Distributed motion constraints for algebraic connectivity of robotic networks

IEEE Conference on Decision and Control

December, 2008

Michael Schuresko

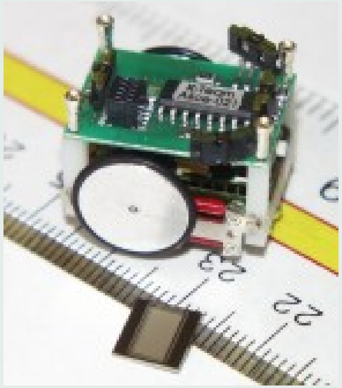
Jorge Cortés

Applied Mathematics and Statistics
University of California at Santa Cruz

Mechanical and Aerospace Engineering
University of California at San Diego

Distributed control of swarms

Schooling fish



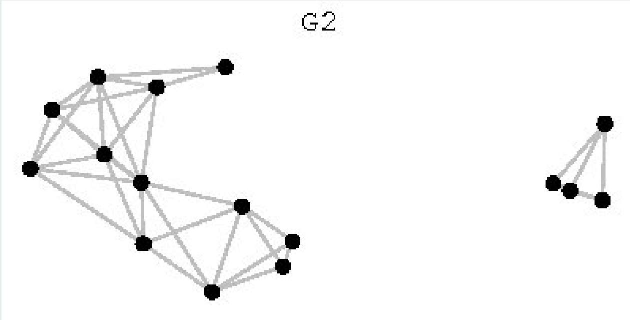
Tiny robot, courtesy (CS), see (CAS00)

(i) Large number of robots with limited communication.

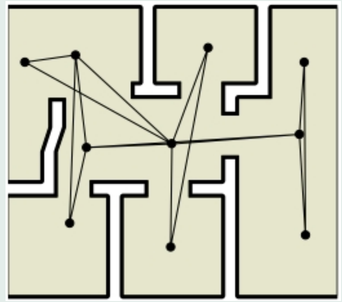
(ii) Control algorithm and communication law on each robot.

Goal is to write control algorithm and communication law for individual robots such that the whole swarm achieves some collective task.

Spatially-induced graphs



R-disk communication graph



Visibility graph

We model communication networks with spatially induced graphs

(i) Set of robot positions induce ***graph,*** $G = (V, E)$.

(ii) Edge between robots i and j indicates communication is possible between i and j.

(iii) Mapping between set of positions and graph should be invariant under permutation of robot identities

Here we show the r***-disk graph***. We like to pick graphs which are reasonable, but crude, approximations of how wireless networks might actually behave.

Robotic Network Model

Network model equivalent to (MBCF07).

Each robot runs a discrete time ***communications law***. At particular time slices, robots communicate with neighbors over proximity graph, and modify values stored in ***logic variables***.

Each robot, i, runs a continuous time ***control law*** which controls the motion of robot i based on i's position state and logic variables. Robots are fully actuated. In our case, they live in $\mathbb{R}^2$.

Outline

(i) Background

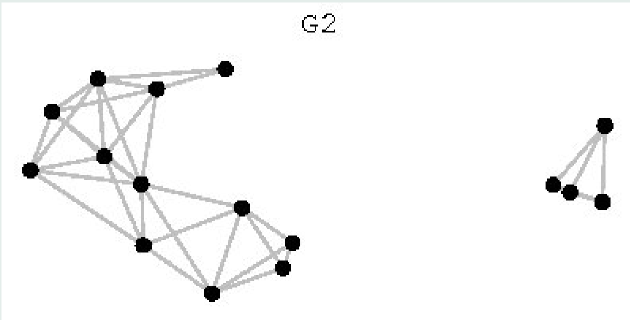
(ii) ***Algebraic Connectivity***

(iii) Key idea / game

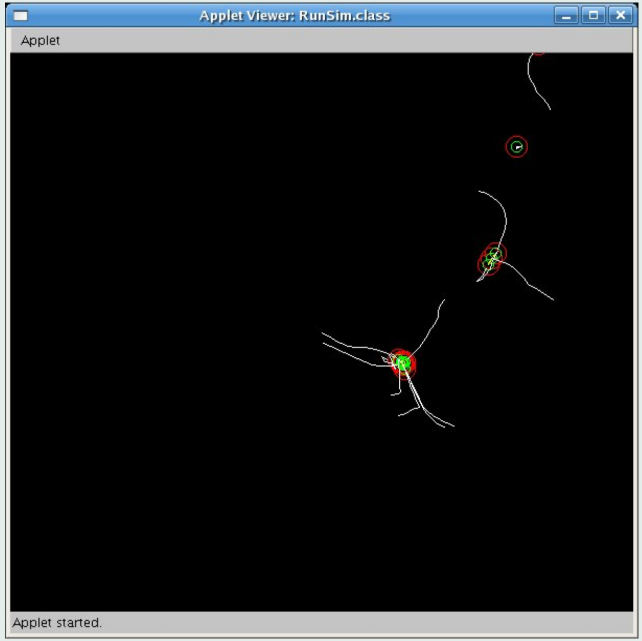
(iv) Final algorithm

(v) Conclusions / Future work

Connectivity and collective behavior



R-disk communication graph



Failure to rendezvous due to lapse in connectivity

If communication network becomes disconnected, it is, at best, as if we have two smaller swarms.

(go through this quickly)

Algebraic Graph Theory

Given a graph, $G = (V, E)$, the Laplacian matrix, $L(G)$ be the matrix

$$L_{i,j} = \begin{cases} -1 & (i,j) \in E \\ \text{deg}(i) & i = j \\ 0 & \text{otherwise} \end{cases}$$

If the graph is weighted, i.e. for each $(i, j) \in E$ there is a $w_{i,j} \in \mathbb{R}$, we can define a weighted Laplacian matrix $L(G)$ by

$$L_{i,j} = \begin{cases} -w_{i,j} & (i,j) \in E \\ \sum_{k \neq i} (w_{i,k}) & i = j \\ 0 & \text{otherwise} \end{cases}$$

The Graph Laplacian

The Laplacian has several nice properties

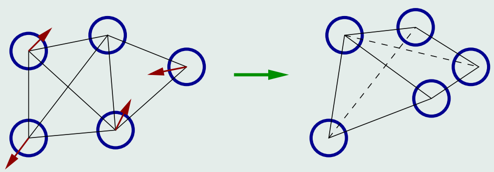
- $L\mathbf{1} = \mathbf{0}$
- The multiplicity of the zero eigenvalue is the number of components of the graph.
- The speed of convergence of common control algorithms for flocking, rendezvous and consensus depend on the second smallest eigenvalue, λ_2 of the Laplacian matrix of the communication graph.

Problem Setup (1 of 2)

Suppose a communication graph for a swarm of robots is weighted, and the weights depend on the relative positions of the two robots sharing a communication link.

Then $\lambda_2(L(G))$ depends on the positions of the robots in the swarm.

An instantaneous motion of a robot creates an instantaneous change in $\lambda_2(L(G))$.



Evolution of graph

Problem Setup (2 of 2)

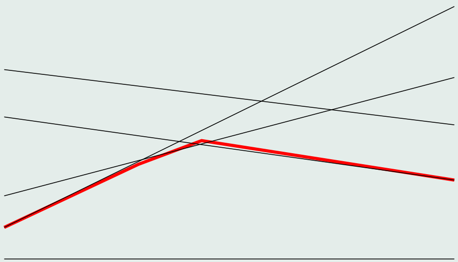
Whenever $L(G)$ has a distinct second smallest eigenvalue, gradient of $\lambda_2(L(G))$ with respect to $L(G)$ is $v_2 v_2^T$ where $L(G)v_2 = \lambda_2(L(G))v_2$.

Nonsmooth. Let $f_{\lambda_i}(L)$ map $L \in \mathrm{Sym}(n)$ to $\lambda_i(L)$.

Nonsmooth gradient is:

$$f^{\circ}_{\lambda_i}(M; X) = \max_{\{v \in \mathbb{S}^n \,:\, Mv = \lambda_i v\}} vv^T \bullet X,$$

$$\partial f_{\lambda_i}(M) = \mathrm{co}_{\{v \in \mathbb{S}^n \,:\, Mv = \lambda_i v\}}\{vv^T\}.$$



Example nonsmooth function

Notation

Quickly

LAP$(n) \subseteq$ Sym(n) is the space of valid Laplacian matrices, i.e. $L \in$ LAP(n) implies $L\mathbf{1} = \mathbf{0}$ and $L_{i,j} \leq 0$ for $i \neq j$.

LAP$_{\pm}(n)$ is an extension of this space. Lacks the $L_{i,j} \leq 0$ requirement. Rates of change of a Laplacian matrices live in LAP$_{\pm}(n)$

$A \leq_{\text{LAP}} B$ if and only if $A_{i,j} \geq B_{i,j}$ for all $i \neq j$.
Interval $[A, B]_{\text{LAP}}$ for $A, B \in$ LAP$_{\pm}(n)$ defined in the natural way.

Prior work

Prior work revolves around finding gradient of f_{λ_2} in space of robot positions and moving in direction of that gradient.

Difficult to do in a distributed fashion. Centralized solutions include (Boy06) and (KM06).

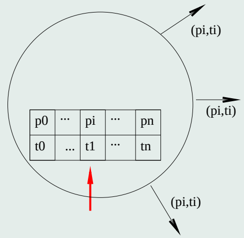
Decentralized solution (dGJ06) follows gradient approach

- Communication complexity required to compute eigenvalue
- Nonsmoothness of eigenvalue gradient.

(see also: Yang and Freeman, Zavlanos and Pappas)

Our solution

- ***Information dissemination*** algorithm.



- Each robot has ***bounds on value of Laplacian matrix***
- Game against world-picking opponent. (appears in next section)

Outline

(i) Background

(ii) Algebraic Connectivity

(iii) ***Key idea / game***

(iv) Final algorithm

(v) Conclusions / Future work

Our solution (1 of 4)

Game in space of matrices.

Given $A, B \in \mathsf{LAP}(n)$ which bound Laplacian, find $X \in \mathsf{LAP}_{\pm}(n)$ which ***doesn't decrease*** λ_2 for any Laplacian in range ***with*** $\lambda_2 \in [\lambda_-, \lambda_+]$.

λ_+, λ_- and λ_2 over time.

Our solution (2 of 4)

Game in space of matrices.

Given $A, B \in \text{LAP}(n)$, $A \leq_{\text{LAP}} B$, and $\lambda_+ \in \mathbb{R}$ find a direction $X \in \text{LAP}_\pm(n)$ having $X \bullet V \geq 0$ for every V in the generalized gradient of some $L \in [A, B]_{\text{LAP}}$ having $f_{\lambda_2}(L) \leq \lambda_+$.

Suffices to find $X \in \text{LAP}_\pm(n)$ having $X \bullet (vv^T) \geq 0$ for every v having $Lv = f_{\lambda_2}(L)v$ for some $L \in [A, B]_{\text{LAP}}$ having $f_{\lambda_2}(L) \leq \lambda_+$.

Our solution (3 of 4)

Find set enclosing the set of every v having $Lv = f_{\lambda_2}(L)v$ for some $L \in [A, B]_{\mathsf{LAP}}$ having $f_{\lambda_2}(L) \leq \lambda_+$.

Such a v must have 2 components.

- Component in m lowest eigenvectors for some m having $f_{\lambda_{m+1}}(A) \geq \lambda_+$
- Component in other eigenvectors of a small enough magnitude that multiplying by $f_{\lambda_{m+1}}(A)$ and adding to contribution of other component to Lv keeps Lv under $\lambda_+ v$.

Our solution (4 of 4)

Pick basis for first component, $M_u(m)$. Pick ball radius enclosing second component, $\epsilon_A(m) = \sqrt{\frac{\lambda_+ - \lambda_2(A)}{\lambda_{m+1}(A) - \lambda_2(A)}}$

For a proposed direction in the space of Laplacian matrices, $X \in \mathsf{LAP}_\pm(n)$

- ***Compute*** $\min(\mathsf{eigs}(M_u^T(m) X M_u(m)))$ and $\min(\min(\mathsf{eigs}(X)), 0)$ (columns of $M_u(m)$ are eigenvectors of A of eigenvalue $< \lambda_{m+1}(A)$).
- ***If*** $(1 - \epsilon_A(m)^2) \min(\mathsf{eigs}(M_u^T(m) X M_u(m))) + \epsilon_A(m)^2 \min(\min(\mathsf{eigs}(X)), 0) \geq 0$ ***then*** direction is "safe"

Actually determines if all Y having $X \leq_{\mathsf{LAP}} Y$ win game.

Outline

(i) Background

(ii) Algebraic Connectivity

(iii) Key idea / game

(iv) ***Final algorithm***

(v) Conclusions / Future work

Motion Test Algorithm

Given proposed motion by an individual robot, compute ***lower bound*** on instantaneous rate of change of Laplacian matrix. If the actual (unknown) rate of change is Y, we want X (known) having $X \leq_{\text{LAP}} Y$.

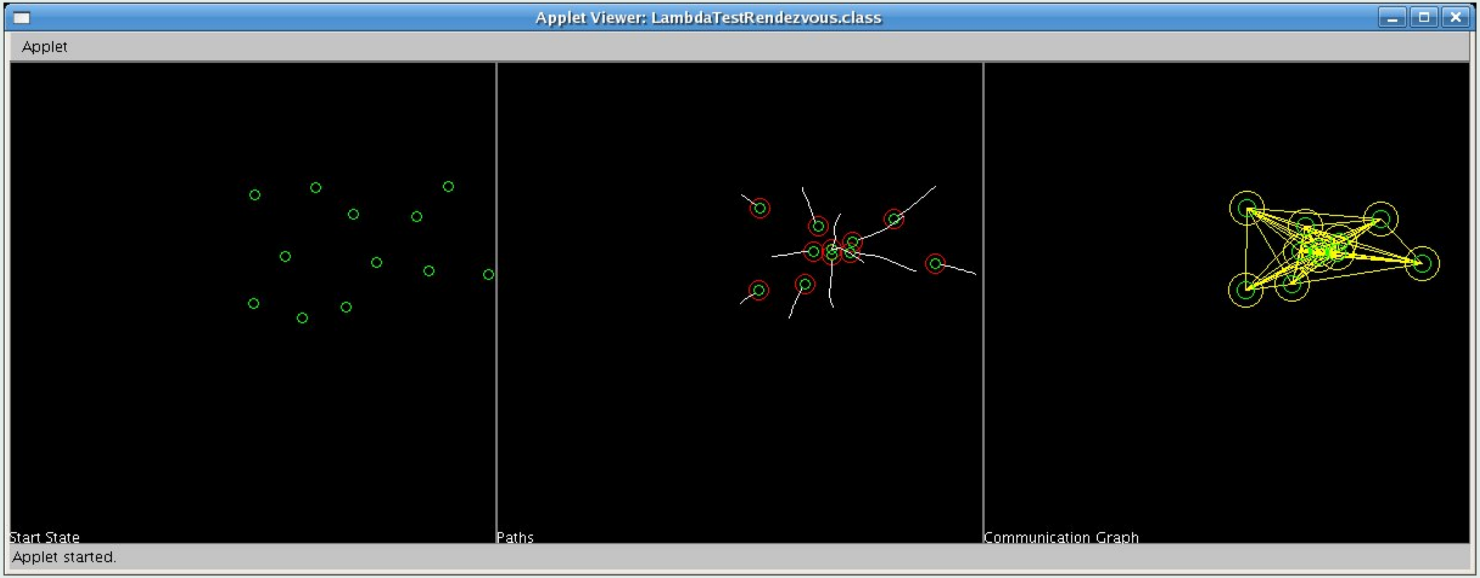
If each robot moves in a direction such that the associated Laplacian rate of change satisfies eigenvalue test, ***then*** $f_{\lambda_2}(L(G))$ does not drop below λ_+.

Motion Projection Algorithm

Combine this with a ***root finder*** on the space of physical directions of robot motion.

Gives an algorithm which finds valid directions.

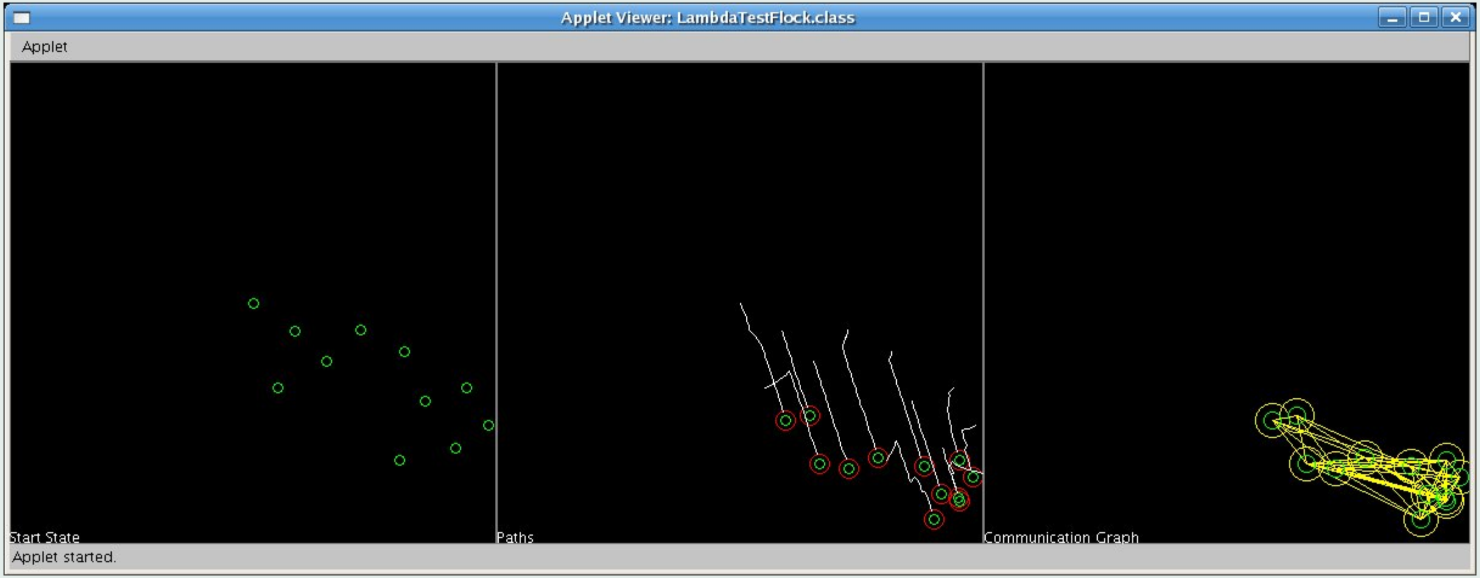
Example 1 : Rendezvous



Without connectivity constraints

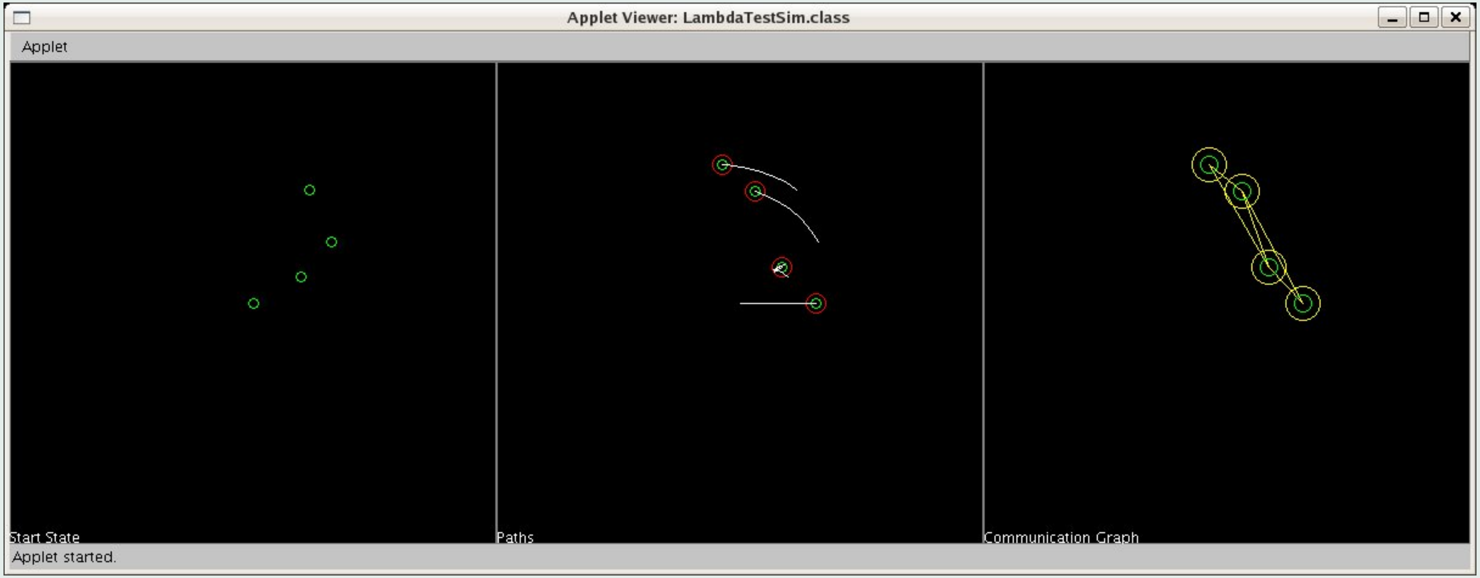
Sim

Example 2 : Flocking



Sim

Example 3 : Multiple control directives



Sim

Outline

(i) Background

(ii) Algebraic Connectivity

(iii) Key idea / game

(iv) Final algorithm

(v) ***Conclusions / Future work***

Conclusions

Connectivity constraints are realizable

- In a more flexible setting than previously thought.
- Without explicit global transfer of information
- In a manner which can be coupled with a wide set of algorithms

Work presented in

- M. D. Schuresko and J. Cortés. Distributed motion constraints for algebraic connectivity of robotic networks. In *IEEE Conf. on Decision and Control*, Cancun, Mexico, December 2008. Accepted. To appear
- M. D. Schuresko and J. Cortés. Distributed motion constraints for algebraic connectivity of robotic networks. In *Journal of Intelligent and Robotic Systems*, 2008. Special issue on "Special Issue on Unmanned Autonomous Vehicles." Submitted

Also see M. D. Schuresko. CCLsim. a simulation environment for robotic networks, 2008. Electronically available at http://www.soe.ucsc.edu/˜mds/cclsim.

Future work

- Understanding when IDEALIZED MOTION TEST ALGORITHM causes agents to lock up.
- Understanding when MOTION TEST ALGORITHM causes agents to lock up and IDEALIZED MOTION TEST ALGORITHM does not.
- Either improving "agents lock up" problem with MOTION TEST ALGORITHM or characterizing conditions on t_{cmm}, v_{max} and the agent communication radius which reduce lock-up to an acceptable level.