

Optimal Decentralization of Multi-Agent Motions

Philip Twu and Magnus Egerstedt
ptwu@gatech.edu, magnus@ece.gatech.edu
Electrical and Computer Engineering
Georgia Institute of Technology
Atlanta, GA 30332, USA

Abstract—This paper addresses how to optimally decentralize the execution of a multi-agent mission defined at the trajectory-level, where the information flow among agents in the system are limited by a predefined network topology. Each agent's decentralized controllers are constrained to be parameterized functions of the relative distances and angles between itself and its neighbors. Starting with a discussion on what it means for a controller to be considered decentralized, the problem is posed as an optimal control problem for switched autonomous systems. We derive optimality conditions for the parameters defining each mode for each agent, which is combined with optimality conditions for when to switch between consecutive modes. Simulations are used to showcase the operation of the proposed optimal decentralization algorithm on a complex example.

I. INTRODUCTION

The issue of decentralized control has received significant attention during the last decade, with two distinctly different approaches emerging. The first approach is concerned with the question of how to show that a given decentralized controller satisfies certain desired global properties. This can be thought of as a bottom-up approach, and examples of this include rendezvous and consensus controllers (e.g. [1], [2], [3], [4], [5]), formation control (e.g. [6], [7]), and swarming inspired controllers (e.g. [8], [9], [10], [11]). The top-down approach involves specifying a global performance metric, and then investigating when the resulting optimal controller is in fact decentralized. Examples of this view include [12], [13], [14], [15], [16].

In this paper we try to bridge these two approaches by assuming that we are given a desired motion, expressed in terms of trajectories of the individual agents. Coupled with these are constraints on what constitutes a decentralized controller. We parameterize these constraints and solve the optimization problem associated with finding the parameters that make the decentralized trajectories track the desired trajectories the best.

The outline of this paper is as follows: In Section 2, some preliminary notation is established and the properties of a decentralized controller are discussed. In section 3, the decentralization problem is formulated as an optimal control problem. Optimality conditions are derived for parameters defining each mode in the system. The result is then combined with optimality conditions for when to switch between consecutive modes. Finally, in Section 4, the proposed optimal decentralization algorithm is showcased

in a simulation that involves tracking a complex drumline-inspired multi-agent trajectory.

II. PRELIMINARIES

In this section we briefly review the graph theory terminology used for describing the system. We then define what it means for a control law to be decentralized, and finally give the system dynamics assumed throughout the rest of the paper.

Given a system of N agents, indexed by $1, \dots, N$, moving in a plane with positions $x_i \in \mathbb{R}^2$, where $i = 1, \dots, N$. Let the graph $G = (V, E)$ describe the information flow amongst agents. The vertex set V is a set of N nodes labeled v_1, \dots, v_N , corresponding to the N agents. The edge set E is defined such that $(v_i, v_j) \in E$ indicates that information is flowing from agent i to j . In this paper it is assumed that the graph describing the information flow is directed and static. For each agent i , the neighbor set $N_i = \{j \mid v_j \in V \wedge (v_j, v_i) \in E\}$ is the set of indices of the agents whose information is available to agent i .

A. Decentralized Control Laws

Decentralized algorithms are used extensively in the realm of computer science for coordinating distributed processors to perform tasks such as leader election and resource allocation. [17] defines these algorithms as having to run concurrently and independently on multiple interconnected processors, where each piece of the algorithm only has access to a limited amount of global information.

Based on that definition, we now explain what it means for a control law to be decentralized when controlling the motion of a multi-agent team: *a control law is decentralized if all notions of direction in the control signal are derived from the pairs of relative distance and angle measurements made between an agent and its neighbors*. The rest of this paper will develop a method to optimally decentralize a multi-agent mission defined at the trajectory-level based on the following assumptions:

Assumption 1: The information flow between agents is given by a predetermined static graph topology.

Assumption 2: Each agent can distinguish the identity of its neighbors from one another.

Assumption 3: The agents have synchronized clocks allowing them to perform open-loop clock-based transitions between different controllers.

Coordinating a team of mobile robots is an example of a situation where these assumptions are reasonable. It is widely accepted in current mobile robotics literature that relative measurements between robots can be obtained in real-time using existing technology. Examples of such include robots that use LIDAR or vision-based sensing. Since these robots all have processors for computing, they also have clocks that can be synchronized and remain synchronized for a period of time afterwards.

Example 2.1: An example of a decentralized control law is a set of N agents executing a weighted consensus control law on their positions, given by

$$\dot{x}_i = -\alpha_i \sum_{j \in N_i} w_{ij} (x_i - x_j),$$

where $\alpha_i, w_{ij} \in \mathbb{R}_+$ are weights on the nodes and edges respectively in the graph. The control law is decentralized because the velocity vector of each agent is completely determined by a linear combination of relative displacement vectors. Most importantly, the control law does not require the agents to share a sense of direction in a global frame.

Example 2.2: A control law where each agent's dynamics are only a function of the relative displacement vectors is not necessarily decentralized. Consider the following modification to the weighted consensus control law, with an additional drift term

$$\dot{x}_i = -\alpha_i \sum_{j \in N_i} w_{ij} (x_i - x_j) + d_i.$$

The additional drift term causes problems since it requires each agent to have a global sense of direction to bias its motion towards.

B. System Dynamics

We now present the dynamics of a multi-agent team using a more generalized, parameterized decentralized control law than weighted consensus. Once again, assume that each agent can measure the relative distance and angle between itself and each of its neighbors. It is possible for an agent to perform simple operations on these vectors such as scaling and rotation. Scaling of the relative displacement vector means multiplying the relative distance measurement by a constant. Rotating the relative displacement vector is done by adding a constant to the relative angle measurement. The scaled and rotated displacement vectors can also be added together using vector addition. A generic decentralized control law should incorporate all three of the mentioned transformations on the displacement vectors.

Assume that there is some mission defined at the trajectory-level for the agents. The trajectory starts at time $t = 0$, and ends at time $t = T$. We introduce a total of K global switch times τ_1, \dots, τ_K into the system that satisfy the constraint

$$0 = \tau_0 \leq \tau_1 \leq \dots \leq \tau_K \leq \tau_{K+1} = T, \quad (1)$$

with the k th mode occurring at the time interval $[\tau_{k-1}, \tau_k)$. The system is therefore a switched autonomous system

with $K + 1$ modes. Under global clock-based switching, each agent has $K + 1$ modes and each of its modes are parameterized by scaling and rotation factors associated with each neighbor.

The dynamics for the i th agent operating in the k th mode are thus

$$\dot{x}_i = - \sum_{j \in N_i} r_{ijk} \text{Rot}(\theta_{ijk})(x_i - x_j), \quad (2)$$

with $r_{ijk} \in \mathbb{R}$ and $\theta_{ijk} \in [0, 2\pi)$ being the constants used for scaling and rotating the displacement vector respectively between itself and agent j . The matrix

$$\text{Rot}(\phi) = \begin{bmatrix} \cos(\phi) & -\sin(\phi) \\ \sin(\phi) & \cos(\phi) \end{bmatrix}$$

defines the two-dimensional rotation matrix for counter-clockwise rotation of a vector. By making the substitutions

$$a_{ijk} = r_{ijk} \cos(\theta_{ijk})$$

$$b_{ijk} = r_{ijk} \sin(\theta_{ijk})$$

and letting

$$M_{ijk} = \begin{bmatrix} a_{ijk} & -b_{ijk} \\ b_{ijk} & a_{ijk} \end{bmatrix}$$

the dynamics in (2) can be rewritten as

$$\dot{x}_i = - \sum_{j \in N_i} M_{ijk} (x_i - x_j). \quad (3)$$

The entire system dynamics can be collected together in matrix form. Let $x \in \mathbb{R}^{2N}$ be

$$x = \begin{bmatrix} x_1 \\ \vdots \\ x_N \end{bmatrix},$$

i.e., x contains the positions of the N agents. The $(2N \times 2N)$ adjacency matrix A_k associated with the k th mode is defined in terms of (2×2) blocks by

$$A_{ijk} = \begin{cases} M_{ijk} & \text{if } (v_j, v_i) \in E \\ 0 & \text{otherwise.} \end{cases} \quad (4)$$

The $(2N \times 2N)$ degree matrix D_k associated with the k th mode is also defined in terms of (2×2) blocks and is given by

$$D_{ijk} = \begin{cases} \sum_{z|(v_z, v_i) \in E} M_{izk} & \text{if } i = j \\ 0 & \text{otherwise} \end{cases} \quad (5)$$

Finally, define the weighted Laplacian L_k associated with the k th mode as

$$L_k = D_k - A_k. \quad (6)$$

The evolution of the switched autonomous system of N agents with $K + 1$ modes is now described by the dynamics

$$\dot{x} = -L_k x \quad \forall t \in [\tau_{k-1}, \tau_k), \quad (7)$$

with mode indices $k = 1, \dots, K + 1$.

III. OPTIMAL DECENTRALIZATION

For a system of N agents with positions $x(t)$ and initial positions $x(0) = x_0$, we wish to obtain a decentralized version of a desired trajectory given by $x_d(t)$. In other words, given a desired trajectory for the multi-agent system, imitate that behavior using only decentralized control laws. The problem can be formulated as an optimal control problem where the objective is to minimize the cost functional

$$J = \frac{1}{2} \int_0^T \|x(t) - x_d(t)\|^2 dt \quad (8)$$

for a system with dynamics (7) by optimizing the parameters a_{ijk} and b_{ijk} for each of the $K+1$ modes, and the K global switch times τ_k , while satisfying (1).

In this section, we first derive the costate dynamics and optimality conditions for optimizing parameterized modes in a general setting. The results are then specialized to the decentralized system (7). Results from previous work on optimizing switching times are presented and also specialized for the decentralized system. The resulting optimality conditions and costate equations can be used in conjunction with a steepest descent algorithm. Together they optimize the modes and global switching times of the decentralized system to minimize J , and thus optimally decentralize $x_d(t)$.

A. Parameterized Mode Optimization

Consider a system that evolves as a switched autonomous system starting at time $t = 0$, and ending at time $t = T$, with $K+1$ modes and K switching times. Each of the modes' dynamics are given by the function f but are parameterized by different scalar parameters c_k for each mode k . The switching times are τ_1, \dots, τ_K satisfying (1), with the k th mode occurring in the time interval $[\tau_{k-1}, \tau_k)$. The dynamics of the system are

$$\dot{x} = F(x, C, t) \quad (9)$$

with

$$F(x, C, t) = f(x, c_k) \quad \forall t \in [\tau_{k-1}, \tau_k), \quad (10)$$

where c_k is free to be chosen and $C = [c_1, \dots, c_{K+1}]^T$. The objective is to choose C so as to minimize the generalized cost functional

$$J = \int_0^T H(x(t)) dt. \quad (11)$$

Theorem 3.1: The optimality condition for each c_k in (10) with respect to cost (11) is

$$\nabla J_{c_k} = \int_{\tau_{k-1}}^{\tau_k} \frac{\partial f}{\partial c_k}(\tau, c_k)^T p(\tau) d\tau = 0, \quad (12)$$

where p is the costate with dynamics

$$\dot{p} = - \left(\frac{\partial F}{\partial x} \right)^T p - \left(\frac{\partial H}{\partial x} \right)^T \quad (13)$$

and boundary condition

$$p(T) = 0. \quad (14)$$

Proof: Perturbing the parameter c_k that defines the k th mode, the dynamics of the perturbed system are

$$\dot{x} + \Delta \dot{x} = \begin{cases} \vdots \\ f(x, c_{k-1}), & t \in [\tau_{k-2}, \tau_{k-1}) \\ f(x + \Delta x, c_k + \Delta c_k), & t \in [\tau_{k-1}, \tau_k) \\ f(x + \Delta x, c_{k+1}), & t \in [\tau_k, \tau_{k+1}) \\ \vdots \end{cases}$$

with $x(0) + \Delta x(0) = x(0) = x_0$. The dynamics of the deviation Δx are given by a first-order approximation as

$$\Delta \dot{x} = \begin{cases} 0, & t \in [0, \tau_{k-1}) \\ \frac{\partial f}{\partial x} \Delta x + \frac{\partial f}{\partial c_k} \Delta c_k, & t \in [\tau_{k-1}, \tau_k) \\ \frac{\partial f}{\partial x} \Delta x, & t \in [\tau_k, \tau_{k+1}) \\ \vdots \end{cases}$$

where $\Delta x(0) = \Delta x(\tau_{k-1}) = 0$. Letting $\Phi(\cdot)$ be the state transition matrix for the Δx linear system, the dynamics can be rewritten as

$$\Delta x = \begin{cases} 0, & t \in [0, \tau_{k-1}) \\ \int_{\tau_{k-1}}^t \Phi(t, \tau) \frac{\partial f}{\partial c_k}(\tau) \Delta c_k d\tau, & t \in [\tau_{k-1}, \tau_k) \\ \Phi(t, \tau_k) \int_{\tau_{k-1}}^{\tau_k} \Phi(\tau_k, \tau) \cdot \frac{\partial f}{\partial c_k}(\tau) \Delta c_k d\tau, & t \in [\tau_k, T]. \end{cases}$$

To derive the optimality conditions, it is necessary to calculate

$$J(c_k + \Delta c_k) - J(c_k) = \langle \nabla J_{c_k}, c_k \rangle = \Delta J_{c_k}, \quad (15)$$

with $\langle \cdot, \cdot \rangle$ denoting the inner product and $\nabla J_{c_k} = \left(\frac{\partial J}{\partial c_k} \right)^T$.

$$\Delta J_{c_k} = \int_0^T (H(x + \Delta x) - H(x)) dt \approx \int_0^T \frac{\partial H}{\partial x} \Delta x dt,$$

which simplifies to

$$\Delta J_{c_k} = \int_{\tau_{k-1}}^{\tau_k} \left(\int_{\tau}^T \frac{\partial H}{\partial x}(t) \Phi(t, \tau) dt \right) \frac{\partial f}{\partial c_k}(\tau) d\tau \Delta c_k.$$

Define the costate

$$p(\tau)^T = \int_{\tau}^T \frac{\partial H}{\partial x}(t) \Phi(t, \tau) dt \quad (16)$$

the expression for ΔJ_{c_k} can be rewritten as

$$\Delta J_{c_k} = \int_{\tau_{k-1}}^{\tau_k} p(\tau)^T \frac{\partial f}{\partial c_k}(\tau) d\tau \Delta c_k.$$

Seeing that the previous equation matches the form in (15), we finally arrive at the optimality condition for c_k , given by

$$\nabla J_{c_k} = \int_{\tau_{k-1}}^{\tau_k} \frac{\partial f}{\partial c_k}(\tau, c_k)^T p(\tau) d\tau. \quad (17)$$

Now it is necessary to derive an expression for the dynamics and boundary conditions of the costate. Taking the time-derivative of the costate defined in (16) results in

$$\dot{p}(\tau)^T = \frac{\partial}{\partial \tau} \left(- \int_{\tau}^T \frac{\partial H}{\partial x}(t) \Phi(t, \tau) dt \right).$$

Applying the chain rule and substituting in the state transition matrix property $\frac{\partial}{\partial \tau} \Phi(t, \tau) = -\Phi(t, \tau) \frac{\partial F}{\partial x}(\tau)$, we get

$$\dot{p}(\tau)^T = -\frac{\partial H}{\partial x}(\tau) - \int_{\tau}^T \frac{\partial H}{\partial x}(t) \Phi(t, \tau) \frac{\partial F}{\partial x}(\tau) dt.$$

Move terms out of the integral to get

$$\dot{p}(\tau)^T = -\left(\int_{\tau}^T \frac{\partial H}{\partial x}(t) \Phi(t, \tau) dt \right) \frac{\partial F}{\partial x}(\tau) - \frac{\partial H}{\partial x}(\tau)$$

where by reapplying the definition of the costate in (16), we see that the costate evolves as

$$\dot{p} = -\left(\frac{\partial F}{\partial x} \right)^T p - \left(\frac{\partial H}{\partial x} \right)^T. \quad (18)$$

The boundary condition for the costate is found by letting $\tau = T$ in (16) to get

$$p(T) = 0. \quad (19)$$

To apply these results to the problem of optimal decentralization, the optimality conditions and costate dynamics need to be specialized for the decentralized system (7) and cost (8). The parameters a_{ijk} and b_{ijk} associated with each of the $K+1$ modes need to be optimized in order to minimize J . Define $a_k = [\dots, a_{ijk}, \dots]^T$ and $b_k = [\dots, b_{ijk}, \dots]^T$ over all valid combinations of i and j allowed by the graph topology. These are vectors containing all the parameters appearing in the system dynamics for a particular mode k .

Corollary 3.1: Optimality conditions for a_k and b_k with respect to cost (8) are given by

$$\nabla J_{a_k} = \int_{\tau_{k-1}}^{\tau_k} \frac{\partial f}{\partial a_k}(\tau, a_k)^T p(\tau) d\tau = 0 \quad (20)$$

$$\nabla J_{b_k} = \int_{\tau_{k-1}}^{\tau_k} \frac{\partial f}{\partial b_k}(\tau, b_k)^T p(\tau) d\tau = 0. \quad (21)$$

The $\frac{\partial f}{\partial a_k}$ and $\frac{\partial f}{\partial b_k}$ matrices can be populated using (2×1) blocks based on

$$\frac{\partial f_i}{\partial a_{ijk}} = -(x_i - x_j) \quad (22)$$

$$\frac{\partial f_i}{\partial b_{ijk}} = -\begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} (x_i - x_j), \quad (23)$$

where f_i is the (2×1) block of f corresponding to the dynamics of agent i .

Proof: The expressions (20) and (21) are the same as (12) with a_k and b_k substituted in for c_k . It is unclear whether or not there exists elegant matrix equations to express $\frac{\partial f}{\partial a_k}$ and $\frac{\partial f}{\partial b_k}$. However, looking at the individual agents' dynamics (3) with a_{ijk} and b_{ijk} substituted in for M_{ijk}

$$\dot{x}_i = -\sum_{j \in N_i} \begin{bmatrix} a_{ijk} & -b_{ijk} \\ b_{ijk} & a_{ijk} \end{bmatrix} (x_i - x_j) \quad (24)$$

we see that a_{ijk} and b_{ijk} do not appear anywhere else except in agent i 's dynamics. Therefore, it can be differentiated by

a_{ijk} and b_{ijk} to obtain the expressions for $\frac{\partial f_i}{\partial a_{ijk}}$ and $\frac{\partial f_i}{\partial b_{ijk}}$. Those results can then be put composed to obtain $\frac{\partial f}{\partial a_k}$ and $\frac{\partial f}{\partial b_k}$. ■

The decentralized system dynamics (7) and cost functional (8) also need to be substituted into the costate dynamics (13).

Corollary 3.2: Costate dynamics for calculating the optimality conditions of a_k and b_k in (20) and (21) are

$$\dot{p}(t) = L_k^T p(t) - x(t) + x_d(t) \quad \forall t \in [\tau_{k-1}, \tau_k]. \quad (25)$$

B. Switch Time Optimization

Optimality conditions for switching times in a switched autonomous system were derived in [18]. They are restated here, for the sake of easy reference:

Theorem 3.2: The optimality condition with respect to cost functional (11) for switching times in a switched autonomous system where mode k has dynamics f parameterized by c_k is

$$\frac{\partial J}{\partial \tau_k} = p(\tau_k)^T (f(x(\tau_k), c_k) - f(x(\tau_k), c_{k+1})) = 0. \quad (26)$$

The costate dynamics are the same as (13) and boundary conditions (14).

Corollary 3.3: The switch time optimality conditions specialized for the system (7) with cost (8) are given by

$$\frac{\partial J}{\partial \tau_k} = p(\tau_k)^T (L_{k+1} - L_k) x(\tau_k). \quad (27)$$

with costate dynamics the same as (25).

IV. SIMULATION RESULTS

The proposed optimal decentralization algorithm is demonstrated in a simulation where agents are tasked with tracking a complex trajectory inspired by drumline routines. Drumline formations are traditionally designed by choreographers to be executed in a centralized manner. The position and path taken by band members at each moment in time have been predetermined to a high level of detail. As a result, band members spend a lot of time practicing to follow these predetermined paths. However, such an approach requires each band member to memorize paths taken throughout the entire dance sequence and have global sensing capabilities to know if they're in the correct position. Optimal decentralization is used to mimic the original routine with high fidelity using decentralized control laws.

The optimal decentralization algorithm was used on a drumline-inspired trajectory involving $N = 21$ agents with arbitrarily chosen initial values for each of their parameterized modes. A total of $K = 22$ global switching times were inserted initially evenly spaced between the starting time $t = 0$ and ending time $t = T = 10.78$. The system therefore could to have up to 23 modes. A variant of the standard steepest descent with Armijo stepsize algorithm was used to stochastically take turns optimizing the parameterized modes with high probability and switch times with low probability to drive the cost J to a local minimum. The reason for introducing stochasticity is because in practice, optimizing

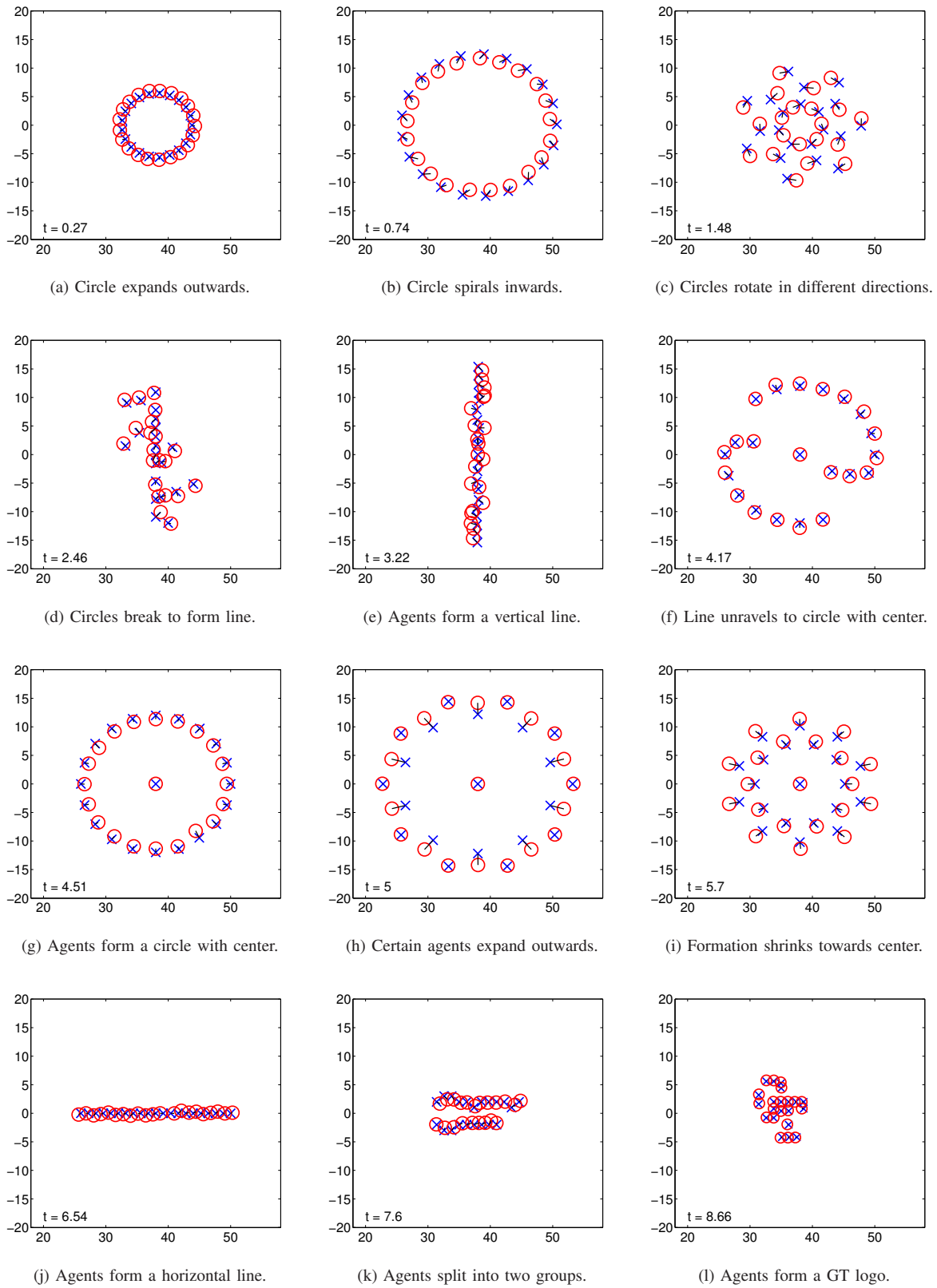


Fig. 1: Simulation of optimally decentralized version of a drumline-inspired dance with $N = 21$ agents. The resulting locations of the agents are marked by O's with lines connecting them to their desired location marked by X's.

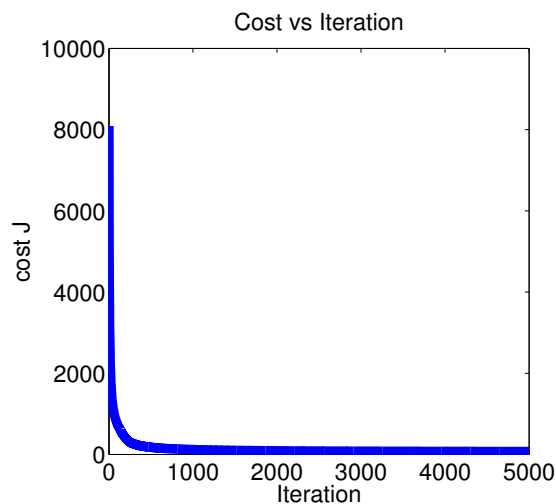


Fig. 2: Convergence of cost J after running steepest descent for 5000 iterations.

the switch times too much before the modes themselves have had time to become well-defined tends to drive the cost to undesirable local minima with high final costs. The convergence of the cost J after a run of 5000 iterations is shown in Figure 2.

The optimally decentralized trajectories resulting from the optimization are shown in Figure 1 where the actual locations of the agents are marked by O's with lines connecting them to their desired location marked by X's. From the simulation results, it is clear that the resulting decentralized control laws successfully mimicked the original trajectory.

V. CONCLUSIONS

This paper defined and gave examples of what it means for a controller to be decentralized in the setting of controlling multi-agent trajectories. A decentralized control law in the form of a switched autonomous system was proposed where each mode was parameterized by constants defining the interaction of each agent with its neighbors. The problem of optimally decentralizing a target trajectory was presented and posed as an optimal control problem. Optimality conditions for each agent's parameter-defined modes were derived and combined with optimality conditions for the global switching times. The derived optimality conditions were used to optimize the system to mimic the desired trajectory using only decentralized control laws. Simulation results showed that the resulting decentralized trajectory closely matched the original desired trajectory.

The optimal decentralization algorithm proposed in this paper results in decentralized control laws that are open-loop and assumes that each agent can perform clock-based mode switching. Future work will involve finding closed-loop control laws to achieve the same result and replacing clock-based switching with locally-detectable event-based switching.

Acknowledgments

The authors wish to acknowledge ONR for supporting this work through MURI HUNT, and Prof. Christopher Moore for discussions about the Georgia Tech drumline.

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, Sept. 2004.
- [2] H. G. Tanner, A. Jadbabaie, and G. J. Pappas, "Flocking in fixed and switching networks," *IEEE Trans. Autom. Control*, vol. 52, no. 5, pp. 863–868, May 2007.
- [3] 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, Jun. 2003.
- [4] W. Ren and R. W. Beard, "Consensus seeking in multiagent systems under dynamically changing interaction topologies," *IEEE Trans. Autom. Control*, vol. 50, no. 5, pp. 655–661, May 2005.
- [5] M. Ji and M. Egerstedt, "Distributed coordination control of multi-agent systems while preserving connectedness," *IEEE Trans. Robot. Autom.*, vol. 23, no. 4, pp. 693–703, Aug. 2007.
- [6] H. G. Tanner, G. J. Pappas, and V. Kumar, "Leader-to-formation stability," *IEEE Trans. Robot. Autom.*, vol. 20, no. 3, pp. 443–455, Jun. 2004.
- [7] T. Eren, W. Whiteley, B. D. O. Anderson, A. S. Morse, and P. N. Belhumeur, "Information structures to secure control of rigid formations with leader-follower architecture," in *Proc. 2005 American Control Conf.*, Portland, OR, 2005, pp. 2966–2971.
- [8] I. D. Couzin and N. R. Franks, "Self-organized lane formation and optimized traffic flow in army ants," in *Proc. Royal Society of London, Series B* 270, 2003, pp. 139–146.
- [9] Z. Lin, M. Broucke, and B. Francis, "Local control strategies for groups of mobile autonomous agents," *IEEE Trans. Autom. Control*, vol. 49, no. 4, pp. 622–629, Apr. 2004.
- [10] Y. Liu, K. M. Passino, and M. Polycarpou, "Stability analysis of one-dimensional asynchronous swarms," *IEEE Trans. Autom. Control*, vol. 48, no. 10, pp. 1848–1854, Oct. 2003.
- [11] 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.
- [12] L. Xiao, S. Boyd, and S. Lall, "A space-time diffusion scheme for peer-to-peer least squares estimation," in *Proc. 5th Int. Conf. on Information Processing in Sensor Networks*, Nashville, TN, 2006, pp. 168–176.
- [13] M. Rotkowitz and S. Lall, "A characterization of convex problems in decentralized control," *IEEE Trans. Autom. Control*, vol. 51, no. 2, pp. 274–286, Feb. 2006.
- [14] B. Bamieh, F. Paganini, and M. A. Dahleh, "Distributed control of spatially invariant systems," *IEEE Trans. Autom. Control*, vol. 47, no. 7, pp. 1091–1107, Jul. 2002.
- [15] A. Rantzer, "A separation principle for distributed control," in *Proc. 45th IEEE Conf. on Decision and Control*, San Diego, CA, 2006, pp. 3609–3613.
- [16] N. Motee and A. Jadbabaie, "Optimal control of spatially distributed systems," *IEEE Trans. Autom. Control*, vol. 53, no. 7, pp. 1616–1629, Aug. 2008.
- [17] N. A. Lynch, *Distributed Algorithms*, Morgan Kaufmann, 1996.
- [18] M. Egerstedt, Y. Wardi, and H. Axelsson, "Transition-time optimization for switched-mode dynamical systems," *IEEE Trans. Autom. Control*, vol. 51, no. 1, pp. 110–115, Jan. 2006.