$$\begin{aligned}&\leq (\hat{k}_1+\epsilon_2)\left\|\Delta\mathbf{G}_2\mathbf{x}_3'+\Delta\mathbf{f}_2-\dot{\boldsymbol{\phi}}_\zeta\right\|+\left\|\mathbf{G}_1^T\mathbf{P}\right\|\|\Delta\mathbf{f}_1\|\\&+\left\|\dot{\mathbf{f}}_{2\zeta}\right\|+\left\|\ddot{\boldsymbol{\phi}}_{n\zeta}\right\|\leq s_1(\hat{k}_1+\epsilon_2)\|\mathbf{x}\|+\left\|\mathbf{G}_1^T\mathbf{P}\right\|\|\mathbf{G}_1\|\|\boldsymbol{\Theta}\|\|\mathbf{x}\|\\&+\sum_{\ell=0}^{\alpha_3}b_\ell\|\mathbf{x}\|^\ell+\sum_{\ell=0}^{\alpha_4}a_\ell\|\mathbf{x}\|^\ell\leq\sum_{\ell=0}^{\alpha_5}y_\ell\|\mathbf{x}\|^\ell\end{aligned}\tag{31}$$

where the positive constants $y_1 \triangleq s_1(\hat{k}_1+\epsilon_2)+\left\|\mathbf{G}_1^T\mathbf{P}\right\|\|\mathbf{G}_1\|\|\boldsymbol{\Theta}\|+a_1+b_1$ and $\alpha_5 \in max\{\alpha_3,\alpha_4\}, y_\ell \triangleq a_\ell+b_\ell, \ell=0,2,\ldots,\alpha_5$ are unknown and known respectively. According to assumption A3, (21), and (31), one can compute the upper bound of $\boldsymbol{\zeta}_2$ as

$$\begin{aligned}\|\boldsymbol{\zeta}_2\|&=\left\|\begin{bmatrix}\boldsymbol{\zeta}_2'\\\boldsymbol{\zeta}_2''\end{bmatrix}\right\|=\left\|\Delta\mathbf{f}_3+\Delta\mathbf{G}_3\mathbf{u}+\begin{bmatrix}\mathbf{G}_2^{-1}\boldsymbol{\zeta}_2'''\\\mathbf{0}\end{bmatrix}\right\|\\&\leq\|\Delta\mathbf{f}_3\|+\|\Delta\mathbf{G}_3\|\|\mathbf{u}\|+\left\|\mathbf{G}_2^{-1}\right\|\left\|\boldsymbol{\zeta}_2'''\right\|\\&\leq\sum_{\ell=0}^{\alpha_1}d_\ell\|\mathbf{x}\|^\ell+g_u\|\mathbf{u}\|+\left\|\mathbf{G}_2^{-1}\right\|\sum_{\ell=0}^{\alpha_5}y_\ell\|\mathbf{x}\|^\ell\\&\leq\sum_{\ell=0}^{\alpha_2}g_\ell\|\mathbf{x}\|^\ell+g_u\|\mathbf{u}\|\end{aligned}$$

where the positive constants $g_\ell \triangleq d_\ell+\left\|\mathbf{G}_2^{-1}\right\|y_\ell, 0\leq\ell\leq\alpha_2$ and $\alpha_2\in max\{\alpha_1,\alpha_5\}$ are unknown and known, respectively.

REFERENCES

[1] J. Y. Hung, W. Gao, and J. C. Hung, "Variable structure control: A survey," *IEEE Trans. Ind. Electron.*, vol. 40, no. 1, pp. 2–22, Feb. 1993.

[2] K. S. Kim, Y. Park, and S. H. Oh, "Designing robust sliding hyperplanes for parametric uncertain systems: A Riccati approach," *Automatica*, vol. 36, pp. 1041–1048, 2000.

[3] M.-L. Chan, C. W. Tao, and T. T. Lee, "Sliding mode controller for linear systems with mismatched time-varying uncertainties," *J. Franklin Inst.*, vol. 337, pp. 105–115, 2000.

[4] Y. Xia and Y. Jia, "Robust sliding-mode control for uncertain time-delay systems: An LMI approach," *IEEE Trans. Autom. Control*, vol. 48, no. 6, pp. 1086–1092, Jun. 2003.

[5] A. J. Koshkouei and A. S. I. Zinober, "Sliding mode state observation for non-linear systems," *Int. J. Control*, vol. 77, pp. 118–127, 2004.

[6] P. Park, D. J. Choi, and S. G. Kong, "Output feedback variable structure control for linear systems with uncertainties and disturbances," *Automatica*, vol. 43, pp. 72–79, 2007.

[7] H. H. Choi, "LMI-based switching surface design method for a class of mismatched uncertain system," *IEEE Trans. Autom. Control*, vol. 52, no. 4, pp. 736–742, Apr. 2007.

[8] F. Castaños and L. Fridman, "Analysis and design of integral sliding manifolds for systems with unmatched perturbations," *IEEE Trans. Autom. Control*, vol. 51, no. 5, pp. 853–858, May 2006.

[9] I. A. Shkolnikov and Y. B. Shtessel, "Tracking on a class of nonminimum-phase systems with nonlinear internal dynamics via sliding mode control using method of system center," *Automatica*, vol. 42, pp. 837–842, 2002.

[10] C. M. Kwan, "Sliding mode control of linear systems with mismatched uncertainties," *Automatica*, vol. 31, pp. 303–307, 1995.

[11] W.-J. Cao and J.-X. Xu, "Nonlinear integral-type sliding surface for both matched and unmatched uncertain systems," *IEEE Trans. Autom. Control*, vol. 49, no. 8, pp. 1355–1360, Aug. 2004.

[12] X.-G. Yan, S. K. Spurgeon, and C. Edwards, "Decentralized sliding mode control for nonminimum phase interconnected systems based on a reduced-order compensator," *Automatica*, vol. 42, pp. 1821–1828, 2006.

[13] M. Krstić, J. Sun, and P. V. Kokotović, "Robust control of nonlinear systems with input unmodeled dynamics," *IEEE Trans. Autom. Control*, vol. 41, no. 6, pp. 913–920, Jun. 1996.

[14] M. Krstić and P. V. Kokotović, "Adaptive nonlinear output-feedback schemes with marino-tomei controller," *IEEE Trans. Autom. Control*, vol. 41, no. 2, pp. 274–280, Feb. 1996.

[15] J. Zhou, C. Zhang, and C. Wen, "Robust adaptive output control of uncertain nonlinear plants with unknown backlash nonlinearity," *IEEE Trans. Autom. Control*, vol. 52, no. 3, pp. 503–509, Mar. 2007.

[16] B. Yao and M. Tomizuka, "Adaptive robust control of MIMO nonlinear systems in semi-strict feedback forms," *Automatica*, vol. 37, pp. 1305–1321, 2001.

[17] W. Lin and C. Qian, "Semi-global robust stabilization of MIMO nonlinear systems by partial state and dynamic output feedback," *Automatica*, vol. 37, pp. 1093–1101, 2001.

[18] X. Liu, A. Jutan, and S. Rohani, "Almost disturbance decoupling of MIMO nonlinear systems and application to chemical processes," *Automatica*, vol. 40, pp. 465–471, 2004.

[19] G. Bartolini, A. Ferrara, L. Giacomini, and E. Usai, "Properties of a combined adaptive/second-order sliding mode control algorithm for some classes of uncertain nonlinear systems," *IEEE Trans. Autom. Control*, vol. 45, no. 7, pp. 1334–1341, Jul. 2000.

[20] A.-C. Huang and Y.-C. Chen, "Adaptive multiple-surface sliding control for non-autonomous systems with mismatched uncertainties," *Automatica*, vol. 40, pp. 1939–1945, 2004.

[21] D. Swaroop, J. K. Hedrich, P. P. Yip, and J. C. Geodes, "Dynamic surface control for a class of nonlinear systems," *IEEE Trans. Autom. Control*, vol. 45, no. 10, pp. 1893–1899, Oct. 2000.

[22] E. M. Jafarov, M. N. A. Parlakci, and Y. Istefanopulos, "A new variable structure PID-controller design for robot manipulators," *IEEE Trans. Control Syst. Technol.*, vol. 13, no. 1, pp. 122–130, Jan. 2005.

Hybrid Control for Connectivity Preserving Flocking

Michael M. Zavlanos, *Member, IEEE*,
Herbert G. Tanner, *Senior Member, IEEE*,
Ali Jadbabaie, *Member, IEEE*, and George J. Pappas, *Fellow, IEEE*

***Abstract*—In this technical note, we address the combined problem of motion and network topology control in a group of mobile agents with common objective the flocking behavior of the group. Instead of assuming network connectivity, we enforce it by means of distributed topology control that decides on both deletion and creation of communication links between agents, adapting the network to the group's spatial distribution. With this protocol ensuring network connectivity, a decentralized motion controller aligns agent velocity vectors and regulates inter-agent distances to maintain existing network links. The stability of the flocking controller is established in continuous time by means of an observability argument on a quadratic form of the graph Laplacian that exploits the time delay between link deletion and creation caused by the topology control protocol, which induces a dwell time between network switches.**

***Index Terms*—Algebraic graph theory, cooperative control, hybrid systems, multi-agent systems.**

I. INTRODUCTION

Existing results on distributed consensus and flocking algorithms critically rely on maintaining a connected communication network among the agents, either for all time (as in [1]–[3]) or over sequences

Manuscript received June 14, 2007; revised July 08, 2008, and November 25, 2008. First published November 13, 2009; current version published December 09, 2009. This work was supported by ARO MURI SWARMS under Grant W911NF-05-1-0219, by the National Science Foundation (NSF) ITR under Grant 0324977, by the NSF under Grant 0447898. Recommended by Associate Editor M. Egerstedt.

M. M. Zavlanos, A. Jadbabaie and G. J. Pappas are with the GRASP Laboratory, Department of Electrical and Systems Engineering, University of Pennsylvania, Philadelphia, PA 19104 USA (e-mail: zavlanos@grasp.upenn.edu; jadbabai@grasp.upenn.edu; pappasg@grasp.upenn.edu).

H. G. Tanner is with the Department of Mechanical Engineering, University of Delaware, Newark, DE 19716 USA (e-mail: btanner@udel.edu).

Color versions of one or more of the figures in this technical note are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TAC.2009.2033750

of bounded time intervals (as in [4]–[6]). In this technical note, we relax this assumption and propose a distributed control framework that guarantees velocity alignment, cohesion, separation, *and* connectivity of the networked multi-agent system, by construction.

Inspired by the flocking and schooling phenomena observed in nature are many recent applications in control theory and robotics. Any attempt to list related references in this technical note is bound to be partial and incomplete, hence, we rather focus on work that emphasizes on the *connectivity* aspect of networked dynamical systems. In [7], network connectivity is maintained by means of potential fields that guarantee positive definiteness of the second smallest eigenvalue of the graph Laplacian matrix, while in [8] a measure of local connectedness of a network is introduced that under certain conditions is sufficient for global connectedness. Distributed maintenance of nearest neighbor links by means of unbounded "edge tension" functions is addressed in [9], where a control hysteresis is also introduced to avoid infinite control inputs when new links are about to be inserted to the network. Similarly, in [10], a system of interconnected unicycles is steered to a common configuration by means of nonsmooth, potential-based control inputs that turn unbounded when the distance between adjacent agents approaches a certain threshold. Invariance of the level sets of an appropriate function ensures that initially established links will be preserved along the system's trajectories.

In this technical note, we address the problem of velocity synchronization in a network of n interconnected agents, while maintaining connectivity of the underlying proximity-based graph and ensuring collision avoidance among the agents. Unlike previous approaches, our proposed framework allows switching among connected network topologies that are due to both addition, as well as *deletion* of communication links between agents. As in [3] the dynamics of an agent are expressed by a double integrator

$$\dot{x}_i(t) = v_i(t) \tag{1a}$$
$$\dot{v}_i(t) = u_i(t) \tag{1b}$$

where $x_i(t), v_i(t) \in \mathbb{R}^m$ denote the position and velocity vectors of agent i at time t, respectively, and $u_i(t) \in \mathbb{R}^m$ is a *switching* control input that relies exclusively on nearest neighbor information. Unlike [3], the switching signal (set of neighbors) can be also controlled in the discrete space of graphs as in [11], and combined with the continuous agent motion, it gives rise to a closed-loop multi-agent hybrid system. Under the assumption that the communication network is *initially* connected, the overall system is shown to always stabilize to an equilibrium characterized by fixed inter-agent distances and aligned agent velocities. Our convergence results rely on recent work on the stability of switched systems [12] and essentially complete the *safety*, namely connectivity control, results employed from [11] with *liveness* guarantees. Collision avoidance is also guaranteed.

II. Problem Formulation

Consider a network of n agents in $\mathbb{R}^m$ with integrated wireless communication capabilities and denote by (i, j) a communication link between agents i and j. We assume that such links can be enabled and disabled in time due to agent mobility and/or control decisions and, as in [13], we employ proximity graphs to represent the agents' communication network. Motivation for relating the agents' ability to establish communication links to the distance between them comes from the fact that radio signal strength attenuates with distance, with the probability of a successful transmission rapidly decreasing beyond a certain threshold. To capture such radio signals, we propose a rather *qualitative* model for the communication network, where new communication

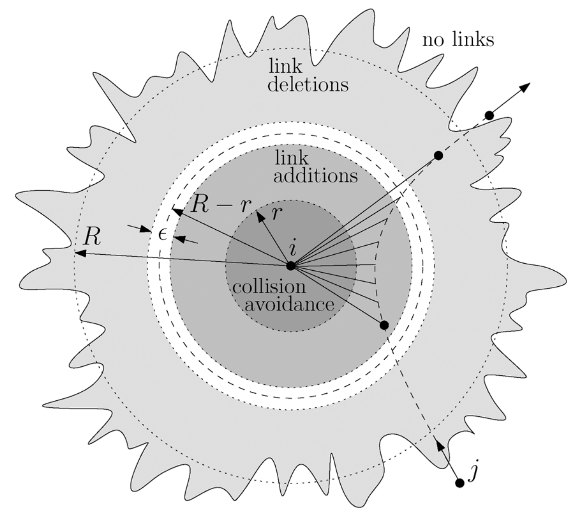

Fig. 1. Partitioning of the neighborhood of an agent according to its ability to communicate and the requirement to avoid collisions. Regions where communication links are established or lost are qualitatively associated to signal strength, ranging from strong to weak and no signal at all. The inner-most disk marks the area around agent i where collision avoidance maneuvers are initiated. Note a region of width $\epsilon > 0$ between the areas where links can be added or deleted, which introduces a dwell time $\tau > 0$ between any changes in the network topology. The dashed line shows the path of agent j, and the solid lines indicate a link between agents i and j.

links can be established only between agents whose distance becomes smaller than some threshold $R > 0$. Beyond that threshold, we assume there is considerable uncertainty over the agents' ability to successfully communicate, hence, communication links are lost. On the other hand, to avoid collisions, agents are not supposed to get too close to each other. Once their distance falls below a threshold $r > 0$, collision avoidance maneuvers must be initiated. The proposed dynamic network is illustrated in Fig. 1 and can be formally captured by a *dynamic proximity graph* $\mathcal{G}(t) = (\mathcal{V}, \mathcal{E}(t))$, where $\mathcal{V} = \{1, \ldots, n\}$ denotes the set of vertices indexed by the set of agents and $\mathcal{E}(t) \subseteq \mathcal{V} \times \mathcal{V}$ denotes the time varying set of links, such that for constants r, R, and ϵ satisfying $0 < 2r < R$ and $0 < \epsilon/2 < \min\{R - 2r, r\}$, we have:

- if $\|x_{ij}(t)\|_2 \in [0, r)$, then $(i, j) \in \mathcal{E}(t)$;
- if $(i, j) \notin \mathcal{E}(t)$ and $\|x_{ij}(t)\|_2 \in [r, R - r - \epsilon/2)$, then (i, j) is a candidate link to be *added* to $\mathcal{E}(t)$;
- if $\|x_{ij}(t)\|_2 \in [R - r - \epsilon/2, R - r + \epsilon/2)$, then (i, j) preserves its membership status in $\mathcal{E}(t)$ (no addition or deletion);
- if $(i, j) \in \mathcal{E}(t)$ and $\|x_{ij}(t)\|_2 \in [R - r + \epsilon/2, R)$, then (i, j) is a candidate link to be *deleted* from $\mathcal{E}(t)$;
- if $\|x_{ij}(t)\|_2 \in [R, \infty)$, then $(i, j) \notin \mathcal{E}(t)$;

where $x_{ij}(t) \triangleq x_i(t) - x_j(t)$. We assume *undirected* networks $\mathcal{G}(t)$, where communication links are bidirectional, i.e., $(i, j) \in \mathcal{E}(t)$ if and only if $(j, i) \in \mathcal{E}(t)$. Any vertices i and j of an undirected graph $\mathcal{G}(t)$ that are joined by a link $(i, j) \in \mathcal{E}(t)$, are called *adjacent* or *neighbors* at time t. Hence, we can define the set of neighbors of agent i at time t, by $\mathcal{N}_i(t) \triangleq \{j \in \mathcal{V} | (i, j) \in \mathcal{E}(t)\}$. If $\mathcal{G}(t)$ is such that there exists a path, i.e., a sequence of distinct vertices such that consecutive vertices are adjacent, between any two of its vertices, then we say that $\mathcal{G}(t)$ is *connected*.

Given any dynamic proximity graph $\mathcal{G}(t) = (\mathcal{V}, \mathcal{E}(t))$ consisting of n mobile agents as in (1), define the set of control laws

$$u_i(t) \triangleq - \sum_{j \in \mathcal{N}_i(t)} (v_i(t) - v_j(t)) - \sum_{j \in \mathcal{N}_i(t)} \nabla_{x_i} \varphi_{ij}(t) \tag{2}$$

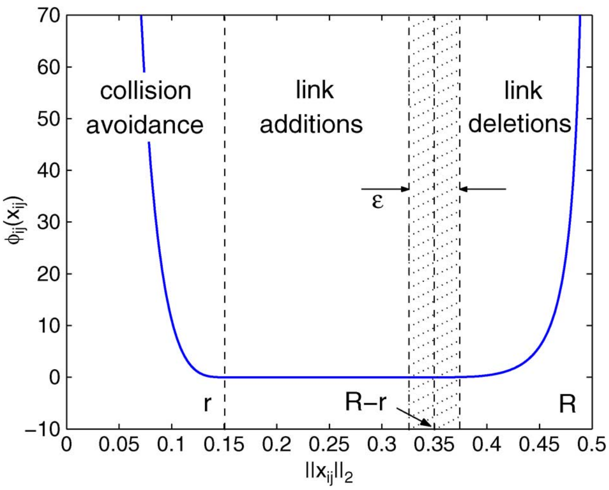

Fig. 2. Artificial potential function $\varphi_{ij}(x_{ij})$. The function is symmetric with respect to x_i and x_j, and when bounded, it guarantees both collision avoidance for $\|x_{ij}\|_2 \to 0$ and edge preservation for $\|x_{ij}\|_2 \to R$. Here, the function is plotted for $r = 0.15$, $R = 0.5$, and $\epsilon = 0.05$.

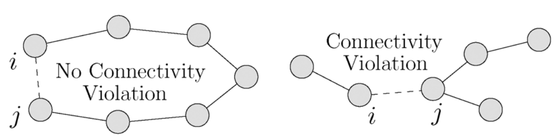

Fig. 3. Control challenges requiring knowledge of the network structure. Without such knowledge, deletion of a link (i, j) can either violate connectivity (right) or not (left).

where $\varphi_{ij}(t)$ is the artificial potential function (Fig. 2)

$$\varphi_{ij}(x_{ij}) \triangleq \begin{cases} \|x_{ij}\|_2^{-2} + P_1(x_{ij}), & \|x_{ij}\|_2 \in (0, r] \\ 0, & \|x_{ij}\|_2 \in (r, R-r) \\ \frac{1}{R^2 - \|x_{ij}\|_2^2} + P_2(x_{ij}), & \|x_{ij}\|_2 \in [R-r, R) \end{cases} \quad (3)$$

where $P_k(x_{ij}) \triangleq a_k\|x_{ij}\|_2^2 + b_k\|x_{ij}\|_2 + c_k$ with $k = 1, 2$ for appropriate constants a_k, b_k and c_k so that the derivatives of φ_{ij} up to second order are continuous in $(0, R)$, i.e., for $\|x_{ij}\|_2 \in [r, R-r]$ they satisfy $\varphi_{ij}(x_{ij}) = \partial\varphi_{ij}/\partial\|x_{ij}\|_2 = \partial^2\varphi_{ij}/\partial\|x_{ij}\|_2^2 = 0$. Hence, the problem addressed in this technical note can be stated as follows.

Problem 1 (Connectivity Preserving Flocking): Given an initially connected network $\mathcal{G}(t_0)$ consisting of n agents described by (1), (2), determine local controllers $\mathcal{N}_i(t)$ so that the overall dynamic network $\mathcal{G}(t)$ is connected for all time, all agent velocities are aligned and collisions among the agents are avoided.

To obtain local controllers $\mathcal{N}_i(t)$ that guarantee invariance of the whole mobile network $\mathcal{G}(t)$ with respect to connectivity, we choose an equivalent algebraic representation of a dynamic graph $\mathcal{G}(t) = (\mathcal{V}, \mathcal{E}(t))$ using the *laplacian matrix*

$$L(t) \triangleq \Delta(t) - A(t) \quad (4)$$

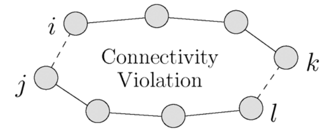

Fig. 4. Control challenges due to multiple link deletions. In the absence of an agreement protocol, simultaneous deletion of links (i, j) and (k, l) violates connectivity.

TABLE I

$a_{jk}^{[i]}(t)$	$v_{jk}^{[i]}(t)$	$a_{jk}^{[i]}(t+1)$
1	1	0
1	0	1
0	1	1
0	0	0

where $A(t) = (a_{ij}(t))$ corresponds to the *adjacency matrix* of the graph $\mathcal{G}(t)$, such that $a_{ij}(t) = 1$ if $(i, j) \in \mathcal{E}(t)$ and $a_{ij}(t) = 0$ otherwise, and $\Delta(t) = \mathrm{diag}(\sum_{j=1}^n a_{ij}(t))$ denotes the *valency matrix*.[1] The spectral properties of the Laplacian matrix are closely related to graph connectivity. In particular, if $\lambda_1(t) \le \lambda_2(t) \le \ldots \le \lambda_n(t)$ are the ordered eigenvalues of the Laplacian matrix $L(t)$, then $\lambda_1(t) = 0$ for all t, with corresponding eigenvector $\mathbf{1}$, and $\lambda_2(t) > 0$ if and only if $\mathcal{G}(t)$ is connected [14]. The above algebraic representation of a graph $\mathcal{G}(t)$, along with auction-based link deletions, leads to the desired distributed connectivity controllers $\mathcal{N}_i(t)$ for Problem 1.

III. Distributed Connectivity Control

Consider a dynamic graph $\mathcal{G}(t) = (\mathcal{V}, \mathcal{E}(t))$ defined by the time varying set of edges $\mathcal{E}(t)$.[2] The goal in this section is to design *local* controllers that allow every agent to add or delete communication links with neighbors without violating connectivity of $\mathcal{G}$. As connectivity is a global graph property, it is necessary that every agent has sufficient knowledge of the network structure in order to safely delete a link with a neighbor (Fig. 3).[3] Such knowledge can be obtained through local *estimates* of the network topology (Section III-A), which, along with a *tie breaking* mechanism obtained by means of *gossip algorithms* and distributed *market-based* control (Section III-B), ensure connectivity even when combinations of multiple deletion requests could possibly violate it (Fig. 4).

A. Local Estimates of the Network Topology

Let $\mathcal{G}_i = (\mathcal{V}, \mathcal{E}_i)$ denote a local *estimate* of the global network $\mathcal{G}$ that agent i can obtain using information from its nearest neighbors $\mathcal{N}_i$. Let also $A_i = (a_{jk}^{[i]})$ denote the adjacency matrix associated with the graph $\mathcal{G}_i$ and define its dynamics by[4]

$$A_i(t+1) := \neg(A_i(t) \leftrightarrow V_i(t)) \quad (5)$$

where the control input $V_i = (v_{jk}^{[i]}) \in \{0, 1\}^{n\times n}$ is such that $v_{jk}^{[i]} = 1$ if a control action is taken to add or delete link (j, k) (Table I). It can

[1]Since we do not allow self-loops, we define $a_{ii}(t) = 0$ for all i.

[2]To simplify notation, we hereafter often drop dependence on time t.

[3]Addition of links can only increase connectivity and does not introduce any significant challenge in controlling the topology of $\mathcal{G}$.

[4]The symbols $\neg, \wedge, \vee, \rightarrow$ and $\leftrightarrow$ stand for the boolean operators Not, And, Or, If, Then and If And Only If, respectively (in the case of matrices, they are applied elementwise on their entries). The discrete time semantics in (5) are associated with discrete communication time instances between adjacent agents (Section V).

be shown that the control input V_i can be decomposed into two disjoint components V_i^a and V_i^d regulating link *additions* and *deletions*, respectively, as [11]

$$V_i \triangleq \left(\left(\underbrace{(\neg A_i \wedge (\vee_{j\in\mathcal{N}_i} A_j))}_{\text{new links provided by current neighbors}} \vee \underbrace{(\neg A_i \wedge E_i)}_{\text{links with new neighbors}}\right) \wedge V_i^a\right) \vee \left(A_i \wedge V_i^d\right)$$

where $E_i = \vee_{j\neq i}(e_i e_j^T \vee e_j e_i^T)$, and e_i is a column vector with all entries 0 except for the ith entry that is 1. This results in the local network dynamics (5) being essentially a *consensus* (with inputs) on the adjacency matrix estimates A_i. In particular, for a fixed network topology (no inputs), the dynamics (5) reduce to

$$A_i := \vee_{j\in\mathcal{N}_i}(A_i \vee A_j),$$

which for local initialization of the estimates A_i with nearest neighbor links, provide every agent with a *rough* picture of the overall network.

B. Controlling Addition and Deletion of Links

Regarding the controller $V_i^a = (v_{jk}^{[i]a})$ that regulates link additions, we require that it captures all communication links that are known to agent i's neighbors $\mathcal{N}_i$ as well as all new links that agent i can create with agents $j \notin \mathcal{N}_i$, i.e.,

$$v_{jk}^{[i]a} \triangleq \underbrace{((j\neq i)\wedge(k\neq i))}_{\text{add all existing links provided by neighbors}} \vee \underbrace{(\|x_{jk}\|_2 \in [r, R-r-\epsilon/2))}_{\text{maintain current neighbors and add new neighbors}}. \quad (6)$$

Unlike link additions, deletion of nearest neighbor links is a challenging task. Although knowledge of the estimate $\mathcal{G}_i$ allows every agent i to determine adjacent links that if deleted individually, preserve network connectivity (Fig. 3), it is not sufficient for dealing with simultaneous link deletions by multiple nonadjacent agents that may disconnect $\mathcal{G}$ (Fig. 4). For this, we require that at most one link be deleted from $\mathcal{G}$ at a time and employ an *auction-based* framework to achieve agreement of all agents regarding the link that is to be deleted (Algorithm 1).

Algorithm 1 Link deletion controller for agent i

Require: $\mathcal{S}_i := \{j \in \mathcal{N}_i^d | \lambda_2(\mathcal{E}_i \setminus \{(i,j)\}) > 0\}$;

Require $r_i \triangleq [i\ j\ b]^T$ and $T_i := e_i$;

1: **if** $\wedge_{j=1}^n T_{ij} = 0$ **then**

2: $r_i := r_j$, where $j = \max\{\arg\max_{k\in\mathcal{N}_i}\{r_{i3}, r_{k3}\}\}$;

3: $T_i := T_i \vee (\vee_{j\in\mathcal{N}_i} T_j)$;

4: **else if** $\wedge_{j=1}^n T_{ij} = 1$ and $r_{i3} > 0$ **then**

5: $v_{jk}^{[i]d} := ((r_{i1}, r_{i2}) = (j,k))$;

6: **else if** $\wedge_{j=1}^n T_{ij} = 1$ and $r_{i3} = 0$ **then**

7: $v_{jk}^{[i]d} := 0$ for all j, k;

8: **end if**

Initialization of an auction requires a set $\mathcal{S}_i \in 2^{\mathbb{N}}$ of *safe* neighbors $j \in \mathcal{N}_i^d \triangleq \{j \in \mathcal{N}_i | \|x_{ij}\|_2 \in [R-r+\epsilon/2, R)\}$ that if agent i deletes a link (i,j) with, then $\mathcal{G}_i$ remains connected. For any $j \in \mathcal{S}_i$, agent i initializes a request $r_i \triangleq [ijb]^T \in \mathbb{R}^3$ containing the link (i,j) to be deleted and a bid $b \geq 0$, such that $b > 0$ if $\mathcal{S}_i \neq \emptyset$ and $b = 0$ otherwise, indicating how "important" this request is.[5] This request is propagated in the network, along with a vector of tokens $T_i \in \{0,1\}^n$, initialized as $T_i := e_i$, indicating that agent i has placed its bid. During every auction, every agent i communicates with its neighbors and updates its vector of tokens T_i (line 3, Algorithm 1), as well as its request r_i with the request r_j corresponding to the agent j that has placed the highest bid r_{j3}, i.e., $j \in \arg\max_{k\in\mathcal{N}_i}\{r_{i3}, r_{k3}\}$. In case of ties on the bids, i.e., if $\arg\max_{k\in\mathcal{N}_i}\{r_{i3}, r_{k3}\}$ contains more than one agents, then the agent j with the maximum label is selected (line 2, Algorithm 1). Note that line 2 of Algorithm 1 is essentially a maximum consensus update on the bids r_{i3} and will converge to a common outcome for all agents when all bids have been compared to each other, which is captured by the condition $\wedge_{j=1}^n T_{ij} = 1$ (lines 4 and 6, Algorithm 1). If at least one agent has placed a positive bid, i.e., if $r_{i3} > 0$ (line 4, Algorithm 1), then controller $V_i^d = (v_{jk}^{[i]d})$ deletes the associated link (r_{i1}, r_{i2}) from $\mathcal{G}_i$ (line 5, Algorithm 1). Otherwise, no link is deleted (line 7, Algorithm 1).

Remark 3.1 (Computational Complexity): Computation of the spectrum of a matrix has worst case complexity $O(n^3)$, where n is the size of the matrix [15]. This complexity can be reduced to $O(n)$ for sparse symmetric matrices [16], as is typically the Laplacian matrix $L(t)$ of large networks. Consequently, our approach is scalable to relatively large networks.

C. Agent Synchronization

Communication time delays, packet losses, and the asymmetric network structure, may result in auctions starting asynchronously, outdated information being used for future decisions, and consequently, agents reaching different decisions for the same auction. In the absence of a common global clock, the desired synchronization is ideally *event triggered*, where by a triggering event we understand the time instant that a message $\mathrm{Msg}[i] \triangleq \{A_i, r_i, T_i\}$ has been received by any of agent i's neighbors $j \in \mathcal{N}_i$. We achieve such a synchronization by labeling every auction in the set {1,2,3} and requiring that all information exchange takes place among neighbors that are in equally labeled auctions. Essentially, "fast" agents wait for their "slower" peers and, hence, all agents are always synchronized in the sequence {1,2,3,1,2,3,...} (Fig. 5).

[5]Letting $b \geq 0$ be a function of the distance $\|x_{ij}\|_2$, $j \in \mathcal{S}_i$ or the size of the neighbor set $|\mathcal{N}_i|$ can be associated with signal strength or power constraint properties of the overall network.

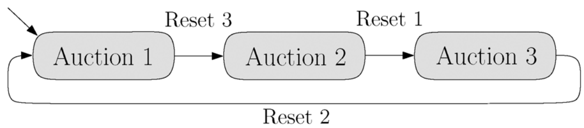

Fig. 5. Agent synchronization. Assume agent i is in auction 1. Necessary for agent i to transition to auction 2 is that all other agents are also in auction 1, since otherwise agent i will be missing tokens from the agents that are not in auction 1 yet (currently in auction 3) and Algorithm 1 will not be able to converge. Once agent i transitions to auction 2, it initializes all variables for that auction with the latest values from auction 1, while it maintains the variables of auction 1 for agents that are still in auction 1 and it clears all variables of auction 3 since, no agent is in this auction any more [11].

D. Correctness

Correctness of the proposed distributed coordination framework is obtained by construction and is discussed in detail in the previous subsections. Those ideas are summarized in the following result.[6]

Theorem 3.2 (Correctness): Assume an initially connected network $\mathcal{G}$ of agents that are initialized with nearest neighbor information. Then, the proposed scheme for distributed addition and deletion of communication links guarantees that $\mathcal{G}$ remains always connected.

Proof: The proof relies on the following observations:[7]

(a) All network estimates $\mathcal{G}_i$ are spanning subgraphs of the overall network $\mathcal{G}$, i.e., $\mathcal{E}_i \subseteq \mathcal{E}$, which implies that connectivity can be checked locally for $\mathcal{G}_i$ and the results can be extended to $\mathcal{G}$.

(b) Agreement of all agents on the link that is to be deleted, is guaranteed by convergence of the max-consensus in Algorithm 1.

(c) Points (a) and (b) above ensure that all agents agree on the link that is to be deleted and that this deletion does not violate connectivity. The synchronization scheme described in Section III-C ensures that no outdated information is used in any auction.

Consequently, links can be deleted continuously one-by-one, without destroying connectivity of the network. ■

IV. Integration With Agent Mobility: Velocity Alignment and Collision Avoidance

With the topology control component of Section III adding and deleting edges at will, the agent control law (2) experiences discontinuities, and induces a switching nonlinear closed-loop dynamical system (1), (2). Let t_p for $p = 1, 2, \ldots$ denote the *switching times* when the topology of $\mathcal{G}(t)$ changes, and define a switching signal $\mathcal{G}(t) : [0, \infty) \to \mathcal{G}_\mathcal{C}$, where $\mathcal{G}_\mathcal{C}$ denotes the set of all connected graphs on n vertices.[8] As discussed in Section III, the topology controller guarantees that the sequence of proximity graphs consists of connected graphs, while communication time delays (Section V) and the idle ϵ-annulus between the regions where links can be added and deleted (Figs. 1 and 2), introduce a dwell time $\tau > 0$ between transitions in the network topology $\mathcal{G}(t)$. Hence, we can now state our main result.

Theorem 4.1 (Connectivity Preserving Flocking): For the closed-loop system (1), (2), assume that $\mathcal{G}(t_0) \in \mathcal{G}_\mathcal{C}$ and $t_p - t_{p-1} > \tau > 0$ for all switching times $t_p > 0$. Then, $\mathcal{G}(t) \in \mathcal{G}_\mathcal{C}$ for all $t > 0$, and $\|x_{ij}(t)\|_2 > 0$ for all $i, j \in \mathcal{V}$ and all $t > 0$. Moreover, $v_i \to v_j$ as $t \to \infty$, for all $i, j \in \mathcal{V}$.

[6]Implementation on a real robotic platform showed that our approach is also feasible as well as highly successful in enabling a network of robots to adapt to disturbances [95% packet loss using multicast *User Datagram Protocol* (UDP)] while maintaining network connectivity [17].

[7]Due to space limitations, a more rigorous proof is omitted. More details can be found in [11].

[8]Note that $\mathcal{G}(t)$ is also a map from the real time-line to the set of graphs.

Proof: Let $t_{p_1}, t_{p_2}, \ldots$ denote an infinite subsequence of switching times such that the switching signal $\mathcal{G}(t)$ in each of the intervals $[t_{p_q}, t_{p_{q+1}})$ for $q = 1, 2, \ldots$ is the same. Denote the union of these intervals by $\mathcal{Q}$ and for all time $t \in \mathcal{Q}$, let $\hat{\mathbf{x}} \triangleq [\ldots x_{ij}^T \ldots]^T \in \mathbb{R}^{mn(n-1)/2}$, $\mathbf{v} \triangleq [\ldots v_i^T \ldots]^T \in \mathbb{R}^{mn}$ and $\mathbf{u} \triangleq [\ldots u_i^T \ldots]^T \in \mathbb{R}^{mn}$ denote the stack vectors of the agent relative positions $x_{ij} \in \mathbb{R}^m$, velocity vectors $v_i \in \mathbb{R}^m$ and control signals vectors $u_i \in \mathbb{R}^m$, respectively. Consider the dynamical system

$$\dot{\hat{\mathbf{x}}} = (B_K \otimes I_m)\mathbf{v} \tag{7a}$$

$$\dot{\mathbf{v}} = \mathbf{u} \tag{7b}$$

where B_K is the incidence matrix of the complete proximity graph [14], and define the function $\varphi_\mathcal{G} : \mathbb{R}^{mn(n-1)/2} \times \mathbb{R}^{mn} \to \mathbb{R}_+$ with[9]

$$\varphi_\mathcal{G} = \frac{1}{2}\left(\|\mathbf{v}\|_2^2 + \sum_{i=1}^{n}\sum_{j\in\mathcal{N}_i}\varphi_{ij}\right). \tag{8}$$

For any $c > 0$, let $\Omega_\mathcal{G} = \{(\hat{\mathbf{x}}, \mathbf{v}) \in \mathbb{R}^{mn(n-1)/2} \times \mathbb{R}^{mn} | \varphi_\mathcal{G} \le c\}$ denote the level sets of $\varphi_\mathcal{G}$ and observe that (cf. [3])

$$\frac{1}{2}\sum_{i=1}^{n}\sum_{j\in\mathcal{N}_i}\dot{\varphi}_{ij} = \sum_{i=1}^{n}\sum_{j\in\mathcal{N}_i}\dot{x}_i^T\nabla_{x_i}\varphi_{ij}. \tag{9}$$

Equation (9) and the Kronecker product notation ($\otimes$) simplifies the expression for $\dot{\varphi}_\mathcal{G}$ to

$$\dot{\varphi}_\mathcal{G} = -\mathbf{v}^T(L_\mathcal{G} \otimes I_m)\mathbf{v}, \tag{10}$$

which is always nonpositive, since the Laplacian $L_\mathcal{G}$ is always positive semidefinite. Hence, for any signal $\mathcal{G}$, the level sets $\Omega_\mathcal{G}$ are positively invariant, implying that for any $(i, j) \in \mathcal{E}$, φ_{ij} remains bounded. On the other hand, if for some $(i, j) \in \mathcal{E}$, $\|x_{ij}\|_2 \to R$, then $\varphi_{ij}(x_{ij}) \to \infty$. Thus, by continuity of $\varphi_\mathcal{G}$ in its domain, it follows that $\|x_{ij}\|_2 < R$, for all $(i, j) \in \mathcal{E}$ and $t \in [t_{p_q}, t_{p_q+1})$. In other words, all links in $\mathcal{G}$ are maintained between switching times, and since $\mathcal{G}(t_{p_q}) \in \mathcal{G}_\mathcal{C}$ for $q = 1, 2, \ldots$, we have that $\mathcal{G}(t) \in \mathcal{G}_\mathcal{C}$ for all $t \in [t_{p_q}, t_{p_q+1})$, hence, for all $t \in \mathcal{Q}$. A similar argument for the case where $\|x_{ij}\|_2 \to 0$ can be used to establish collision avoidance.

Since the addition of an edge between agents i and j can occur only in the region where $\|x_{ij}\|_2 \in [r, R - r - \epsilon/2)$, such an event temporarily has no effect on the value of $\sum_{i=1}^{n}\sum_{j\in\mathcal{N}_i}\varphi_{ij}$ (Fig. 2). In conjunction with (10) we conclude that $\varphi_{\mathcal{G}(t_{p_q})} \ge \varphi_{\mathcal{G}(t_{p_q+1})}$, so if $\varphi_{\mathcal{G}(t_{p_1})}$ is bounded, so is $\varphi_{\mathcal{G}(t_{p_q})}$ for all $q = 1, 2, \ldots$. Moreover, the level sets of $\Omega_\mathcal{G}$ are closed by continuity of $\varphi_\mathcal{G}$ in its domain. Note now that

$$\begin{aligned}\Omega_\mathcal{G} &\subseteq \left\{\mathbf{v} | \|\mathbf{v}\|_2^2 \le c\right\} \cap \left(\cap_{(i,j)\in\mathcal{E}}\{x_{ij} | \varphi_{ij} \le c\}\right) \\ &= \left\{\mathbf{v} | \|\mathbf{v}\|_2^2 \le c\right\} \cap \left(\cap_{(i,j)\in\mathcal{E}}\varphi_{ij}^{-1}([0, c])\right) \triangleq \Omega.\end{aligned} \tag{11}$$

The velocity set $\{\mathbf{v} | \|\mathbf{v}\|_2^2 \le c\}$ is closed and bounded and hence, compact. Moreover, for all $(i, j) \in \mathcal{E}$ the sets $\varphi_{ij}^{-1}([0, c])$ are closed by continuity of φ_{ij} in the interval $(0, R)$. They are also bounded; to see this, suppose there exist indices i and j for which $\varphi_{ij}^{-1}([0, c])$ is unbounded. Then, for any choice of $N \in (0, R)$, there exists an $x_{ij} \in \varphi_{ij}^{-1}([0, c])$ such that $\|x_{ij}\|_2 > N$. Allowing $N \to R$, and given that $\lim_{\|x_{ij}\|_2 \to R}\varphi_{ij} = \infty$, it follows that for any $M > 0$, there is a

[9]We denote by $\mathbb{R}_+$ the set $[0, \infty)$.

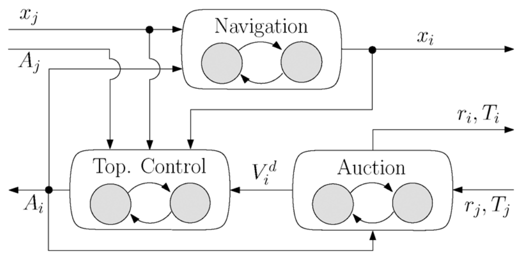

Fig. 6. Hybrid automaton $\mathbb{T}_i \times \mathbb{A}_i \times \mathbb{N}_i$ of a mobile agent i that consists of the composition of a topology control $\mathbb{T}_i$, an auction $\mathbb{A}_i$ and a navigation automaton $\mathbb{N}_i$.

$N > 0$ such that $\varphi_{ij} > M$. If we pick $M > c$ we reach a contradiction, since by definition $x_{ij} \in \varphi_{ij}^{-1}([0,c]) = \{x_{ij} | \varphi_{ij}(x_{ij}) \leq c\}$. Thus, all sets $\varphi_{ij}^{-1}([0,c])$ are bounded and hence, compact. Therefore, the set Ω is compact as an intersection of finite compact sets. It follows that $\Omega_{\mathcal{G}}$ is also compact, as a closed subset of a compact set.

So far we have shown that the level sets $\Omega_{\mathcal{G}}$ of $\varphi_{\mathcal{G}}$ are both positively invariant and compact. The invariance of $\Omega_{\mathcal{G}}$ implies that no collisions between agents occur. We now use these results to show that all agent velocities become asymptotically the same. Note first that compactness and positive invariance of $\Omega_{\mathcal{G}}$ implies that $(\hat{\mathbf{x}}, \mathbf{v}) \in \mathbb{R}^{mn(n-1)/2} \times \mathbb{R}^{mn}$ remains bounded in every bounded time interval $[t_{p_q}, t_{p_{q+1}})$ and, hence, in the union $\mathcal{Q}$. Moreover, since $\varphi_{\mathcal{G}} \in C^2$ in its domain, the right-hand-side of (7) is locally Lipschitz, which implies that $(\dot{\hat{\mathbf{x}}}, \dot{\mathbf{v}})$ is also bounded in every bounded time interval $[t_{p_q}, t_{p_{q+1}})$ and, hence, in the union $\mathcal{Q}$. This suggests that the quantity $\mathbf{v}^T(L_{\mathcal{G}} \otimes I_m)\mathbf{v}$ is uniformly continuous in $\mathcal{Q}$ [12]. Define the auxiliary function

$$y_{\mathcal{Q}}(t) \triangleq \begin{cases} \mathbf{v}^T(L_{\mathcal{G}} \otimes I_m)\mathbf{v}, & t \in \mathcal{Q} \\ 0, & \text{otherwise.} \end{cases} \tag{12}$$

Since $\varphi_{\mathcal{G}(t_p)} \leq \varphi_{\mathcal{G}(t_{p-1})}$ for any consecutive signals $\mathcal{G}(t_{p-1}), \mathcal{G}(t_p) \in \mathcal{G}_{\mathcal{C}}$, characterized by simultaneous addition and deletion of links, and all switching times t_p, we have that

$$\int_0^t |y_{\mathcal{Q}}(s)|\, ds = \int_0^t y_{\mathcal{Q}}(s) ds \leq \varphi_{\mathcal{G}(t_{p_1})} - \varphi_{\mathcal{G}(t)} \leq \varphi_{\mathcal{G}(t_{p_1})}$$

which suggests that $y_{\mathcal{Q}} \in \mathcal{L}_1$. We now proceed to showing that $y_{\mathcal{Q}}(t) \to 0$ as $t \to \infty$. Our argument is along the lines of [12]: suppose that this is not true. Then, there exists an $\varepsilon > 0$ and an infinite sequence of times $s_1, s_2, \ldots$ such that the values $y_{\mathcal{Q}}(s_1), y_{\mathcal{Q}}(s_2), \ldots$ are bounded away from zero by at least ε. It follows from (12) that the times $s_1, s_2, \ldots$ necessarily belong to $\mathcal{Q}$. Since $y_{\mathcal{Q}}$ is uniformly continuous, we can find a $\delta > 0$ such that each s_i is contained in some interval of length δ, on which $y_{\mathcal{Q}}(t) \geq \varepsilon/2$. (Recall that the length of each such interval in $\mathcal{Q}$ is lower bounded by $\tau > 0$.) This contradicts the fact that $y_{\mathcal{Q}} \in \mathcal{L}_1$. Hence, $y_{\mathcal{Q}}(t) \to 0$ as $t \to \infty$, which suggests that $\mathbf{v} \to \mathbf{1} \otimes \zeta^T$ for some $\zeta \in \mathbb{R}^m$. This means that the corresponding components of agent velocities converge asymptotically to common values. ■

V. The Closed-Loop Hybrid Agent

Integration of the discrete topology controllers described in Section III with the continuous motion controllers (1), (2) give rise to a hybrid model $\mathbb{T}_i \times \mathbb{A}_i \times \mathbb{N}_i$ for every agent i, consisting of a *topology control* $\mathbb{T}_i$, an *auction* $\mathbb{A}_i$ and a *navigation* automaton $\mathbb{N}_i$, respectively [11] (Fig. 6). The topology controller of agent i updates its network estimate A_i with addition and deletion of links (Section III-A). For this, it requires the control input V_i^d that regulates link deletions, as well as the network estimates A_j of agent i's neighbors in order to compute the control input V_i^a that regulates link additions (Section III-B). The control input V_i^d is provided by the auction controller and relies on the max-requests r_j and tokens T_j of agent i's neighbors (Algorithm 1). To capture the continuous agent motion, the navigation controller $\mathbb{N}_i$ coordinates with the associated topology and auction controllers to obtain the agent's set of neighbors $\mathcal{N}_i$, which it uses, along with their positions x_j for $j \in \mathcal{N}_i$, to update its own position x_i ((1) and (2)). The updated agent positions are then provided to the topology controller that further updates agent i's network estimate A_i and the resulting set of neighbors $\mathcal{N}_i$. Note that in the proposed hybrid system, all variables are considered *shared*, however, the only variables that are practically needed are the ones provided by every agent's neighbors, which guarantees the local nature of the proposed control framework.

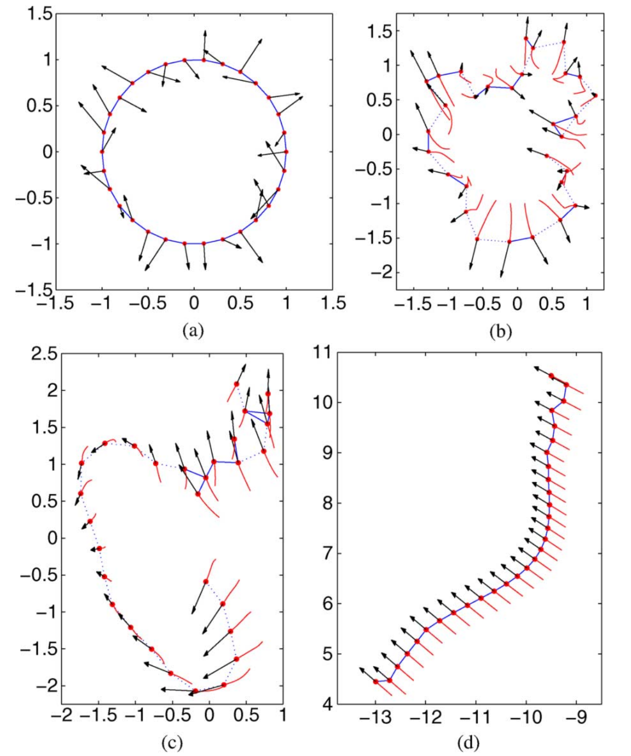

Fig. 7. Decentralized flocking of 30 agents with topology control. Dotted lines indicate communication links that are candidates for deletion. (a) Initial configuration; (b) $t_1 > 0$; (c) $t_2 > t_1$; (d) steady state.

VI. Simulation Results

We apply the proposed hybrid controller in coordination problems where connectivity of the network can not be trivially maintained. We consider $n = 30$ agents in $\mathbb{R}^2$, symmetrically distributed on the perimeter of a unit circle, with initial velocities chosen randomly within the unit square and parameters $r = 0.15$ and $R = 0.5$. The agents are denoted with dots, while the links between them are indicated by either solid or dashed lines, depending on whether the corresponding inter-agent distances are in the $[0, R-r)$ or $[R-r, R)$ region, respectively. Solid curves attached to every agent indicate the recently traveled paths, while arrows correspond to the agents' velocities (Fig. 7). Fig. 8(a)–8(c) show the evolutions with time (log-scale) of the Fiedler eigenvalue $\lambda_2(t)$, the auxiliary function $y_{\mathcal{Q}}(t)$, and the minimum distance $\min_{ij}\{\|x_{ij}(t)\|_2\}$ between agents. One can clearly

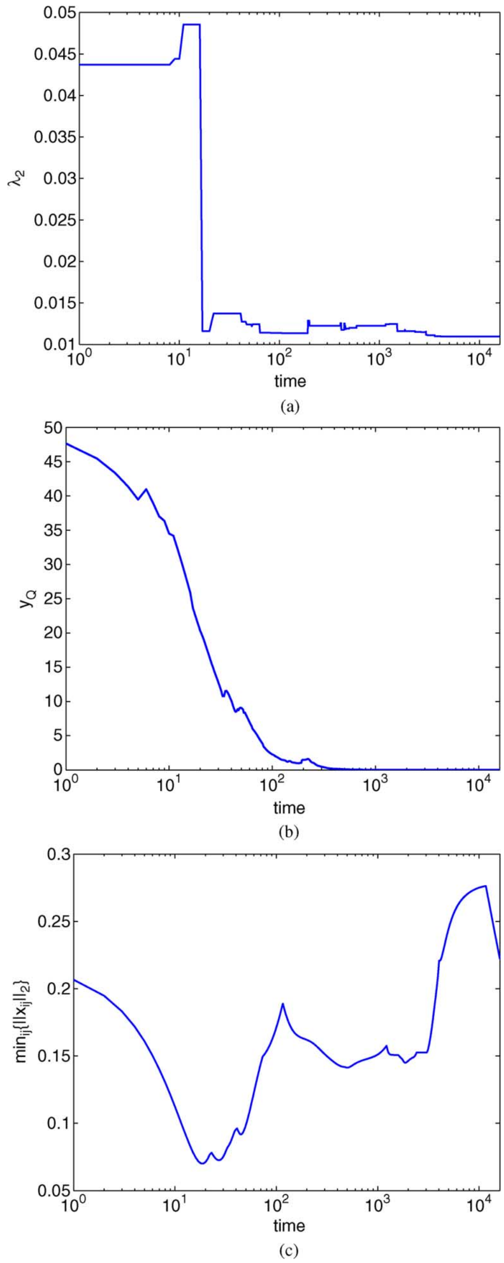

Fig. 8. Decentralized flocking with topology control. (a) Shows how the second smallest eigenvalue of the network's Laplacian varies in $\mathbb{R}_+$. The topology control protocol allows the network to change without being shattered. (b) Verifies the prediction of the proof of Theorem 4.1 that the auxiliary function y_Q vanishes with time. (c) Demonstrates the collision avoidance capabilities of (2). (a) Fiedler eigenvalue. (b) Auxiliary function. (c) Minimum inter-agent distance.

see that the network always remains connected, all agent velocities are asymptotically aligned and collisions among agents are always avoided, as desired.

VII. Conclusion

In this technical note, we offer a solution to the problem of producing flocking behavior in a group of mobile agents, without assuming network connectivity or forcing all initial links to be maintained over all time. By means of a distributed topology control protocol, all possible network links are subject to creation or deletion, depending on the spatial distribution of the agents at any given time instant. The discrete network protocol ensures connectivity of the dynamic network, as well as a hysteresis between topology changes, properties which a continuous decentralized motion controller exploits to guarantee velocity synchronization and collision avoidance. The two controllers are combined into a hybrid architecture, where topology control facilitates motion coordination, and motion control preserves the topology dictated by the discrete network controller.

References

[1] R. Olfati-Saber and R. M. Murray, "Consensus problems in networks of agents with switching topology and time-delays," *IEEE Trans. Autom. Control*, vol. 49, no. 9, pp. 1520–1533, Sep. 2004.

[2] R. Olfati-Saber, "Flocking for multi-agent dynamic systems: Algorithms and theory," *IEEE Trans. Autom. Control*, vol. 51, no. 3, pp. 401–420, Mar. 2006.

[3] H. G. Tanner, A. Jadbabaie, and G. J. Pappas, "Flocking in fixed and switching networks," *IEEE Trans. Autom. Control*, vol. 52, no. 5, pp. 863–867, May 2007.

[4] A. Jadbabaie, J. Lin, and A. S. Morse, "Coordination of groups of mobile autonomous agents using nearest neighbor rules," *IEEE Trans. Autom. Control*, vol. 48, no. 6, pp. 988–1001, Jul. 2002.

[5] W. Ren and R. Beard, "Consensus seeking in multiagent systems under dynamically changing interaction topologies," *IEEE Trans. Autom. Control*, vol. 50, no. 5, pp. 655–661, May 2005.

[6] J. Lin, A. Morse, and B. Anderson, "The multi-agent rendezvous problem," in *Proc. 42nd IEEE Conf. Decision and Control*, Maui, HI, Dec. 2003, pp. 1508–1513.

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

[8] D. P. Spanos and R. M. Murray, "Robust connectivity of networked vehicles," in *Proc. 43rd IEEE Conf. Decision and Control*, Paradise Island, Bahamas, Dec. 2004, pp. 289–2898.

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

[10] D. V. Dimarogonas and K. J. Kyriakopoulos, "Connectivity preserving state agreement for multiple unicycles," in *Proc. American Control Conf.*, New York, Jul. 2007, pp. 1179–1184.

[11] M. M. Zavlanos and G. J. Pappas, "Distributed connectivity control of mobile networks," *IEEE Trans. Robotics*, vol. 24, no. 6, pp. 1416–1428, December 2008.

[12] J. P. Hespanha, D. Liberzon, D. Angeli, and E. D. Sontag, "Nonlinear norm-observability notions and stability of switched systems," *IEEE Trans. Autom. Control*, vol. 50, no. 2, pp. 154–168, Feb. 2005.

[13] J. Cortes, S. Martinez, and F. Bullo, "Robust rendezvous for mobile autonomous agents via proximity graphs in arbitrary dimensions," *IEEE Trans. Autom. Control*, vol. 51, no. 8, pp. 1289–1298, Aug. 2006.

[14] C. Godsil and G. Royle, *Algebraic Graph Theory*. New York: Springer-Verlag, 2001.

[15] G. H. Golub and C. F. V. Loan, *Matrix Computations*. Baltimore, MD: Johns Hopkins Univ. Press, 1996.

[16] S.-T. Yau and Y. Y. Lu, "A new approach to sparse matrix eigenvalues," in *Proc. 1st IEEE Conf. Regional Aerospace Control Systems*, Westlake Village, CA, May 1993, pp. 132–137.

[17] N. Michael, M. M. Zavlanos, V. Kumar, and G. J. Pappas, "Maintaining connectivity in mobile robot networks," in *Proc. 11th Int. Symp. Experimental Robotics*, Athens, Greece, Jul. 2008.