Motion Coordination for Multi-Agent Networks

Workshop on Network Embedded Sensing and Control
University of Notre Dame, IEEE Control Systems Society
October 17-18, 2005

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, Ketan Savla
Jorge Cortés (UCSC), Emilio Frazzoli (UCLA), Sonia Martínez (UCSD)
Department of Defense and National Science Foundation

Multi-agent networks

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

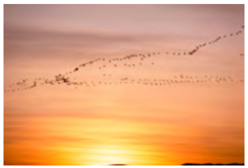
Individual members in the group can

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

Example networks from biology and engineering

Biological populations and swarms


Wildebeest herd in the Serengeti


Geese flying in formation


Atlantis aquarium, CDC Conference 2004

Multi-vehicle and sensor networks
embedded systems, distributed robotics

Distributed information systems, large-scale complex systems
intelligent buildings, stock market, self-managed air-traffic systems

Broad challenge

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

Problem	lack of understanding of how to assemble and coordinate individual devices into a coherent whole
Distributed feedback	rather than "centralized computation for known and static environment"
Approach	integration of control, comm, sensing, computing

Outline

Today's Objective: Systematic methodologies
to model, analyze and design multi-agent networks

Part I : Network Models
multi-agent network: motion/communication, tasks, complexity

Part II: Analysis and Design – Scenarios:
deployment, rendezvous, vehicle routing, connectivity maintenance

Part I: Synchronous robotic network

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

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

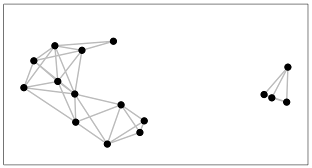
(ii) $\mathcal{A} = \{A_i\}_{i \in I}$, with $A_i = (X, U, X_0, f)$ is a set of identical control systems; **set of physical agents**

(iii) $E_{\text{cmm}} : X^N \to$ subsets of $I \times I$; **communication edge map**

Example networks

First-order agents with disk graph

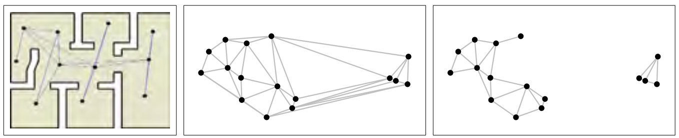
- agents locations are $x_1, \dots, x_N \in \mathbb{R}^d$
- first-order dynamics $\dot{x}_i(t) = u_i(t)$
- communication graph is r-disk graph



(i) **First-order agents with visibility graph**

(ii) **First-order agents with Delaunay graph**

(iii) **First-order agents with r-limited-Delaunay graph**



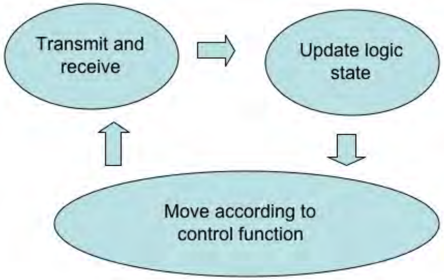
Control and communication law

A **control and communication law** $\mathcal{CC}$ for $\mathcal{S}$

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

Synchronous evolution

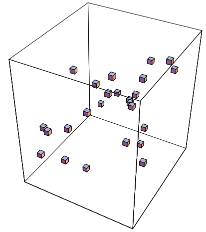
Execution cycle = discrete-time comm + continuous time motion

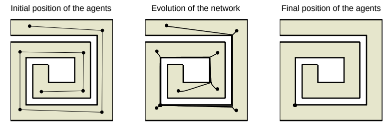


Example ctrl+comm laws (i)

Aggregation laws

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

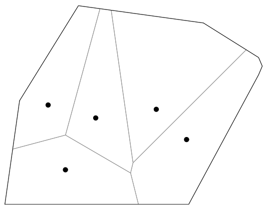


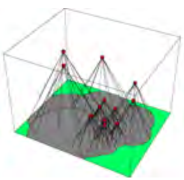


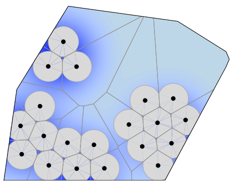
Example ctrl+comm laws (ii)

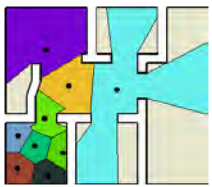
Dispersion laws

```
At each comm round:
 1: acquire neighbors' positions
 2: compute own dominance region
 3: move towards incenter /
    circumcenter / centroid of own
    dominance region
```









Coordination task

- **Coordination task** is $(\mathcal{W}, \mathcal{T})$ where $\mathcal{T}\colon X^N \times \mathcal{W}^N \to \{\texttt{true}, \texttt{false}\}$
- $\mathcal{CC}$ with logic vars W is **compatible** with $(\mathcal{W}, \mathcal{T})$ if $\mathcal{W} \subset W$
- $\mathcal{CC}$ **achieves** $\mathcal{T} = (\mathcal{W}, \mathcal{T})$ if all evolutions $t \mapsto (x(t), w(t))$ satisfy $\mathcal{T}(x(t), w(t)) = \texttt{true}$ for all t sufficiently large

Example tasks:

Motion: deploy, gather, flock, reach pattern

Logic-based: achieve consensus, synchronize, form a team

Sensor-based: search, estimate, identify, track, map

Cost, complexity and scalability

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

(i) **time complexity to achieve $\mathcal{T}$ with $\mathcal{CC}$**

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

(ii) **time complexity of $\mathcal{T}$**

$$\mathrm{TC}(\mathcal{T}) = \inf \{\mathrm{TC}(\mathcal{T}, \mathcal{CC}) \mid \mathcal{CC} \text{ achieves } \mathcal{T}\}$$

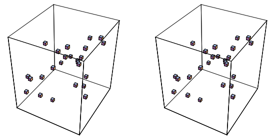
Example complexity analysis

(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, for $d = 1$,

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



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

$$\mathrm{TC}(\mathcal{T}_{(r\epsilon)\text{-deployment}}, \mathcal{CC}_{\text{centroid}}) \in O(N^3 \log(N\epsilon^{-1}))$$

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:

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

(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 of Part I: Models for Robotic Networks

(i) ad-hoc communication topology

(ii) distributed algorithms over given information flow

(iii) cooperative control

(iv) **todo:** quantization, asynchronism, delays

Key outcomes

(i) multi-agent "lingua franca" for control/robotics/CS/networking

(ii) need a meaningful+tractable model to
define, characterize and compare algorithms

(iii) beatiful richness

Part II: Analysis and Design

Scenarios examples of networks, tasks, ctrl+comm laws

(i) deployment
(ii) rendezvous
(iii) vehicle routing

Analysis tools

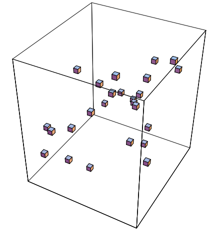
(i) stability theory: nonlinear, nonsmooth and hybrid
(ii) geometric graphs and geometric optimization
(iii) algebraic graph theory

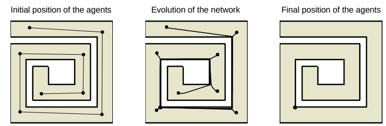
Scenario 1: aggregation laws for rendezous

Aggregation laws

At each comm round:

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



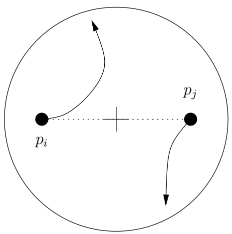


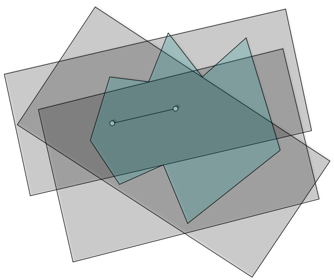
Task: rendezvous with connectivity constraint

Scenario 1: aggregation laws for rendezous, cont'd

Pair-wise motion constraint set for connectivity maintenance

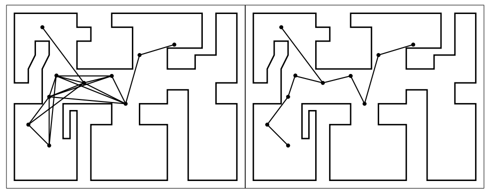
- for every pair of agents, constrain motion to maintain connectivity
- distributed computation of maximal set
- set is continuous function of agents' positions



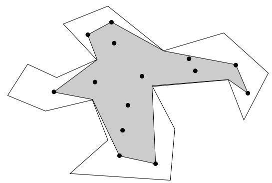


Scenario 1: aggregation laws for rendezous, cont'd

Reducing number of constraints



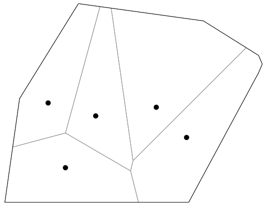
Lyapunov function: perimeter of minimum perimeter polygon

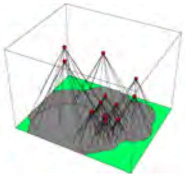


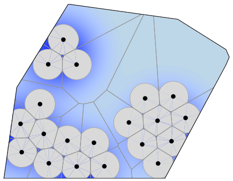
Scenarios: dispertion laws for deployment

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
```



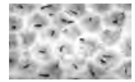




Scenarios: optimal deployment

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, Hexagonal territories. Anim. Behav. '74

(ii) what if each vehicle moves toward center of mass of own Voronoi cell?

(iii) what if each vehicle moves away from closest vehicle?

DESIGN of performance metric

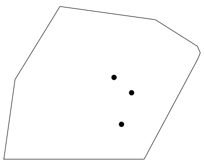
(iv) how to cover a region with n minimum radius overlapping disks?

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

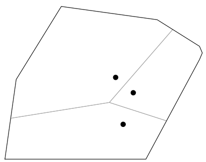
(vi) where to place mailboxes in a city / cache servers on the internet?

Scenario 2: "simple" emerging behaviors

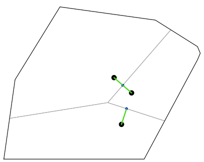
consider n points in Q



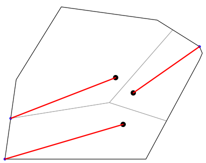
draw Voronoi partition



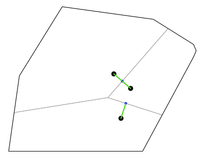
identify closest point

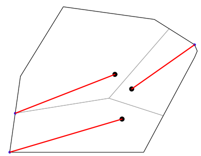


identify furthest point

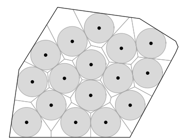


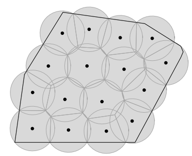
Scenario 2: "simple" emerging behaviors, cont'd





Basic greedy behaviors
"move away from closest"
"move towards furthest"





Conjectures: critical points or periodic trajectories? convergence? optimize? local minima? equidistant?

Scenario 2: "simple" emerging behaviors, end

Thm 1: Semidefinite Lyapunov functions are LL&R

$$\mathcal{H}_{\text{SP}}(p_1,\dots,p_n) = \text{smallest radius} = \min_{i\in\{1,\dots,n\}} \left\{\tfrac{1}{2}\|p_i - p_j\|, \operatorname{dist}(p_i,\partial Q)\right\}$$

$$\mathcal{H}_{\text{DC}}(p_1,\dots,p_n) = \text{largest radius} = \max_{i\in\{1,\dots,n\}} \max_{q\in Q} \left\{\min_i \|q - p_i\|\right\}$$

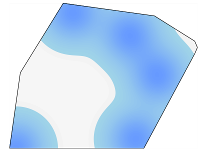
Lem 2: At fixed partitions V_i, in finite time $p_i \to \mathrm{IC}(V_i)$ or, resp., $p_i \to \mathrm{CC}(V_i)$

Thm 3: Agent i converges if i is **active**

Conjecture: All agents converge to in- or circum-centers

Scenario 3: 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**

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

- define **multi-center function**

$$\mathcal{H}_{\text{C}}(p_1,\dots,p_n) = E_\phi\left[\min_i f(\|q - p_i\|)\right]$$

Scenario 3: distributed gradient result

For a general non-decreasing $f : \overline{\mathbb{R}}_+ \to \mathbb{R}$
piecewise differentiable with finite-jump discontinuities at $R_1 < \cdots < R_m$

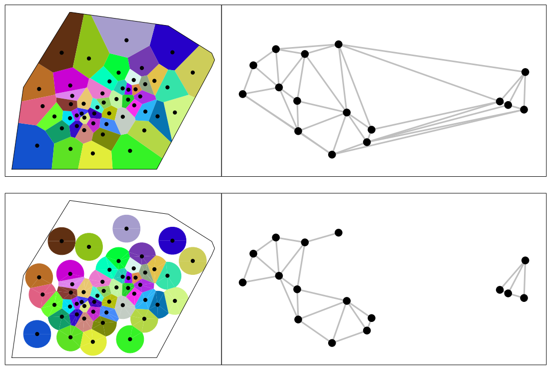
Thm: $\mathcal{H}_{\text{C}}$ satisfies on $Q^n \setminus \{(p_1,\dots,p_n) \in (\mathbb{R}^2)^n \mid p_i = p_j \text{ with } i \neq j\}$

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

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

Gradient depens on information contained in V_i

On Voronoi and limited-Voronoi partitions



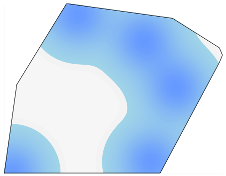
$\partial\mathcal{H}_{\text{C}}$ is spatially distributed **over Delaunay graph**, but not **disk graph**

Scenario 3.a: min-expected-distance deployment

Assume: Delauney comm graph (i.e., Voronoi partition of full environment)

Scenario 3.a —expected value performance measure
(unlimited-range sensor or communication radius)

given distribution density function ϕ



minimize $\mathcal{H}_{\mathsf{C}}(p_1, \dots, p_n) = E_\phi \left[\min_i \|q - p_i\|^2 \right]$

Scenario 3.a: coverage algorithm

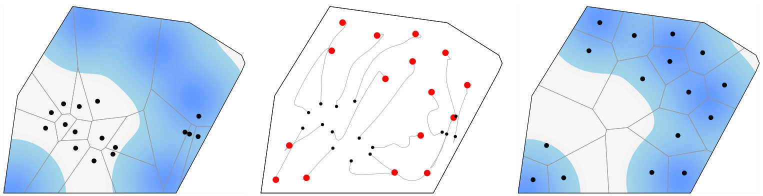
Name: Coverage behavior
Goal: distributed optimal agent deployment
Requires: (i) own Voronoi cell computation
(ii) centroid computation

For all i, agent i synchronously performs:

1: determine own Voronoi cell V_i
2: determine centroid C_{V_i} of V_i
3: move towards centroid (e.g. $u_i = \mathrm{sat}(C_{V_i} - p_i)$)

Scenario 3.a: simulation

run: 16 agents, density ϕ is sum of 4 Gaussians, 1st order dynamics



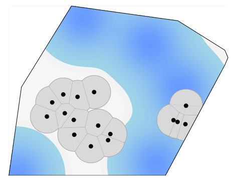
initial configuration — gradient descent — final configuration

Scenario 3.b: max-area deployment

Assume: disk comm graph with r

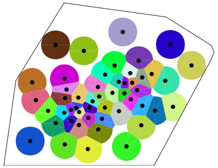
Scenario 3.b —area covered by balls $r/2$

given distribution density function ϕ



maximize $\mathrm{area}_\phi(\cup_{i=1}^n B_{\frac{r}{2}}(p_i)) = \int_Q \left(\max_i 1_{B_{\frac{r}{2}}(p_i)}(q) \right) \phi(q) dq$

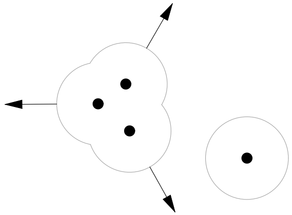
Scenario 3.b: partition of covered area



Partition of $\cup_i B_{r/2}(p_i)$:
$$\{V_1 \cap B_{r/2}(p_1), \dots, V_n \cap B_{r/2}(p_n)\}.$$

Limited Delauney neighbors
those with adjacent cells

For constant density $\phi = 1$,



$$\int_{\mathrm{arc}(r)} n_{B_{\frac{r}{2}}(p)}\, \phi$$

Scenario 3.b: coverage algorithm

Name: Coverage behavior
Goal: distributed optimal agent deployment
Requires: (i) own cell computation
(ii) weighted normal computation

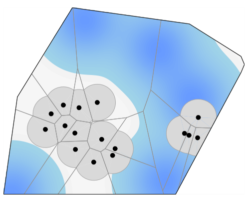
For all i, agent i synchronously performs:

1: determines own cell $V_i \cap B_{\frac{r}{2}}(p_i)$
2: determines weighted normal $\int_{\mathrm{arc}(r)} n_{B_{\frac{r}{2}}(p)}\, \phi$
3: moves in the direction of weighted normal

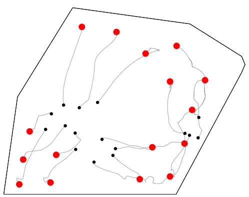
Caveat: convergence only to local maximum of $\mathrm{area}_\phi(\cup_{i=1}^n B_{\frac{r}{2}}(p_i))$

Scenario 3.b: simulation

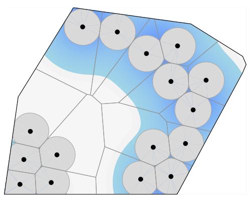
run: 20 agents, density ϕ is sum of 4 Gaussians, 1st order dynamics



initial configuration



gradient descent



final configuration

Scenario 3.c: truncation

Thm 1: $\mathcal{H}_{\mathsf{C}}$ **constant-factor approximation** by

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

for **truncated** $f_{\frac{r}{2}}(x) = f(x)\; 1_{[0,\frac{r}{2})}(x) + (\sup_Q f) \cdot 1_{[\frac{r}{2},+\infty)}(x)$,

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

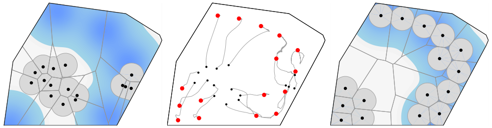
Thm 2 Gradient of $\mathcal{H}_{\frac{r}{2}}$ is **spatially distributed over r-limited Delaunay graph**

$$\frac{\partial \mathcal{H}_{\frac{r}{2}}}{\partial p_i} = 2M_{V_i(P)\cap B_{\frac{r}{2}}(p_i)}(C_{V_i(P)\cap B_{\frac{r}{2}}(p_i)} - p_i) - \left(\left(\frac{r}{2}\right)^2 - \mathrm{diam}(Q)^2\right) \sum_{k=1}^{\mathcal{M}_i(r)} \int_{\mathrm{arc}_{i,k}(r)} n_{B_{\frac{r}{2}}(p_i)}\, \phi$$

Scenario 3.c: Simulations

Limited range

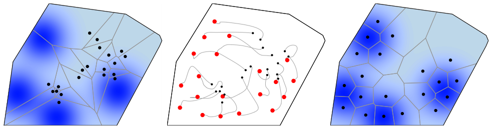
run #1: 16 agents, density ϕ is sum of 4 Gaussians, time invariant, 1st order dynamics



initial configuration — gradient descent of $\mathcal{H}_{\frac{r}{2}}$ — final configuration

Unlimited range

run #2: 16 agents, density ϕ is sum of 4 Gaussians, time invariant, 1st order dynamics



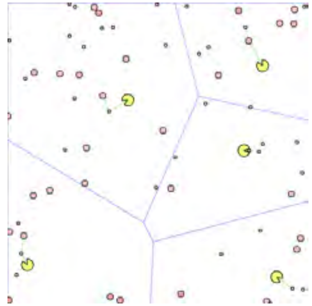
initial configuration — gradient descent of $\mathcal{H}_{\mathrm{C}}$ — final configuration

Scenario 4: Vehicle Routing

Objective: Given agents $(p_1, \ldots, p_n)$ moving in environment Q service targets in environment

Model:

- targets arise randomly in space/time
- vehicle know of targets arrivals



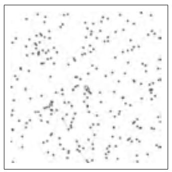
Scenario 4 —min expected waiting time

Scenario 4: receding-horizon TSP algorithm, I

Name: (Single Vehicle) Receding-horizon TSP

For $\eta \in (0, 1]$, single agent performs:

1: while no targets, dispersion/coverage algorithm
2: while targets waiting,
  (i) compute optimal TSP tour through all targets
  (ii) service the η-fraction of tour with maximal number of targets



Asymptotically optimal in light and high traffic

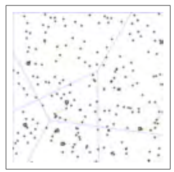
Scenario 4: receding-horizon TSP algorithm, II

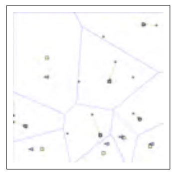
Name: Receding-horizon TSP

For $\eta \in (0, 1]$, agent i performs:

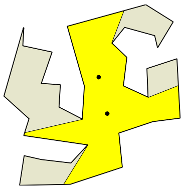
1: compute own Voronoi cell V_i
2: apply Single-Vehicle RH-TSP policy on V_i

Asymptotically optimal in light and high traffic (simulations only)



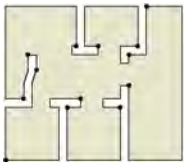


Scenario 5: Visibility-based deployment



Gradient-based approach

Partition-based approach



tree-navigation-based algorithm

Summary

- first provably correct algorithm for distributed art gallery problem
- general results on nonsmooth analysis and control design
- 2D and 3D version ongoing

Emerging Motion Coordination Discipline

(i) **network modeling**

network, ctrl+comm algorithm, task, complexiy

coordination algorithm

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

(ii) **Systematic algorithm design**

- meaningful aggregate cost functions
- class of (gradient) algorithms local, distributed
- geometric graphs
- stability theory for networked hybrid systems