Coordination in Multi-Agent Networks

9th International Workshop on
Hybrid Systems: Computation and Control (HSCC)
Santa Barbara, March 29-31, 2006

Also: UCLA, April 21, 2006 and Caltech, May 15, 2006

Francesco Bullo

Mechanical Engineering
Center for Control, Dynamical Systems and Computation

University of California at Santa Barbara
http://motion.mee.ucsb.edu

Ack: Anurag Ganguli, Sara Susca, Ketan Savla
Jorge Cortés (UCSC), Emilio Frazzoli (UCLA), Sonia Martínez (UCSD)
AFOSR MURI Appers, ARO MURI Swarms, NSF Sensors, ONR YIP

Multi-agent networks

What kind of systems?
Groups of systems with control, sensing, communication and computing

Individual members in the group can

- sense its immediate environment
- communicate with others
- process the information gathered
- take a local action in response


Wildebeest herd in the Serengeti


Geese flying in formation


Atlantis aquarium, CDC 2004

Example networks from engineering

Sensor networks and robotic sensor networks
embedded systems, distributed robotics

General actuator/sensor networks
robocup, air-traffic systems

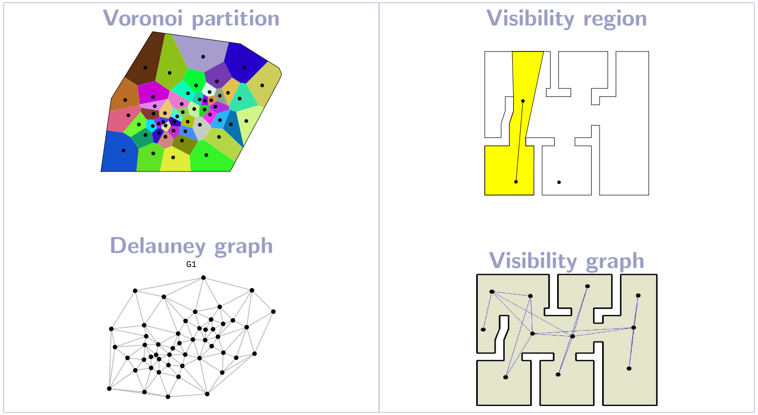

High altitude long endurance UAVs

Broad challenge

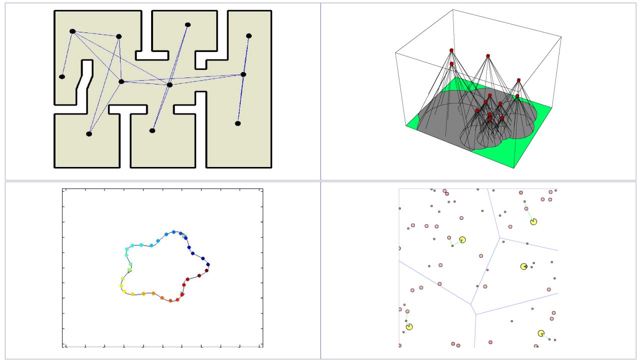
Useful engineering through small, inexpensive, limited-comm vehicles/sensors

Problem	lack of understanding of how to assemble and co-ordinate individual devices into a coherent whole
1. Feedback	rather than open-loop computation for known/static setup
2. Information flow	who knows what, when, why, how
3. Optimization	design efficient algorithms
Approach	integration of control, comm, sensing, computing

Geometric tools



Research in animation



(i) elementary motion tasks (deployment, rendezvous, self-assembly)

(ii) sensing tasks (map building, localization, vehicle routing, search)

Outline

I: Models for Multi-Agent/Robotic Networks

II: Algorithms and Tools

Part I: Models for Multi-Agent Networks

References

(i) I. Suzuki and M. Yamashita. Distributed anonymous mobile robots: Formation of geometric patterns. *SIAM Journal on Computing*, 28(4):1347–1363, 1999

(ii) N. A. Lynch. *Distributed Algorithms*. Morgan Kaufmann Publishers, San Mateo, CA, 1997. ISBN 1558603484

(iii) D. P. Bertsekas and J. N. Tsitsiklis. *Parallel and Distributed Computation: Numerical Methods*. Athena Scientific, Belmont, MA, 1997. ISBN 1886529019

(iv) S. Martínez, F. Bullo, J. Cortés, and E. Frazzoli. On synchronous robotic networks – Part I: Models, tasks and complexity. *IEEE Transactions on Automatic Control*, April 2005. Submitted

Objective

(i) meaningful + tractable model

(ii) feasible operations and their cost

(iii) control/communication tradeoffs

Part I: Robotic network

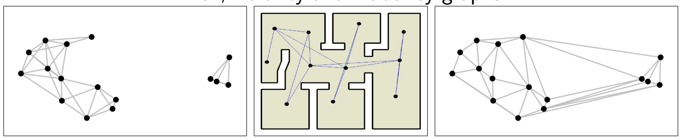
A **uniform/anonymous robotic network** $\mathcal{S}$ is

(i) $I = \{1, \dots, N\}$; **set of unique identifiers (UIDs)**

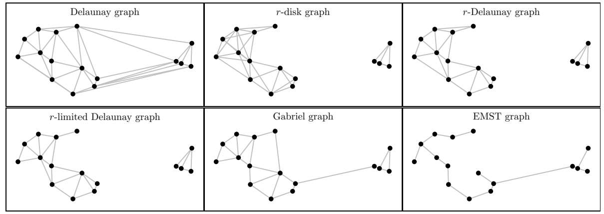
(ii) $\mathcal{A} = \{A_i\}_{i \in I}$, with $A_i = (X, U, f)$ is a **set of physical agents**

(iii) **interaction graph**

Disk, visibility and Delauney graphs



Communication models for robotic networks



Relevant graphs

(i) fixed, directed, balanced

(ii) switching

(iii) geometric or state-dependent

(iv) random, random geometric

Message model

(i) message

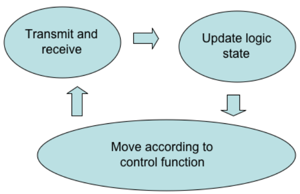
(ii) packet/bits

(iii) absolute or relative positions

(iv) packet losses

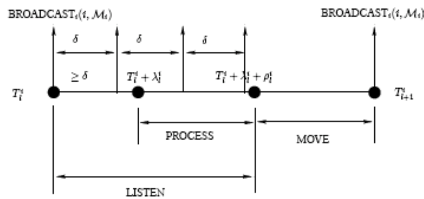
Synchronous control and communication

(i) **communication schedule**	$\mathbb{T} = \{t_\ell\}_{\ell \in \mathbb{N}_0} \subset \mathbb{R}_{\geq 0}$
(ii) **communication alphabet**	L including the `null` message
(iii) set of values for **logic variables**	W
(iv) **message-generation function**	$\text{msg}\colon \mathbb{T} \times X \times W \times I \to L$
(v) **state-transition functions**	$\text{stf}\colon \mathbb{T} \times W \times L^N \to W$
(vi) **control function**	$\text{ctrl}\colon \mathbb{R}_{\geq 0} \times X \times W \times L^N \to U$



Asynchronous extensions

- each agent has different communication/activation schedule
- bounded delay between transmission and reception
- each agent performs cycle



Task and complexity

- Coordination task is $(\mathcal{W}, \mathcal{T})$ where $\mathcal{T}\colon X^N \times \mathcal{W}^N \to \{\mathtt{true}, \mathtt{false}\}$

  Logic-based: achieve consensus, synchronize, form a team
  Motion: deploy, gather, flock, reach pattern
  Sensor-based: search, estimate, identify, track, map

- For $\{\mathcal{S}, \mathcal{T}, \mathcal{CC}\}$, define costs/complexity:
  control effort, communication packets, computational cost
- Time complexity to achieve $\mathcal{T}$ with $\mathcal{CC}$

$$\mathrm{TC}(\mathcal{T}, \mathcal{CC}, x_0, w_0) = \inf\ \{\ell \mid \mathcal{T}(x(t_k), w(t_k)) = \mathtt{true}\,,\ \text{for all } k \geq \ell\}$$
$$\mathrm{TC}(\mathcal{T}, \mathcal{CC}) = \sup\{\,\mathrm{TC}(\mathcal{T}, \mathcal{CC}, x_0, w_0) \mid (x_0, w_0) \in X^N \times \mathcal{W}^N\}$$
$$\mathrm{TC}(\mathcal{T}) = \inf\{\,\mathrm{TC}(\mathcal{T}, \mathcal{CC}) \mid \mathcal{CC} \text{ achieves } \mathcal{T}\}$$

Outline

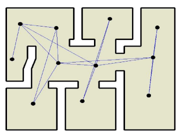
I: Models for Multi-Agent/Robotic Networks
II: Algorithms and Tools
- S1: visibility-based deployment
- S2: deployment
- S3: rendezvous

Scenario 1: Visually-based deployment for guards

Model

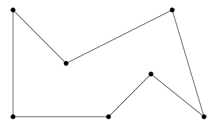
(i) Environment: non-self-intersecting polygon Q
(ii) Sensing and communication within visibility polygon
(iii) Asynchronous operation

Objective: Achieve complete visibility of nonconvex environment
(with simultaneous map building)



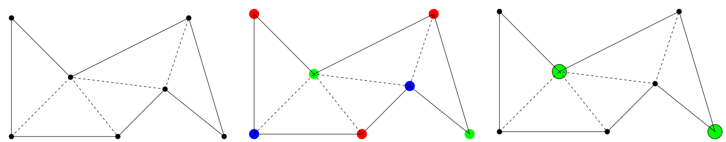
S1: Art Gallery Problem and Theorem

Art Gallery Problem (Klee '73):
Imagine placing guards inside a nonconvex polygon with n vertices: how many guards are required and where should they be placed in order for each point in the polygon to be visible by at least one guard?

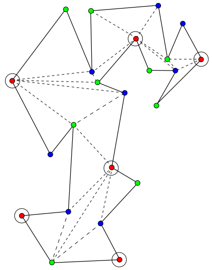


Theorem (Chvátal '75): $\lfloor n/3 \rfloor$ guards are sufficient and sometimes necessary

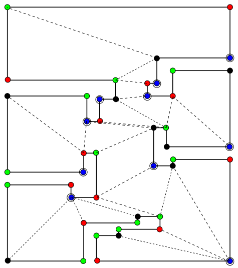
"Triangulation + coloring" proof (Fisk '78):



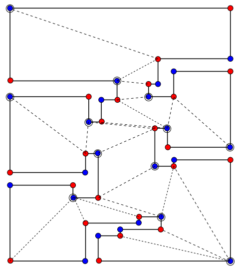
Art Gallery Theorems



- Fisk '78
- $\lfloor \frac{n}{3} \rfloor$ sufficient and occasionally necessary



- Kahn, Klawe, Kleitman '93
- $\lfloor \frac{n}{4} \rfloor$ sufficient and occasionally necessary



- Pinciu '03
- $\frac{n}{2} - 2$ sufficient and occasionally necessary

S1: Approach

Approach:

Geometric Structure

Local Navigation and Global Exploration

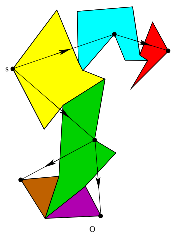
Distributed Information Processing

(i) **Represent the environment by a graph**

(ii) Node-to-node navigation and deployment over a graph

(iii) Distributed information exchange

S1: Vertex-induced tree



- the graph $\mathcal{G}_Q(s)$ is a rooted tree
- maximum # nodes in the vertex-induced tree is $\left\lfloor \frac{n}{2} \right\rfloor$, where $n = |\operatorname{Ve}(Q)|$

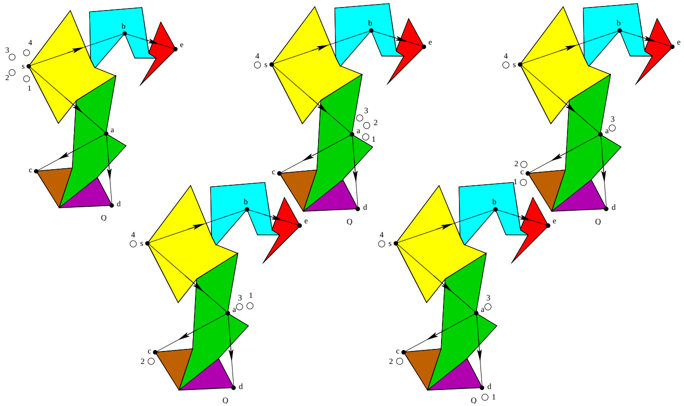
S1: Approach

(i) Represent the environment by a graph

(ii) **Node-to-node navigation and deployment over a graph**

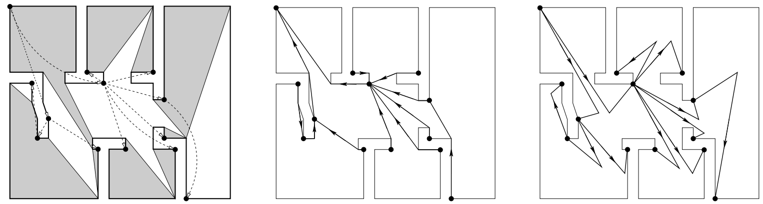
(iii) Distributed information exchange

S1: Depth-first deployment



S1: Node-to-node navigation

the planned paths "from node to parent" and "from node to children:"



S1: Algorithm

Assume: All agents initially at root s

Incremental partition and depth-first deployment

```
At each comm round:
compares UID with agents at the same node
if i is largest UID then
  stay
else
  obtain W from agent with maximum UID
  move according to depth-first deployment
end if
```

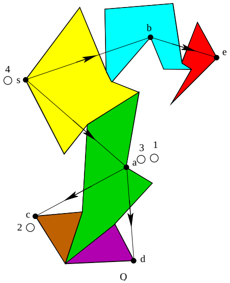
S1: Approach

(i) Represent the environment by a graph

(ii) Node-to-node navigation and deployment over a graph

(iii) **Distributed information exchange**

S1: Geographic info required for navigation



- Required memory: $W = \{p_{\text{parent}}, p_{\text{last}}, v', v''\}$
  p_{parent} is parent node to current agent's position
  p_{last} is last node visited by the agent
  (v', v'') is the gap toward the parent node
- Init: four values set to the initial agent position
- Actions:

msg W broadcast together with UID

stf After move from k_{parent} to k_{child} through gap g_1, g_2, update: $p_{\text{parent}} := k_{\text{parent}}$, $p_{\text{last}} := k_{\text{parent}}$, $(v', v'') := (g_1, g_2)$

stf After move from k_{child} to k_{parent}, update: $p_{\text{last}} := k_{\text{child}}$ and agent acquires correct $\{p_{\text{parent}}, v', v''\}$ from incoming messages

S1: Convergence and complexity results

(i) At least one agent on $\min\{N, |\mathcal{G}_Q(s)|\}$ nodes of $\mathcal{G}_Q(s)$ in time

$$t^* \leq \mathcal{T}_{\text{motion}} + \mathcal{T}_{\text{comm/sens/proc}},$$

where $\mathcal{T}_{\text{motion}} \leq 2\Big(\mathcal{L}_{\text{ford}}(\mathcal{G}_Q(s)) + \mathcal{L}_{\text{back}}(\mathcal{G}_Q(s))\Big)$ and $\mathcal{T}_{\text{comm/sens/proc}} \leq 2(|\mathcal{G}_Q(s)| - 1)(\max \lambda_l^i + \rho_l^i)$

(ii) Task achieved for $N \geq \frac{n}{2}$

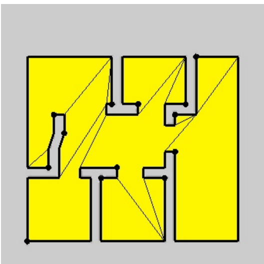
(iii) as N and $n \to +\infty$, if the diameter of Q is bounded, then

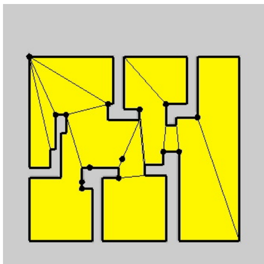
$$t^* \in O(\min\{N, n\}) \quad \Longrightarrow \quad \mathrm{TC}(\mathcal{T}_{\text{vis-based deployment}}, \mathcal{CC}_{\text{partition+deployment}}) \in O(N)$$

(iv) in worst case, constant factor optimal run-time because

$$\mathrm{TC}(\mathcal{T}_{\text{vis-based deployment}}) \in \Omega(n)$$

S1: Simulations and extensions





S1: Robustness properties

Communication delays
Arbitrary bounded delays tolerated

Packet drops
As long as packets are “not dropped always”

Changing environments
Sudden "opening of a door”

Agent arrivals and departures
Completely robust to arrivals at the root
Partially/completely robust to departures depending on deployment algorithm

Sensor noise
Precise deployment still achieved

Outline

I: Models for Multi-Agent/Robotic Networks

II: Algorithms and Tools

- **S1:** visibility-based deployment
- **S2:** deployment
- **S3:** rendezvous

Deployment

(i) J. Cortés, S. Martínez, T. Karatas, and F. Bullo. Coverage control for mobile sensing networks. *IEEE Transactions on Robotics and Automation*, 20(2):243–255, 2004

(ii) A. Howard, M. J. Matarić, and G. S. Sukhatme. Mobile sensor network deployment using potential fields: A distributed scalable solution to the area coverage problem. In *International Conference on Distributed Autonomous Robotic Systems (DARS02)*, pages 299–308, Fukuoka, Japan, June 2002

(iii) J. Cortés, S. Martínez, and F. Bullo. Spatially-distributed coverage optimization and control with limited-range interactions. *ESAIM. Control, Optimisation & Calculus of Variations*, 11:691–719, 2005

(iv) A. Ganguli, J. Cortés, and F. Bullo. Distributed deployment of asynchronous guards in art galleries. In *American Control Conference*, Minneapolis, MN, June 2006. To appear

Scenario 2: Deployment problems and algorithms

ANALYSIS of cooperative distributed behaviors

(i) how do animals share territory?
what if every fish in a swarm goes toward center of own dominance region?

Barlow et al, Animal Behavior '74

(ii) what if each vehicle moves away from closest vehicle(s)?

DESIGN of performance metric

(iii) how to cover a region with n minimum radius overlapping disks?

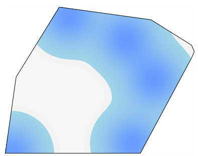
(iv) how to design a minimum-distortion (fixed-rate) vector quantizer? (Lloyd '57)

(v) where to place mailboxes in a city / cache servers on the internet?

(vi) how to place cameras to illuminate environment?

S2: General multi-center function

Objective: Given agents $(p_1, \dots, p_n)$ in convex environment Q
unspecified comm graph, achieve optimal coverage



Expected environment coverage

- let ϕ be distribution density function
- let f be a **performance/penalty function**

$f(\|q - p_i\|)$ is price for p_i to service q

- define **multi-center function**

$$\mathcal{H}_{\mathsf{C}}(p_1, \dots, p_n) = E_\phi\left[\min_i f(\|q - p_i\|)\right]$$
$$= \int_Q \min_i f(\|q - p_i\|)\phi(q)dq = \sum_i \int_{V_i} f(\|q - p_i\|)\phi(q)dq$$

S2: Distributed gradient result

For a general non-decreasing $f\colon \mathbb{R}_{\geq 0} \to \mathbb{R}$
piecewise differentiable with finite-jump discontinuities at $R_1 < \dots < R_m$

Thm:

$$\frac{\partial \mathcal{H}_{\mathsf{C}}}{\partial p_i}(p_1, \dots, p_n) = \int_{V_i} \frac{\partial}{\partial p_i} f(\|q - p_i\|)\phi(q)dq$$
$$+ \sum_{\alpha=1}^{m} \Delta f_\alpha(R_\alpha)\Big(\sum_{k=1}^{M_i(2R_\alpha)} \int_{\mathrm{arc}_{i,k}(2R_\alpha)} n_{B_{R_\alpha}(p_i)} d\phi \Big)$$

$=$ **integral over V_i** $+$ **integral along arcs inside V_i**

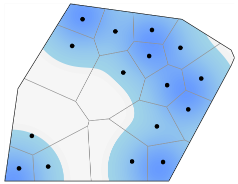
Gradient depends on information contained in V_i

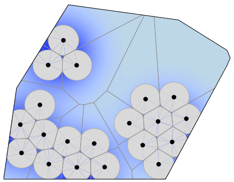
S2: Dispersion laws for deployment

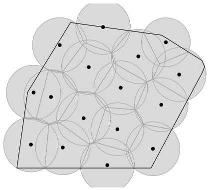
Dispersion laws

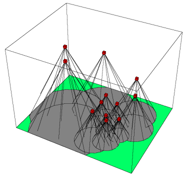
At each comm round:

1: acquire neighbors' positions
2: compute own dominance region
3: move towards incenter / circumcenter / centroid of own dominance region



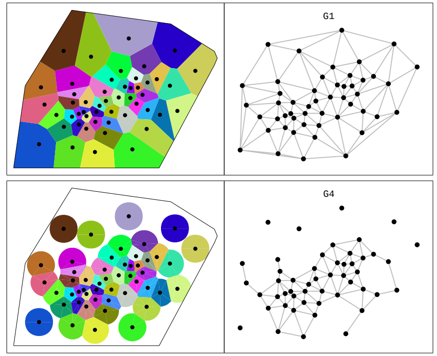






S2: On Voronoi and limited Voronoi partitions

Problem: $\frac{\partial \mathcal{H}_C}{\partial p_i}$ is distributed over Delaunay graph, but not disk graph
Solution: modify function so that its gradient is distributed over disk graph



S2: Truncation

problem $\partial\mathcal{H}_C$ distributed over Delaunay graph, but comm. is r-disk graph

approach truncate $f_{\frac{r}{2}}(x) = f(x)\ 1_{[0,\frac{r}{2})}(x) + (\sup_Q f)\cdot 1_{[\frac{r}{2},+\infty)}(x)$,

$$\mathcal{H}_{\frac{r}{2}}(p_1,\dots,p_n) = E_\phi\left[\min_i f_{\frac{r}{2}}(\|q-p_i\|)\right]$$

Result 1: Gradient of $\mathcal{H}_{\frac{r}{2}}$ is distributed over limited Delaunay

$$\frac{\partial \mathcal{H}_{\frac{r}{2}}}{\partial p_i} = \text{integral over } V_i \cap B_{\frac{r}{2}}(p_i) + \text{integral along arcs inside } V_i \cap B_{\frac{r}{2}}(p_i)$$

Result 2: $\mathcal{H}_C$ constant-factor approximation

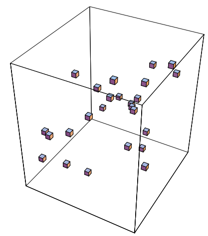
$$\beta\,\mathcal{H}_{\frac{r}{2}}(P) \le \mathcal{H}_C(P) \le \mathcal{H}_{\frac{r}{2}}(P)\,, \quad \beta = \left(\frac{r}{2\,\mathrm{diam}(Q)}\right)^2$$

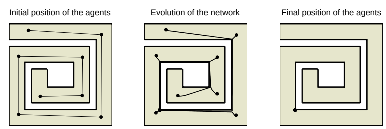
Scenario 3: aggregation laws for rendezous

Aggregation laws

At each comm round:

1: acquire neighbors' positions
2: compute connectivity constraint set
3: move towards circumcenter of neighbors (while remaining connected)





Task: rendezvous with connectivity constraint

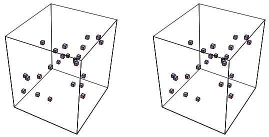
S3: Example complexity analysis

(i) first-order agents with disk graph, for $d = 1$,

$$\mathrm{TC}(\mathcal{T}_{\text{rendezvous}}, \mathcal{CC}_{\text{circumcenter}}) \in \Theta(N)$$

(ii) first-order agents with limited Delaunay graph, for $d = 1$,

$$\mathrm{TC}(\mathcal{T}_{(r\epsilon)\text{-rendezvous}}, \mathcal{CC}_{\text{circumcenter}}) \in \Theta(N^2 \log(N\epsilon^{-1}))$$



S3: Example proof technique

For $N \geq 2$ and $a, b, c \in \mathbb{R}$, define the $N \times N$ Toeplitz matrices

$$\mathrm{Trid}_N(a,b,c) = \begin{bmatrix} b & c & 0 & \dots & 0 \\ a & b & c & \dots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \dots & a & b & c \\ 0 & \dots & 0 & a & b \end{bmatrix}$$

$$\mathrm{Circ}_N(a,b,c) = \mathrm{Trid}_N(a,b,c) + \begin{bmatrix} 0 & \dots & \dots & 0 & a \\ 0 & \dots & \dots & 0 & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & \dots & 0 & 0 \\ c & 0 & \dots & 0 & 0 \end{bmatrix}$$

To be studied for interesting a, b, c as $N \to +\infty$

S3: Tridiagonal Toeplitz and circulant systems

Let $N \geq 2$, $\epsilon \in]0,1[$, and $a, b, c \in \mathbb{R}$. Let $x, y \colon \mathbb{N}_0 \to \mathbb{R}^N$ solve:

$$\begin{aligned} x(\ell+1) &= \mathrm{Trid}_N(a,b,c)\, x(\ell), & x(0) &= x_0, \\ y(\ell+1) &= \mathrm{Circ}_N(a,b,c)\, y(\ell), & y(0) &= y_0. \end{aligned}$$

(i) if $a = c \neq 0$ and $|b| + 2|a| = 1$, then $\lim_{\ell \to +\infty} x(\ell) = \mathbf{0}$, and the maximum time required for $\|x(\ell)\|_2 \leq \epsilon \|x_0\|_2$ is $\Theta\big(N^2 \log \epsilon^{-1}\big)$;

(ii) if $a \neq 0$, $c = 0$ and $0 < |b| < 1$, then $\lim_{\ell \to +\infty} x(\ell) = \mathbf{0}$, and the maximum time required for $\|x(\ell)\|_2 \leq \epsilon \|x_0\|_2$ is $O\big(N \log N + \log \epsilon^{-1}\big)$;

(iii) if $a \geq 0$, $c \geq 0$, $b > 0$, and $a + b + c = 1$, then $\lim_{\ell \to +\infty} y(\ell) = y_{\text{ave}} \mathbf{1}$, where $y_{\text{ave}} = \frac{1}{N} \mathbf{1}^T y_0$, and the maximum time required for $\|y(\ell) - y_{\text{ave}} \mathbf{1}\|_2 \leq \epsilon \|y_0 - y_{\text{ave}} \mathbf{1}\|_2$ is $\Theta\big(N^2 \log \epsilon^{-1}\big)$.

Summary: Coordination as Emerging Discipline

- **network modeling**

  network, ctrl+comm algorithm, task, complexity

- **coordination algorithm**

  optimal deployment, rendezvous, vehicle routing
  scalable, adaptive, asynchronous, agent arrival/departure

- **Systematic algorithm design**

  (i) geometric structures
  (ii) meaningful aggregate cost functions
  (iii) class of (gradient) algorithms local, distributed
  (iv) distributed information processing
  (v) stability theory for networked hybrid systems

Open problems

(i) more general complexity analysis: communication/control/time
(ii) algorithms for asynchronous networks with agent arrival/departures
(iii) robotic networks over random geometric graphs (multipath, fading)
(iv) general pattern formation problem
(v) $\lfloor n/3 \rfloor$ algorithm for visibility-based deployment
(vi) integrated motion coordination and sensor/estimation tasks
(vii) connections with biological networks