Formation Reconfiguration for Mobile Robots with Network Connectivity Constraints

Leenhapat Navaravong, Zhen Kan, John M. Shea, and Warren E. Dixon, University of Florida

Abstract

Future systems of networked autonomous vehicles, such as unmanned aerial or ground vehicles, may rely on peer-to-peer wireless communication to coordinate their actions. The physical formation of the network may need to be reconfigured at times based on the specified missions. However, reconfiguring the physical formation also impacts the link connectivity and hence the connectivity of the network. If the network is partitioned, the autonomous vehicles can no longer coordinate their movements, and the mission may fail. In this article, we discuss techniques to transform the formation of a system of autonomous vehicles while preserving network connectivity. Several different approaches to address this problem are presented, with the focus on a method that utilizes ideas from routing packets in networks. We also briefly discuss the problem of formation selection and give an example of formation optimization in which communication costs are minimized under constraints on preserving network connectivity and the amount of movement required.

Systems of autonomous vehicles under cooperative control provide versatile platforms for commercial and military applications. For instance, [1] provides a list of "some of the main applications for cooperative control of multivehicle systems." This list includes:

- **Military systems:** formation flight, cooperative classification and surveillance, cooperative attack and rendezvous, and mixed initiative systems
- **Mobile sensor networks:** environmental sampling and distributed aperture observing
- **Transportation systems:** intelligent highways and air traffic control

These types of tasks usually require that the vehicles coordinate their actions; thus, the vehicles must be able to exchange information over some form of communications network. Thus, in many applications, maintaining network connectivity during formation control will be an important issue. For convenience of discussion, we use the terms *mobile robots* or *robots* to refer to autonomous vehicles under cooperative control, such as aerial, underwater, surface, and space vehicles.

For most applications, communications will be over a wireless network in which the communications links between robots are dependent on the propagation of electromagnetic signals between the robots. Because electromagnetic power density decreases with distance, the network topology is highly dependent on the physical formation of the system. Thus, formation control techniques must be designed that can maintain network connectivity while achieving the desired formation goals.

In this article, we discuss the problem of formation control with network connectivity (FC+NC) constraints or goals, hereafter referred to as FC+NC problems. Our focus is on techniques that utilize networking concepts to address this important topic. We first provide an overview of the prior research on this topic, which has primarily come from the controls community. Then we discuss some of our recent research, which applies ideas from routing in computer networks and poses formation control problems to minimize communication costs. Finally, we discuss open topics of research.

Overview of Research from Controls Community

Overviews of techniques for formation control (without necessarily including network considerations) are given in [1–3]. Some of the main approaches to formation control are given in [2] as leader-follower, virtual structure, and behavior-based. Another list of approaches is given in [1] as optimization-based, potential-field solutions, string-stability-based, and swarming. One of the most widely used approaches in formation control is to use artificial potential fields to guide the movement of the robots. Attractive potential fields are centered at the goal locations, and repulsive potential fields are generated around obstacles. Driven by the negative gradient of the potential field, each mobile robot will converge to a minimum of the potential field, which is typically the desired final position. An example of the generated artificial potential field is shown in Fig. 1 in which the destination is assigned a minimum potential value, and the obstacle is assigned a maximum potential value.

In [2, Section III], Chen and Wang discuss the research (up

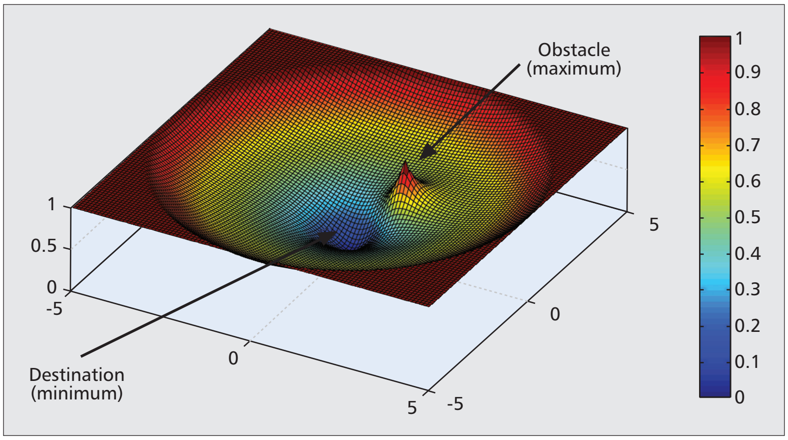

Figure 1. *An example of the artificial potential field generated for a disk-shaped workspace with destination at the origin and an obstacle located at $[1, 1]^T$.*

to 2005) on the impact of network connectivity on the analysis of the stability and controllability of formations of robots. The papers cited in that section represent some of the earliest work on FC+NC problems. Much of the work in this area, as well as later work on FC+NC problems, utilizes tools from graph theory, in particular, algebraic graph theory. Because of the importance of this approach to FC+NC problems, we give an overview of this topic here based on [2, references therein].

The network connecting the robots can be represented by a time-varying graph $\mathcal{G}(t)$. For simplicity, we consider the case where the graph is undirected and suppress the time dependence. Thus, $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ is the set of vertices (representing the robots) and $\mathcal{E}$ is the set of edges connecting the vertices in $\mathcal{V}$. An edge $(x, y) \in \mathcal{E}$ is an unordered pair which specifies that a communication link exists between x and y; x and y are said to be *adjacent* or *neighbors*, and this relation is denoted $x \sim y$.

The relation between the physical positions of the mobile robots and the network connectivity depends on the nature of the communication links connecting the robots. Most of the work on FC+NC problems assumes that the communication links follow a homogeneous protocol model, in which two robots can communicate if they are within a specified maximum communication range and cannot communicate if they are outside of that range. We assume the use of this homogeneous protocol model in this article. This model may accurately model communications in many systems of unmanned aerial vehicles (UAVs) working above obstructions; however, shadowing and multipath fading make this model inaccurate in many scenarios involving ground vehicles or operations in urban environments.

The edge connection information can be collected into a matrix called the *adjacency matrix*, $A(\mathcal{G})$, which is defined by

$$A_{i,j}(\mathcal{G}) = \begin{cases} 1, & i \sim j \\ 0, & \text{otherwise.} \end{cases}$$

The connectivity of the graph $\mathcal{G}$ can be determined from $A(\mathcal{G})$ by computing eigenvalues of the graph Laplacian, $L(\mathcal{G})$. Let $\Delta(\mathcal{G})$ be a diagonal matrix, in which the (i, i) entry is the number of neighbors of vertex i (also known as the *degree* or *valency* or vertex i); the (i, i) entry is the sum of the elements in the ith row or column of $A(\mathcal{G})$. Then $L(\mathcal{G}) = \Delta(\mathcal{G}) - A(\mathcal{G})$.

The Laplacian is symmetric and positive semidefinite. Let $\lambda_i(L)$ be the ith smallest eigenvalue. Then $\lambda_1(L) = 0$, and the multiplicity of the eigenvalue zero is equal to the number of connected components in the graph (i.e., the number of connected subgraphs that do not share connections with each other). Thus, for a network to be connected, there must be only one zero eigenvalue. The second smallest eigenvalue, $\lambda_2(L)$, is called the Fiedler value and gives a quantitative measure of how connected the graph is. The associated eigenvector can be used to determine a set of links that if removed will cause the network to partition.

In the early work on FC+NC problems, the network graph was assumed fixed, and the impact of the network on the stability or controllability under particular control strategies was evaluated. For instance (again see [2, references therein], if the system can be modeled using first order dynamics, where the dynamics of agent i are given by

$$\dot{x}_i = u_i,$$

where x_i is the state of agent i, and the control law at agent i averages the values from its neighbors,

$$u_i = -\frac{1}{\Delta_{ii}} \sum_{j \sim i} (x_i - x_j),$$

the controllability of the system of robots is determined by the interconnection graph. Moreover, the algebraic representation of the graph's interconnections can be used to give a simple form for the whole system's dynamics as

$$\dot{x}_i = -\Delta^{-1/2} L \Delta^{-1/2} x.$$

Perhaps surprisingly, connectivity can decrease controllability, and a complete graph with this update law can be shown to be uncontrollable. On the other hand, results using algebraic graph theory indicate that formation stability may easily be achievable for the complete graph.

Although the earliest work that identified connectivity as a control objective was published in 1999, most of the work on formation control with connectivity constraints or goals has been published in 2005 and later [3, references therein]. Much of the control theory work on FC+NC problems focuses on formation control with maintenance of existing communication links [1, 4]. This can be achieved in formation control systems using the artificial potential field approach by treating network connectivity as an artificial obstacle [4].

Another branch of work on FC+NC problems focuses on the design of controllers to enhance some measure of network connectivity or on maintaining network connectivity (which is less restrictive than maintaining link connectivity) during formation control. The Introduction of [3] gives a good overview of this work. Here, we give a brief overview of how algebraic graph theory is used in these works.

As previously discussed, the Fiedler value, $\lambda_2(L)$, gives an indication of the connectivity of a network. Thus, optimization techniques can be used to maximize $\lambda_2(L)$. However, "the Fiedler value is a nondifferentiable function of the Laplacian matrix" [3], which makes it difficult to use in control strategies. However, several approaches have been developed, as discussed in [3]. One alternative to using the Fiedler value as a measure of network connectivity is to use a sum of powers of the adjacency matrix,

$$S_K(\mathcal{G}) = \sum_{k=0}^{K} A(\mathcal{G})^k.$$

The i, jth entry of S_K is the number of paths of length ≤ K between every pair of nodes in the graph. Thus, if $\mathcal{G}$ is connected, every entry of $S_{n-1}(\mathcal{G})$ will be non-zero, and $S_{n-1}(\mathcal{G})$ will be positive definite, where n is the total number of agents in the system. This sum-of-powers of the adjacency matrix can then be used to develop optimization-based controllers for connectivity maintenance, which are generally centralized. Changes in the topology can be accommodated through a controller that utilizes global techniques, such as gossip and auctions, to guarantee that link breakages will not disconnect the network.

These works develop an important principle that will make these problems more accessible to network researchers: the overall FC+NC problem can be decomposed such that network connectivity control is performed in the discrete space over the network graph, while motion control is performed in the continuous space using conventional control techniques, such as potential fields. Thus, networking researchers can contribute to the problems of formation control without having to become experts in motion control. This approach is demonstrated in the next section.

Networking Approaches to Formation Control

The approaches described earlier have several limitations. The approaches presented in [2] are primarily focused on the impact of a given network connectivity graph on the control algorithm. Those described in [1] are mostly focused on maintaining network connectivity during formation control. The approaches in [3] are designed to optimize connectivity or only allow limited reconfiguration of the formation. In these previous works, the absolute or relative poses of the agents are pre-specified. Thus, additional work on FC+NC problems is needed to address the following FC+NC problems:

- Techniques are needed to reconfigure a systems of mobile robots from arbitrary initial connected formations to arbitrary final connected formations, with no break in connectivity during the reconfiguration.
- Many systems may use robots that have identical capabilities. Such systems should use *anonymous reconfiguration*, in which the robots are identical and can take any position in the final topology.[1]
- Techniques are needed that allow optimization of a final connected formation to optimize measures such as communication costs or task-specific utility functions, while maintaining network connectivity and obeying constraints on the amounts of movement.

In [5], we propose a networking-based approach that can be used to address all three problems listed above. In this section, we present an overview of the work in [5] and explain how it can be used to simultaneously address problems 1 and 2. Later, we give an example that illustrates how these techniques can be used to address Problem 3; our example application minimizes communication costs under constraints on the amount of movement.

The key idea from [5] is that mobile robots can be treated as packets that are then routed through the network.[2] There are several issues that have to be solved in implementing this concept. The first is that robots can only be moved if they will not cause the network to partition and if they can be moved along paths that preserve their connectivity to the network. The second is that we wish to perform anonymous formation reconfiguration. That is, we do not wish to specify which nodes in the initial topology will take which positions in the final topology; rather, we only care that there is a mobile robot in each position specified in the final topology. To address these issues, we first invoke the separation principle described in [3]: the problems of network topology control and physical position are treated separately. Utilizing the navigation function of the controller developed in [4], the desired physical formation can be achieved once the network topology is a superset of the network topology in the desired physical formation. Here, we assume that the goal is to transform the network topology from the initial to the specified final topology and that there are physical position-control techniques that can move mobile robots to positions that correspond to the edges and vertices of the initial topology.

The two issues described above can be addressed by using a concept from computer networking called *prefix routing* [6–8]. For convenience of discussion, we refer to any member of a network as a "node" below. In the simplest form of prefix routing, one node is elected to be the root.[3] The neighbors of the root are labeled as children of the root in a prefix tree, or *trie*. Then those nodes label their children, and the process continues until each node has a unique label. In prefix routing, for a node to send a packet to another node, it first determines the destination node's prefix label, and then uses maximal prefix matching logic to route the packet to its destination. (We give an example of this routing approach in the context of repositioning mobile robots below.)

In [5], techniques based on prefix routing are introduced as a way to transform an initially connected network topology into a desired network topology, while maintaining network connectivity and allowing anonymous reconfiguration. We summarize the basic approach from [5] here. We first note that for arbitrary initial and desired final topologies, there are always spanning trees that eliminate loops while preserving connectivity. Thus, in [5], we assume that the final topology is a tree.

The first step required by [5] is for one robot to be selected (for instance, through distributed consensus) as the root of a prefix routing trie. The root is assigned the label 0. The children of a node with label L_i will be assigned the labels $L_i \oplus$ "1", $L_i \oplus$ "2", …, where $\oplus$ denotes string concatenation. Thus, the children of the root are labeled 01, 02, … . Every robot is assigned a label according to its position in the prefix trie (because of loops in the initial graph, the labeling is not unique). An example of this is shown in the initial tree topology shown in Fig. 2a. Note that the initial network topology before assigning prefix labels may not be a tree. For example, it may be that 01 ~ 02 and 0111 ~ 0211 in the initial network topology. However, after assigning prefix labels, the network will be treated as a trie (note that [6–8] provide techniques to use multiple overlayed tries to utilize the other connections in the network, thereby improving routing efficiency). Note that the length of the label corresponds to its depth in the trie.

Once the trie is established, the network topology can be reconfigured by moving some of the mobile robots. Consider the following two observations:

- Suppose two robots, A and B, share a communications link, and C is another robot that is moving along the line connecting A to B. Then at all points between A and B, C will

[1] *Semi-anonymous techniques should be created for general heterogeneous networks in which some robots are identical.*

[2] *Our approach to reconfiguring the network topology can be considered to be path planning. Considerations like vehicle dynamics and collision avoidance need to be handled by the physical control algorithms.*

[3] *Many distributed consensus algorithms are available for this purpose.*

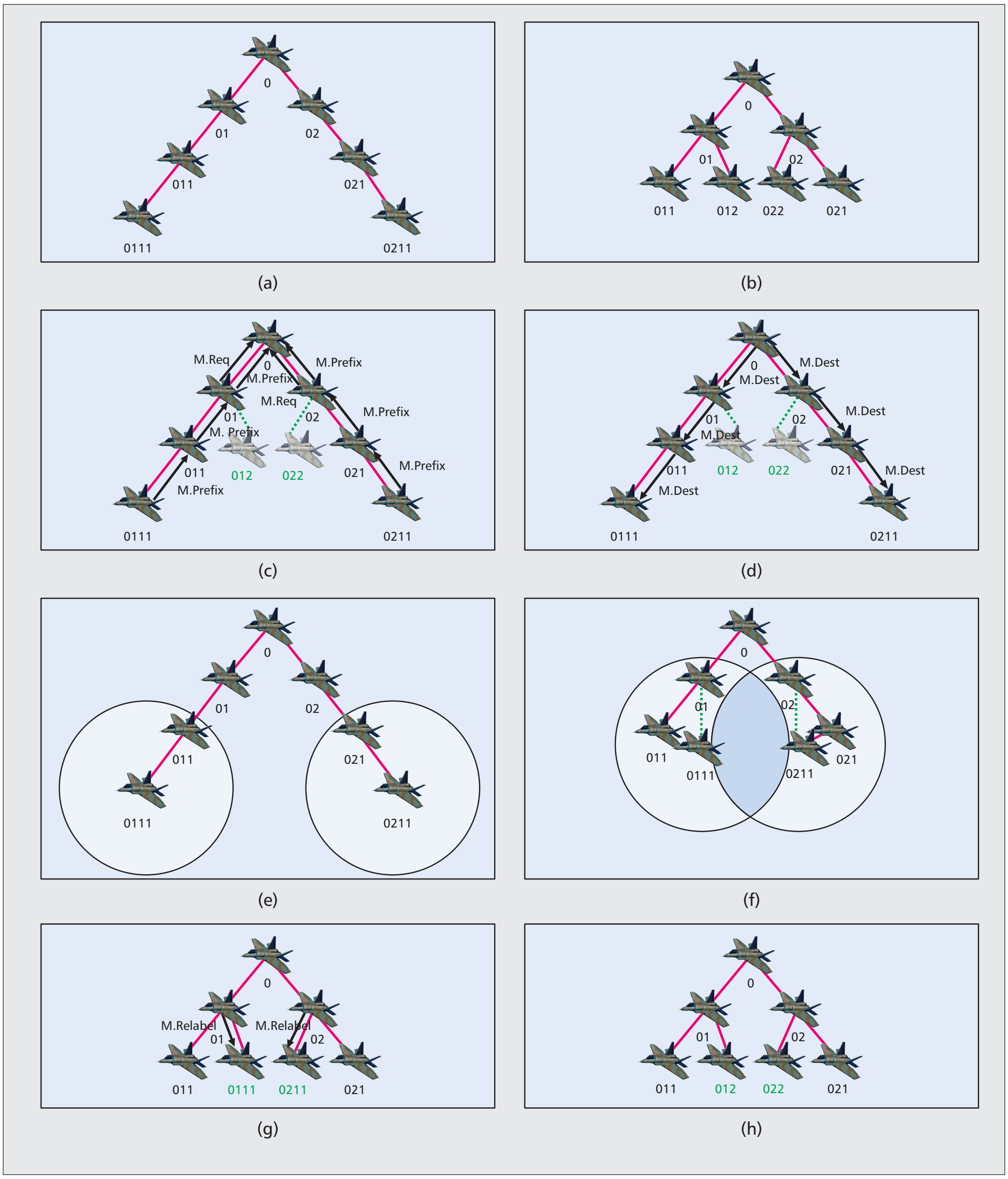

Figure 2. *Network topology reconfiguration: a) initial network topology; b) desired network topology; c) mobile robots 0111 and 0211 send messages including their prefix labels, and mobile robots 01 and 02 send requesting messages to the root (robot 0); d) the root assigns destinations for mobile robots 0111 and 0211 and sends a messge including their destination prefix labels to each of them; e) mobile robots 0111 and 0211 start moving to their destinations by first moving toward mobile robots 011 and 021, respectively; f) mobile robots 0111 and 0211 are now able to communicate with requesting mobile robots 01 and 02, respectively; g) requesting mobile robots 01 and 02 send messages to robots 0111 and 0211 to relabel them as 012 and 022, respectively; h) the desired network topology is achieved.*

be closer to A and B than A is to B. Under the protocol model, C will have communication links to both A and B. We can visualize this as C moving along the edge connecting A and B in the network graph. Thus, robots moving along the positions corresponding to the edges of the network topology graph will not become disconnected.

- In a tree topology, leaf nodes are those nodes that do not connect other parts of the network. Thus, leaf nodes can be repositioned without affecting network connectivity, provided those leaf nodes maintain connectivity to the rest of the network.

Based on these observations, topology reconfiguration can be achieved by "routing" leaf nodes from the initial trie to a position where they are needed in the final trie. Here, by routing, we mean that the mobile robots associated with those nodes follow physical paths that are approximately given by the lines connecting the robots associated with the nodes along the path through the graph from their initial position to their position in the desired trie. A node that needs to be repositioned but that is not a leaf node must wait for all of its descendants to move to another part of the graph before that node can be repositioned.

The remaining problem is to determine which nodes should be repositioned to transform the graph from the initial trie topology to the desired final trie topology. In [5], this is done using a centralized approach, in which the root will determine

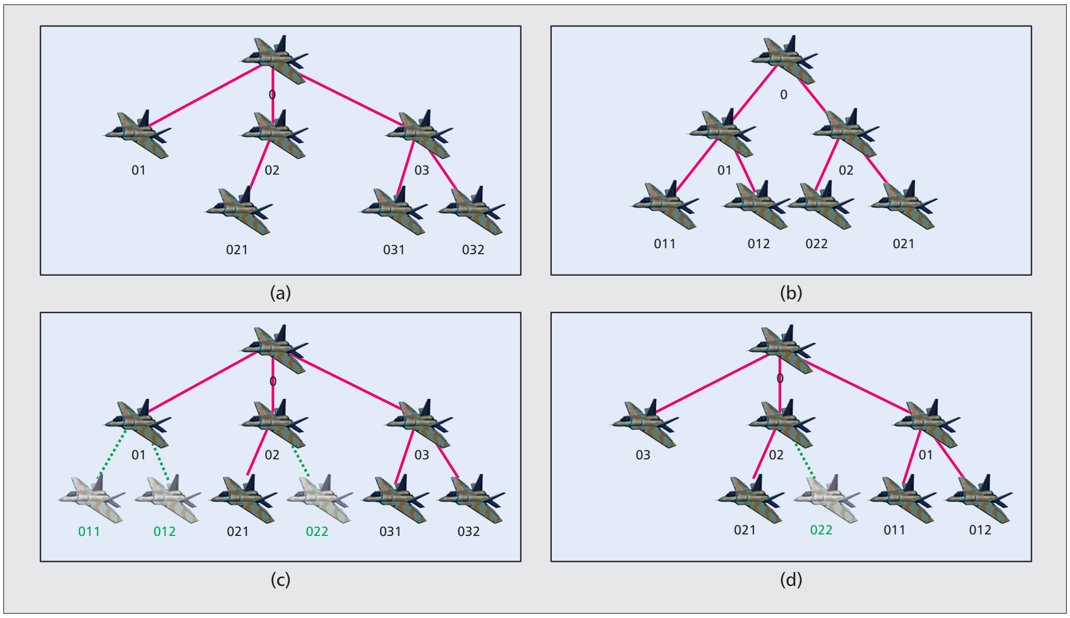

Figure 3. *Example of relabeling to reduce the total amount of movement needed to achieve the desired topology: a) initial network topology; b) desired network topology; c) initial network topology with missing nodes illustrated by dim UAVs; d) after swapping the prefix labels of mobile robots in the leftmost and rightmost branches, network topology with missing nodes illustrated by dim UAVs.*

how the nodes should be repositioned, and the root additionally coordinates the movement of the mobile robots. The nodes that are repositioned are those nodes in the initial trie that are in positions where there is no node in the desired trie. We call these *extra nodes*. These nodes will be routed through the trie to positions where the desired trie requires a node but there is no node in the initial trie. We call these *missing nodes*. The assignment of extra nodes to take the place of missing nodes is determined by the root and coordinated through an exchange of messages, as described in [5].

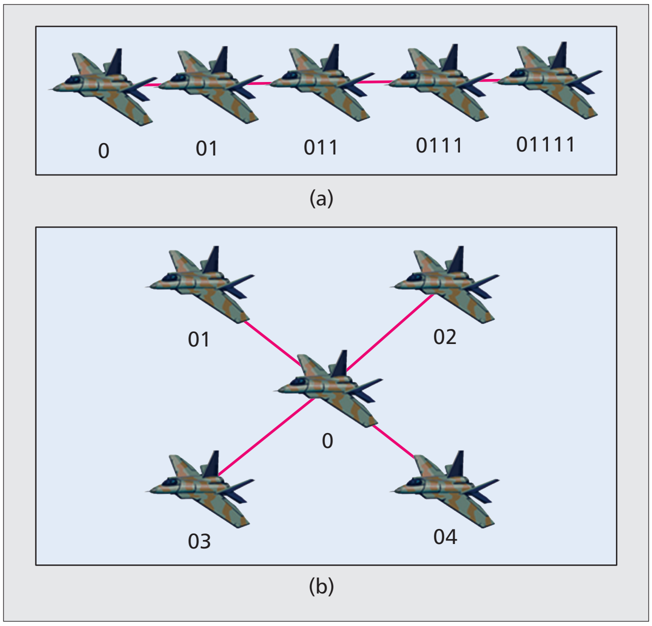

Figure 4. *Conversion from linear configuration to star configuration: a) a system of UAVs with linear network configuration due to consecutive launch; b) tasks may use a star configuration for a system of UAVs to achieve larger coverage area while maintaining network connectivity.*

To illustrate this approach, we provide a simple example, which is illustrated in Fig. 2. The initial and desired network tries are shown in Figs. 2a and 2b. After prefix assignment is complete, each node will check if its prefix label is present in the final trie. If not, the node is an extra node and it will send a message **M.Prefix** with its prefix to the root, as shown in Fig. 2c. Each node whose prefix exists in the final topology will check to determine if any of the nodes listed as its children in the final trie are missing in the initial trie. If so, then these are missing nodes, and the node sends a message **M.Req** to the root requesting that an extra node be repositioned to fill in the missing node. In the example, nodes 01 and 02 are missing children with labels 012 and 022, respectively, in the desired network topology, so both 01 and 02 send **M.Req** messages to the root, as shown in Fig. 2c. Nodes 0111 and 0211 are extra nodes and send **M.Prefix** messages including their prefix label to a root.

After the mobile robot that is the root node gathers all the messages from the mobile robots in the initial network, it uses an algorithm ([5]) to determine the destination for each mobile robot that needs to be repositioned. The root then sends a message with a destination prefix label to each extra node, as shown in Fig. 2d. Extra nodes that have no descendants will use maximal prefix matching logic [6–8] to route themselves to their destination position. Extra nodes that have descendants will wait for all their descendants to move up to connect to another part of the network before moving themselves. After an extra node reaches its respective requesting node, its prefix label will be relabeled by its requesting node. After all the extra nodes fill in for all the missing nodes, the desired network topology is achieved. This is illustrated in Figs. 2g and 2h, in which the extra nodes 0111 and 0211 are relabeled by the requesting nodes 01 and 02, respectively, to achieve the desired network configuration.

The basic algorithm described above is extended in several ways to reduce the total amount of node movement, which generally correlates with energy expended, required to reconfigure the network topology. For example, relabeling the branches of the tree can reduce the amount of movement required, as illustrated in Fig. 3. The initial and desired net-

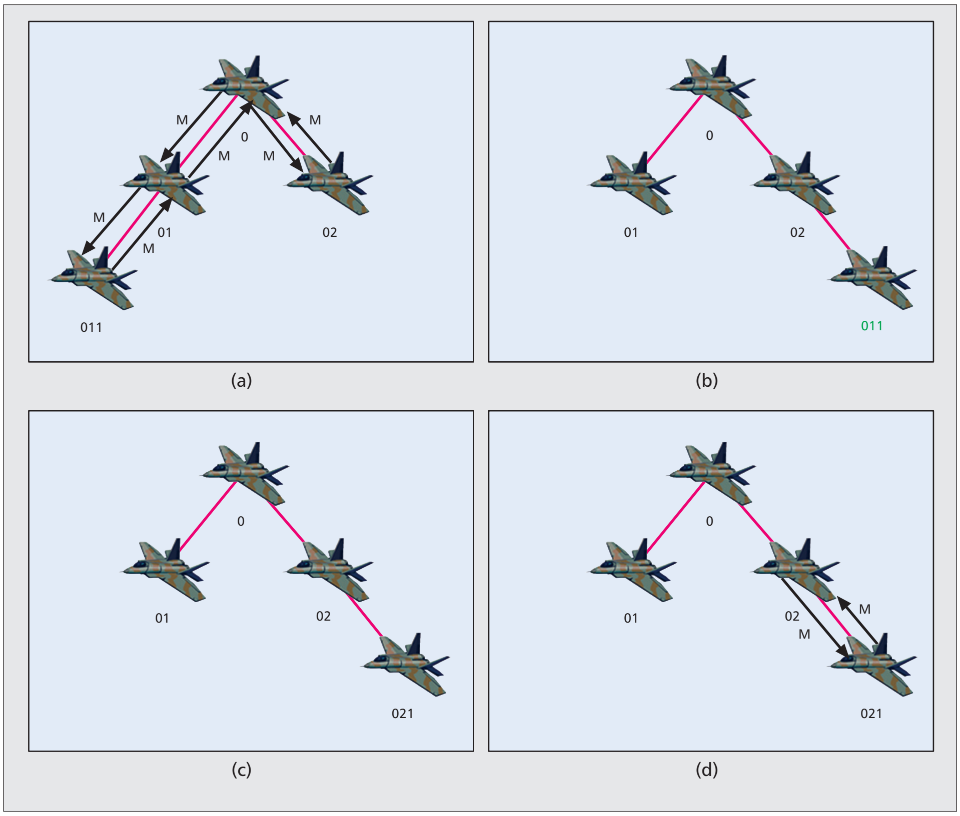

Figure 5. *Aggregate flow minimization by topology reconfiguration: a) when mobile robots 011 and 02 exchange information, multihop routing will cause the same message to be retransmitted by the intermediate robots along the route from 011 to 02; b) to reduce the aggregate network traffic, mobile robot 011 is repositioned to be a neighbor of mobile robot 02; c) after mobile robot 011 is repositioned, its label is changed to 021; d) the aggregate data flow is reduced since mobile robots with original labels 011 and 02 can now communicate directly.*

work topologies for this example are shown in Figs. 3a and 3b, respectively. The initial prefix labeling shown in Fig. 3a will lead to three extra nodes that need to be repositioned: 03, 031, and 032. However, if the prefix labeling is swapped between the leftmost and rightmost branches, as shown in Fig. 3d, only node 03 needs to be repositioned, and the total amount of movement required to achieve the desired network topology is reduced.

Formation Reconfiguration Applications

Earlier we considered a scenario in which the final formation is specified. In general, we can divide formation reconfiguration problems into two classes based on how the final formation is determined:

- For some applications, the formation to be achieved is specified by the application. For instance, for landing UAVs on a small runway, a linear formation may be required, while monitoring a fixed area may require a grid formation.
- In other applications, the formation may be selected during operations to optimize performance of the system. For example, the formation of UAVs may be adapted based on the positions of mobile ground vehicles being tracked or to minimize the communication costs needed to exchange information.

In each of these classes of formation control problems, networking constraints may play an important role. For example, in a search operation, a grid formation may be desired. However, if the vehicles separation is too large, network connectivity may break. Thus, another robot may be used to help maintain network connectivity across the system. An example of this scenario is illustrated in Fig. 4, in which a group of UAVs takes off from a launch area, causing them to initially be in a linear formation. To conduct a grid search while maintaining network connectivity, the formation is transformed to a star formation, with the center UAV acting as a network router. When the final formation is known, the network topology reconfiguration approach of [5] can be coupled with the physical formation control approach in [4] to achieve the desired formation while maintaining network connectivity.

An example of formation optimization is given in [9]. In the scenario considered, the flow of traffic among the robots is nonuniform: some pairs of robots have much higher traffic flows than other pairs of robots. This situation may arise during processes such as sensor fusion or data dissemination in peer-to-peer networks. Generally, messages sent from one mobile robot to another mobile robot in the network will have to be relayed by intermediate mobile robots. We call the total amount of transmission and retransmission the *aggregate traffic*. Clearly, the aggregate traffic will be highly dependent on the network topology; thus, the network topology can be optimized to minimize the aggregate traffic. For example, consider the system of UAVs shown in Fig. 5a. The UAVs with labels 011 and 02 are exchanging messages, but because of the network topology their traffic has to be relayed by every other node in the network. As shown in Fig. 5, node 011 can be repositioned to be a neighbor of node 02. Upon achieving its new position in the network topology, node 011 is relabeled as node 021, and the aggregate traffic is minimized — see Fig. 5c and Fig. 5d.

As noted in [9], reconfiguring the topology may require additional energy, and so constraints on movement may need to be considered. In [9], we consider the problem of optimizing the network topology under constraints on network connectivity and the amount of movement allowed at each node. In that work, we only consider the network topology, so the movement constraints are in terms of the number of edges along which a node allowed to move. In [9], an optimal search using the branch-and-bound technique and a greedy suboptimal approach are presented and compared in terms of performance and complexity.

Conclusion and Future Work

In this article, we have provided an overview of the problem of formation control with network connectivity (FC+NC) constraints or goals, with a special focus on the use of networking techniques in formation reconfiguration with networking constraints. Solutions to FC+NC problems may often be decomposed into parts that address network connectivity maintenance and parts that control the physical positions. Network connectivity can be treated using graphs. Prior work from the controls community (cf. [3]) has focused on controlling measures generated using algebraic graph theory, such as the Fiedler value or controlling the sum-of-powers of the adjacency matrix to be positive definite. We present recent research on using the concepts of prefix routing to reconfigure the network topology. In comparison to work from the controls community, topology reconfiguration based on prefix routing has several nice features. It can achieve arbitrary transformation of the network topology, is perfectly designed for anonymous reconfiguration, and is easily used in systems where the final topology is to be optimized under constraints on connectivity and movement.

Future work may provide performance improvements (e.g., minimizing the required movement) in the process of transforming from one connected initial topology to another connected initial topology, provide better connections between optimization of the network topology and physical formation, and introduce new optimization strategies that optimize utility functions for a task under constraints on network connectivity and amount of movement. Future work should also incorporate more realism into the physical and communications models, including taking into account the dynamics of the robots and the effects of those on communications, and incorporating more realistic channel models. For instance, the problem of determining and controlling connectivity under stochastic channel models may be a challenging area for additional research.

References

[1] R. M. Murray, "Recent Research in Cooperative Control of Multivehicle Systems," *J. Dynamic Systems, Measurement, and Control*, vol. 129, 2007, p. 571.

[2] Y. Q. Chen and Z. Wang, "Formation Control: A Review and A New Consideration," *Proc. IEEE/RSJ Int'l. Conf. Intelligent Robots and Systems*, Aug. 2005, pp. 3181–86.

[3] M. Zavlanos, M. Egerstedt, and G. Pappas, "Graph-Theoretic Connectivity Control of Mobile Robot Networks," *Proc. IEEE*, vol. 99, no. 9, Sept. 2011, pp. 1525–40.

[4] Z. Kan et al., "Ensuring Network Connectivity During Formation Control Using A Decentralized Navigation Function," *Proc. IEEE MILCOM*, San Jose, CA, Nov. 2010, pp. 531–36.

[5] L. Navaravong, J. M. Shea, and W. E. Dixon, "Physical- and Network-Topology Control for Systems of Mobile Robots," Proc. IEEE/AFCEA MILCOM, Baltimore, MD, 2011; http://wireless.ece.ufl.edu/jshea/pubs/BomMILCOM2011.pdf, pp. 1079–84.

[6] J. J. Garcia-Luna-Aceves and D. Sampath, "Efficient Multicast Routing in MANETs Using Prefix Labels," *Proc. IEEE Int'l. Conf. Computer Commun. and Networks*, San Francisco, CA, Aug. 2009, pp. 1–8.

[7] D. Sampath and J. J. Garcia-Luna-Aceves, "PROSE: Scalable Routing in MANETs Using Prefix Labels and Distributed Hashing," *Proc. IEEE Conf. Sensor Mesh, and Ad Hoc. Commun.*, Rome, Italy, June 2009, pp. 1–9.

[8] J. J. Garcia-Luna-Aceves and D. Sampath, "Scalable Integrated Routing Using Prefix Labels and Distributed Hash Tables for MANETs," *Proc. IEEE Int'l. Conf. Mobile Ad Hoc and Sensor Sys.*, Macau SAR, P.R.C., Oct. 2009, pp. 188–98.

[9] L. Navaravong et al., "Optimizing Network Topology to Reduce Aggregate Traffic in Systems of Mobile agents," *Proc. Int'l. Network Analysis Conf.*, Gainesville, FL, Dec. 2011; http://wireless.ece.ufl.edu/jshea/pubs/BomNetwork2011.pdf.

Biographies

LEENHAPAT NAVARAVONG (leenhapat@ufl.edu) received his B.E. degree in telecommunications from Shinawatra University, Pathumthani, Thailand, in 2007, and his M.S. degree in electrical and computer engineering from the University of Florida, Gainesville, in 2010, where he is currently pursuing a Ph.D. degree. He has been working in the Wireless Information Networking Group (WING) Laboratory at the University of Florida since 2009. His research interests are wireless communications and networking with applications to systems of autonomous vehicles.

ZHEN KAN (kanzhen0322@gmail.com) received his Ph.D. degree in 2011 from the Mechanical and Aerospace Engineering Department of the University of Florida. After completing his doctoral studies, he is working as a postdoctoral research fellow with the Air Force Research Laboratory (AFRL) at Eglin AFB and the University of Florida REEF. His current research interests include networked control systems, cooperative control of autonomous agents, vision-based estimation and control, and Lyapunov-based nonlinear control.

JOHN M. SHEA [S'92-M'99] (jshea@ece.ufl.edu) received his B.S. (with highest honors) in computer engineering from Clemson University in 1993, and his M.S. and Ph.D. degrees from Clemson University in 1995 and 1998, respectively. He is currently as associate professor of electrical and computer engineering at the University of Florida, where he is engaged in research on wireless communications and networking with emphasis on applications to systems of autonomous vehicles, cross-layer protocol design, physical-layer security, cooperative diversity techniques, and applications of error control coding. He has been an Editor for *IEEE Transactions on Wireless Communications* since 2008 and an Editor for *IEEE Wireless Communications* since 2010. He was an Associate Editor for *IEEE Transactions on Vehicular Technology* from 2002 to 2007. He was the Technical Program Chair for the Unclassified Program of MILCOM 2010.

WARREN E. DIXON [SM] (wdixon@ufl.edu) received his PH.D. degree in 2000 from the Department of Electrical and Computer Engineering from Clemson University. After working as a Eugene P. Wigner Fellow at Oak Ridge National Laboratory (ORNL), he joined the faculty of the University of Florida in the Mechanical and Aerospace Engineering Department in 2004. His main research interest has been the development and application of Lyapunov-based control techniques for uncertain nonlinear systems. He has published three books, an edited collection, nine chapters, and over 250 refereed journal and conference papers. His work has been recognized by the 2011 American Society of Mechanical Engineers (ASME) Dynamics Systems and Control Division Outstanding Young Investigator Award, 2009 American Automatic Control Council (AACC) O. Hugo Schuck Award, 2006 IEEE Robotics and Automation Society (RAS) Early Academic Career Award, and the 2001 ORNL Early Career Award for Engineering Achievement. He serves on several IEEE CSS and ASME technical committees, and is a member of numerous conference program and organizing committees, and conference editorial boards. He is currently or was formerly an Associate Editor for *ASME Journal of Dynamic Systems, Measurement and Control*, *Automatica*, *IEEE Transactions on Systems Man and Cybernetics: Part B Cybernetics*, and the *International Journal of Robust and Nonlinear Control*. He served as an appointed member to the IEEE CSS Board of Governors (BOG) in 2008. He currently serves as the Executive Director of the Executive Committee of the IEEE CSS BOG.