," in *Proceedings of the IEEE Int. Conf. Robotics and Automation*, May 1998, pp. 2864–2869.

[3] P. Ogren, M. Egerstedt, and X. Hu, "A control lyapunov function approach to multi-agent coordination," *IEEE Transactions on Robotics and Automation*, vol. 18, no. 5, pp. 847–851, Oct. 2002.

[4] G. A. Kaminka and R. Glick, "Towards robust multi-robot formations," *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 582–8, May 2006.

[5] L. Vig and J. A. Adams, "Multi-robot coalition formation," *IEEE Transactions on Robotics*, vol. 22, no. 4, pp. 637–49, August 2006.

[6] C. Doake, H. Corr, H. Rott, P. Skvarca, and N. Young, "Breakup and conditions for stability of the northern larsen ice shelf, antarctica," *Nature*, vol. 391, no. 6669, pp. 778–780, Feb. 1998.

[7] B. S. Smith, M. Egerstedt, and A. Howard, "Automatic generation of persistent formations for multi-agent networks under range constraints," in *Proceedings of the First International Conference on Robot Communication and Coordination*, 2007.

[8] ——, "Automatic deployment and formation control of decentralized multi-agent networks," in *Proceedings of the IEEE International Conference on Robotics and Automation*, May 2008, pp. 134–139.

[9] R. Olfati-Saber and R. Murray, "Graph rigidity and distributed formation stabilization of multi-vehicle systems," *41st IEEE Conference on Decision and Control. Proceedings*, vol. 3, pp. 2965 – 71, 2002.

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

[11] T. Eren, W. Whiteley, A. S. Morse, B. D. Anderson, and P. N. Belhumeur, "Information structures to secure control of globally rigid formations," *Proceedings of the 2004 American Control Conference*, vol. 6, pp. 4945–4950, 2004.

[12] M. Ji and M. B. Egerstedt, "Distributed coordination control of multiagent systems while preserving connectedness," *IEEE Transactions on Robotics*, vol. 23, no. 4, pp. 693–703, Aug. 2007.

[13] S. Sandeep, B. Fidan, and C. Yu, "Decentralized cohesive motion control of multi-agent formations," in *14th Mediterranean Conference on Control and Automation*, Jun. 2006, pp. 1–6.

[14] J. M. Hendrickx, B. D. O. Anderson, J.-C. Delvenne, and V. D. Blondel, "Directed graphs for the analysis of rigidity and persistence in autonomous agent systems," *International Journal of Robust and Nonlinear Control*, 2000.

[15] T. Eren, W. Whiteley, B. D. Anderson, A. S. Morse, and P. N. Belhumeur, "Information structures to secure control of rigid formations with leader-follower architecture," *Proceedings of the 2005 American Control Conference*, vol. 4, pp. 2966–2971, Jun. 2005.

[16] G. Laman, "On graphs and rigidity of plane skeletal structures," *Journal of Engineering Mathematics*, vol. 4, no. 4, pp. 331–340, October 1970.