Dynamic Vehicle Routing for Robotic Networks: Models, Fundamental Limitations and Algorithms

Francesco Bullo

Center for Control,
Dynamical Systems & Computation
University of California at Santa Barbara
http://motion.me.ucsb.edu

U.S. Army Research Laboratory
Adelphi Laboratory Center, 16 April 2010

Acknowledgements

- Funded in large part by ARO MURI "Swarms" W911NF-05-1-0219
- Funded in part by ONR award N00014-07-1-0721
- Funded in part by Institute for Collaborative Biotechnologies, ARO award DAAD19-03-D-0004

Collaborators on Robotic Coordination

Ruggero Carli (UCSB), Jorge Cortés (UCSD), Joey W. Durham (UCSB), Paolo Frasca (Roma), Anurag Ganguli (UtopiaCompression), Sonia Martínez (UCSD), Karl Obermeyer (UCSB), Stephen L. Smith (MIT), and Sara Susca (Honeywell)

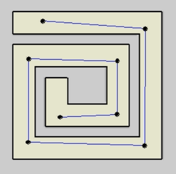
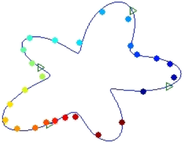
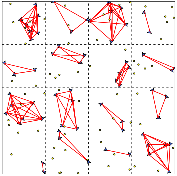
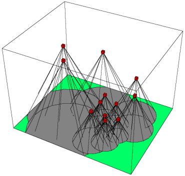
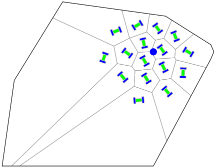
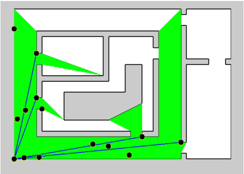
Collaborators on Dynamic Vehicle Routing

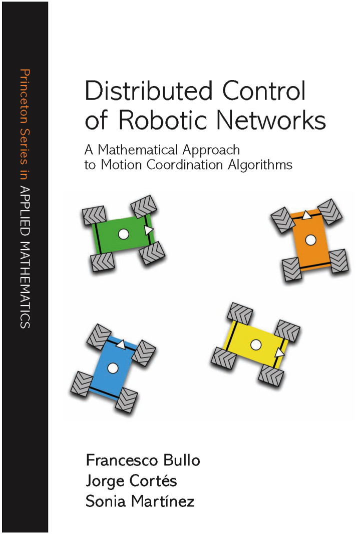
Shaunak D. Bopardikar (UCSB), John J. Enright (MIT), Emilio Frazzoli (MIT), João P. Hespanha (UCSB) Marco Pavone (MIT/JPL), Ketan Savla (MIT), and Stephen L. Smith (MIT)

Today's Outline

1. Robotic Coordination: Brief Review
2. Dynamic Vehicle Routing (DVR)
3. Extensions
   - DVR for Nonholonomic Vehicles
   - DVR for Moving Demands
   - DVR with heterogeneous demands requiring teams
   - DVR with priority levels
4. DVR Load Balancing via Territory Partitioning
5. Conclusions

Robotic coordination



1. intro to distributed algorithms (graph theory, synchronous networks, and averaging algos)
2. geometric models and geometric optimization problems
3. model for robotic, relative sensing networks, and complexity
4. algorithms for rendezvous, deployment, boundary estimation

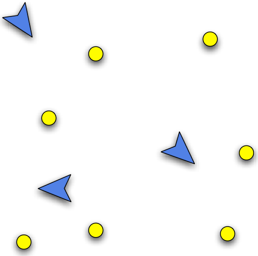
Status: Published by Princeton Univ Press. Manuscript and slides freely available at http://coordinationbook.info

Today's Outline

Prototypical Dynamic Vehicle Routing Problem

Given:

- a group of vehicles, and
- a set of service demands



Objective:
provide service in minimum time
service = take a picture at location

Vehicle routing (All info known ahead of time, Dantzig '59)

Determine a set of paths that allow vehicles to service the demands

Dynamic vehicle routing (New info in real time, Psaraftis '88)

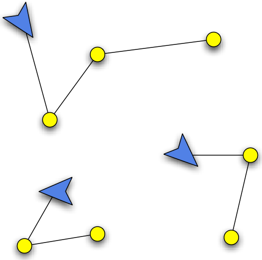
- New demands arise in real-time
- Existing demands evolve over time

Prototypical Dynamic Vehicle Routing Problem

Given:

- a group of vehicles, and
- a set of service demands

Objective:
provide service in minimum time
service = take a picture at location



Vehicle routing (All info known ahead of time, Dantzig '59)

Determine a set of paths that allow vehicles to service the demands

Dynamic vehicle routing (New info in real time, Psaraftis '88)

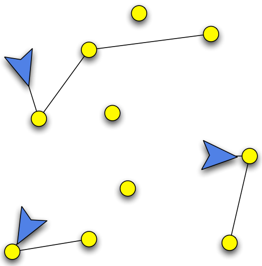
- New demands arise in real-time
- Existing demands evolve over time

Prototypical Dynamic Vehicle Routing Problem

Given:

- a group of vehicles, and
- a set of service demands

Objective:
provide service in minimum time
service = take a picture at location



Vehicle routing (All info known ahead of time, Dantzig '59)

Determine a set of paths that allow vehicles to service the demands

Dynamic vehicle routing (New info in real time, Psaraftis '88)

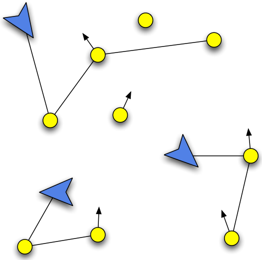
- New demands arise in real-time
- Existing demands evolve over time

Prototypical Dynamic Vehicle Routing Problem

Given:

- a group of vehicles, and
- a set of service demands

Objective:
provide service in minimum time
service = take a picture at location



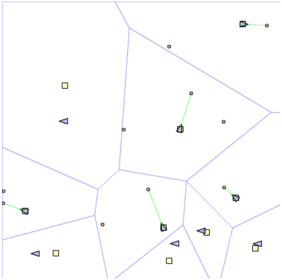
Vehicle routing (All info known ahead of time, Dantzig '59)

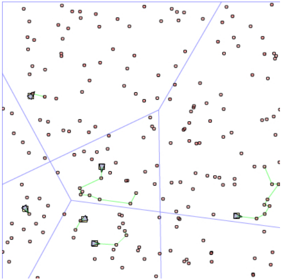
Determine a set of paths that allow vehicles to service the demands

Dynamic vehicle routing (New info in real time, Psaraftis '88)

- New demands arise in real-time
- Existing demands evolve over time

Light and heavy load regimes





Literature review on DVR

- Shortest path through randomly-generated and worst-case points (Beardwood, Halton and Hammersly, 1959 — Steele, 1990)
- Traveling salesman problem solvers (Lin, Kernighan, 1973)
- DVR formulation on a graph (Psaraftis, 1988)
- DVR on Euclidean plane (Bertsimas and Van Ryzin, 1990–1993)
- Unified receding-horizon policy (Papastavrou, 1996)

Recent developments in DVR for robotic networks:

- Adaptation and decentralization (Pavone, Frazzoli, FB: TAC, in press)
- Nonholonomic / Dubins UAVs (Savla, Frazzoli, FB: TAC 2008)
- Pickup delivery tasks (Waisanen, Shah, and Dahleh: TAC 2008)
- Heterogeneous vehicles and team forming (Smith and Bullo: SCL 2009)
- Distinct-priority demands (Smith, Pavone, FB, Frazzoli: SICON, in press)
- Moving demands (Bopardikar, Smith, Hespanha, FB: TAC, in press)

Algo #1: Receding-Horizon Shortest-Path policy

Receding-Horizon Shortest-Path (RH-SP)

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

```
1: while no customers, move to center
2: while customers waiting
```

1. compute shortest path through current targets
2. service η-fraction of path

Algo #1: Receding-Horizon Shortest-Path policy

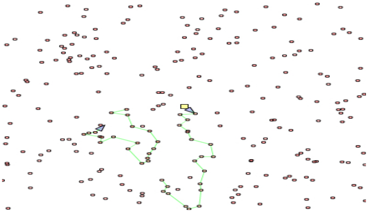
Receding-Horizon Shortest-Path (RH-SP)

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

1: while no customers, move to center

2: while customers waiting

1. compute shortest path through current targets
2. service η-fraction of path



M. Pavone, E. Frazzoli, and F. Bullo. Distributed and adaptive algorithms for vehicle routing in a stochastic and dynamic environment. *IEEE Transactions on Automatic Control*, August 2009. (Submitted, Apr 2009) to appear

RH-SP analysis

Implementation:

- NP-hard computation, but effective heuristics

Stability:

1. queue is stable if **service time** $<$ **interarrival time**
2. **service time** $= \dfrac{\textbf{length shortest path}(n)}{n}$ $\quad (n = \#\text{ customers})$
3. queue is stable if (length of shortest path(n)) = **sublinear f(n)**

RH-SP analysis

Implementation:

- NP-hard computation, but effective heuristics

Stability:

1. queue is stable if **service time** $<$ **interarrival time**
2. **service time** $= \dfrac{\textbf{length shortest path}(n)}{n}$ $(n = \#\ \text{customers})$
3. queue is stable if (length of shortest path(n)) $=$ **sublinear f(n)**

RH-SP analysis

Implementation:

- NP-hard computation, but effective heuristics

Stability:

1. queue is stable if **service time $<$ interarrival time**
2. **service time** $= \dfrac{\textbf{length shortest path}(n)}{n}$ $(n = \#$ customers$)$
3. queue is stable if (length of shortest path(n)) = **sublinear f(n)**

RH-SP analysis

Implementation:

- NP-hard computation, but effective heuristics

Stability:

1. queue is stable if **service time** $<$ **interarrival time**
2. **service time** $= \dfrac{\textbf{length shortest path}(n)}{n}$ $\qquad (n = \#\ \text{customers})$
3. queue is stable if (length of shortest path(n)) $=$ **sublinear f(n)**

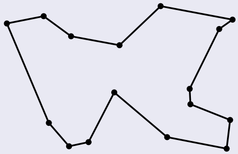
Implementation:

- NP-hard computation, but effective heuristics

Stability:

1. queue is stable if **service time $<$ interarrival time**
2. **service time** $= \dfrac{\textbf{length shortest path}(n)}{n}$ $\qquad (n = \#\ \text{customers})$
3. queue is stable if (length of shortest path(n)) $=$ **sublinear f(n)**

Combinatorics in Euclidean space (Steel '90)



Worst-case and expected bounds

$$\text{length shortest path}(n) \le \beta_{\text{worst}} \sqrt{n}$$

$$\lim_{n \to +\infty} \text{length shortest path}(n) = \beta_{\text{expected}} \sqrt{n}$$

RH-SP analysis: continued

Adaptation: the policy does not require knowledge of

1. vehicle velocity v, environment Q
2. arrival rate λ and spatial density function f
3. expected on-site service $\bar{s}$

Performance:

1. in light load, delay is optimal
2. in heavy load, delay is within a multiplicative factor from optimal
3. multiplicative factor depends upon f and is conjectured to equal 2

no known adaptive algo with better performance
very little known outside of asymptotic regimes

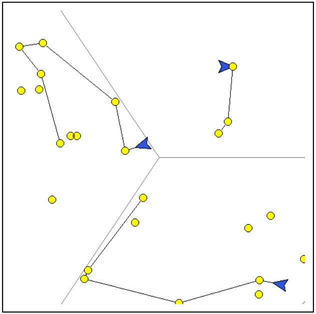
Algo #2: Load balancing via territory partitioning

RH-SP + Partitioning

Each agent i:

1: computes own cell v_i in optimal partition

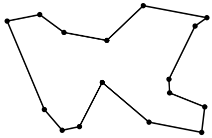
2: applies RH-SP policy on v_i



Today's Outline

Euclidean TSP and Dubins TSP

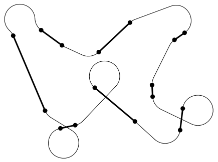
Euclidean TSP (ETSP)



- NP-hard
- effective heuristics available
- length(ETSP) $\in O(\sqrt{n})$

Dubins TSP (DTSP)

Given a set of points find the shortest tour with bounded curvature



- not a finite dimensional problem
- no prior algorithms or results (as of 2006)
- length(DTSP) sub-linear in n ?

K. Savla, E. Frazzoli, and F. Bullo. Traveling Salesperson Problems for the Dubins vehicle. *IEEE Transactions on Automatic Control*, 53(6):1378–1391, 2008

Stochastic DTSP

Problem Statement Given a set of n *independently and uniformly distributed* points, design polynomial-time algorithm with smallest *expected* DTSP tour length

Theorem: For n iid uniformly distributed points:

$$\mathbb{E}[\text{length of DTSP}(n)] \sim n^{2/3}$$

Lower bound proof based on "area of reachable set"

1. area of reachable set in time t by Dubins with radius ρ is $O(t^3)$
2. expected number of points in area is $O(nt^3)$ (for n iid uniform targets)
3. expected distance to nearest target is $O(n^{-1/3})$
4. length of tour cannot be less than n times this distance

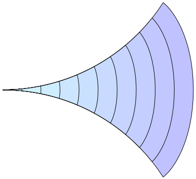
J. J. Enright and E. Frazzoli. UAV routing in a stochastic time-varying environment. In *IFAC World Congress*, Prague, Czech Republic, July 2005

Stochastic DTSP

Problem Statement Given a set of n *independently and uniformly distributed* points, design polynomial-time algorithm with smallest *expected* DTSP tour length

Theorem: For n iid uniformly distributed points:

$$\mathbb{E}[\text{length of DTSP}(n)] \sim n^{2/3}$$

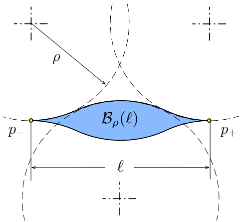


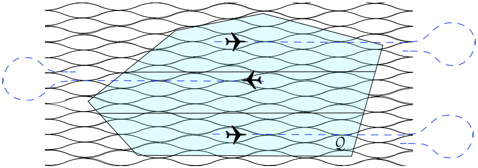
Lower bound proof based on "area of reachable set"

1. area of reachable set in time t by Dubins with radius ρ is $O(t^3)$
2. expected number of points in area is $O(nt^3)$ (for n iid uniform targets)
3. expected distance to nearest target is $O(n^{-1/3})$
4. length of tour cannot be less than n times this distance

J. J. Enright and E. Frazzoli. UAV routing in a stochastic time-varying environment. In *IFAC World Congress*, Prague, Czech Republic, July 2005

Constructive upper bound
based on environment tiling tuned to vehicle dynamics





Key properties of the bead

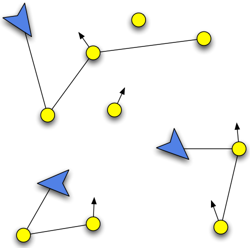
1. Beads tile the plane
2. Approaching and leaving a bead horizontally, Dubins can service a target

first analysis of joint combinatorics, dynamics and stochastic
extensions to STLC systems by Itani, Dahleh and Frazzoli
extensions to multi-vehicle Dubins

Today's Outline

1. Robotic Coordination: Brief Review
2. Dynamic Vehicle Routing (DVR)
3. Extensions
   - DVR for Nonholonomic Vehicles
   - DVR for Moving Demands
   - DVR with heterogeneous demands requiring teams
   - DVR with priority levels
4. DVR Load Balancing via Territory Partitioning
5. Conclusions

Dynamic vehicle routing for moving demands



Very little is know about moving demands:

1. no polynomial time algorithms for shortest path
2. no length estimates
3. no efficient DVR algorithms

S. D. Bopardikar, S. L. Smith, F. Bullo, and J. P. Hespanha. Dynamic vehicle routing for translating demands: Stability analysis and receding-horizon policies. *IEEE Transactions on Automatic Control*, 55(11), 2010. (Submitted, Mar 2009) to appear

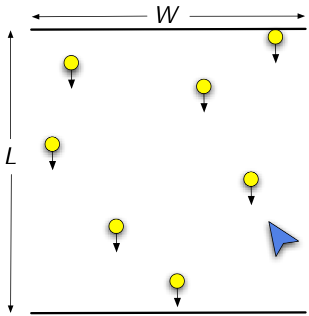
Translating demands: problem setup

Problem parameters:

- speed ratio v:

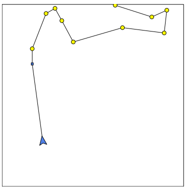
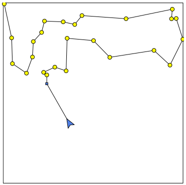
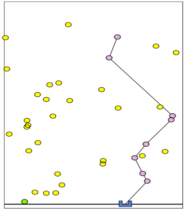
$$v = \frac{\text{demand speed}}{\text{vehicle speed}}$$

- arrival rate λ
- segment width W
- deadline distance L

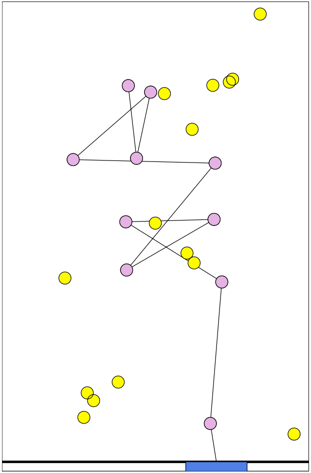


	$L = +\infty$ Stabilize queue	L is finite Maximize capture fraction
$v < 1$		
$v \geq 1$		

Translating demands: policies

	$L = +\infty$ Stabilize queue	L is finite Maximize capture fraction
$v < 1$		
$v \geq 1$	**Not possible for any $\lambda > 0$**	

Moving demands: more general scenarios



Relaxed assumptions:

- Non-Poisson
- Non-uniform
- Different speeds
- Different directions
- Finite capture radius

More general setup:

- Higher dimensions
- Advance information

S. L. Smith, S. D. Bopardikar, and F. Bullo. A dynamic boundary guarding problem with translating demands. In *IEEE Conf. on Decision and Control*, pages 8543–8548, Shanghai, China, December 2009

Today's Outline

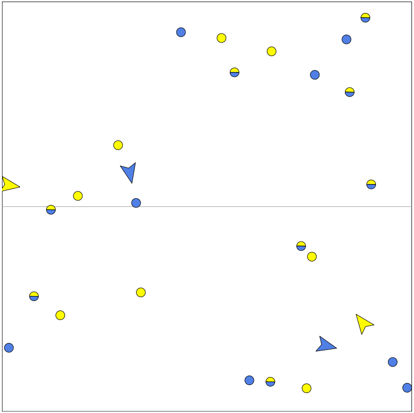
DVR with heterogeneous demands requiring teams

Problem setup:

- Heterogeneous vehicles
- Tasks require vehicle teams

Goal: Minimize task delay

Consider only unbiased policies:
Equal expected delay to all tasks



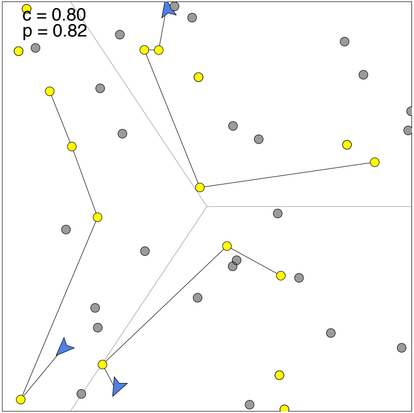
- Provably efficient policies in certain scenarios
- Very rich problem

S. L. Smith and F. Bullo. The dynamic team forming problem: Throughput and delay for unbiased policies. *Systems & Control Letters*, 58(10-11):709–715, 2009

Today's Outline

Problem setup:

- n vehicles
- Two classes of tasks α, β
  - α – high priority
  - β – low priority



Goal: minimize $cD_\alpha + (1-c)D_\beta$

$c \in (0,1)$ gives bias toward α

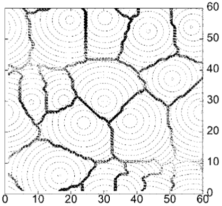
- Provably efficient policy
- Extends to m classes

S. L. Smith, M. Pavone, F. Bullo, and E. Frazzoli. Dynamic vehicle routing with priority classes of stochastic demands. *SIAM Journal on Control and Optimization*, 48(5):3224–3245, 2010

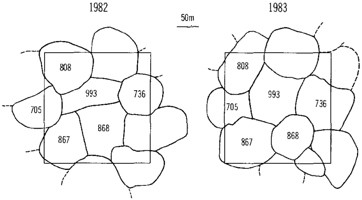
Today's Outline

Territory partitioning akin to *animal territory dynamics*

Tilapia mossambica, "Hexagonal Territories," Barlow et al, '74



Red harvester ants, "Optimization, Conflict, and Nonoverlapping Foraging Ranges," Adler et al, '03



Sage sparrows, "Territory dynamics in a sage sparrows population," Petersen et al '87

Optimal partitioning cost functions

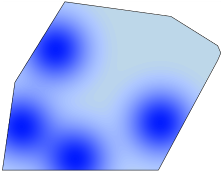
Expected wait time (light load problem)

$$H(p, v) = \int_{v_1} \|q - p_1\| dq + \cdots + \int_{v_n} \|q - p_n\| dq$$

- n robots at $p = \{p_1, \ldots, p_n\}$
- environment is partitioned into $v = \{v_1, \ldots, v_n\}$

$$H(p, v) = \sum_{i=1}^{n} \int_{v_i} f(\|q - p_i\|)\phi(q) dq$$

- $\phi : \mathbb{R}^2 \to \mathbb{R}_{\geq 0}$ density
- $f : \mathbb{R}_{\geq 0} \to \mathbb{R}$ penalty function



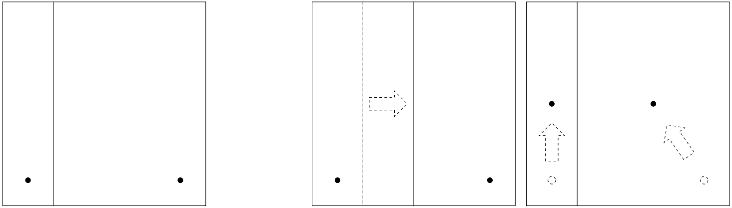
From optimality conditions to algorithms

$$H(p,v) = \sum_{i=1}^{n} \int_{v_i} f(\|q - p_i\|)\phi(q)dq$$

Theorem (Alternating Algorithm, Lloyd '57)

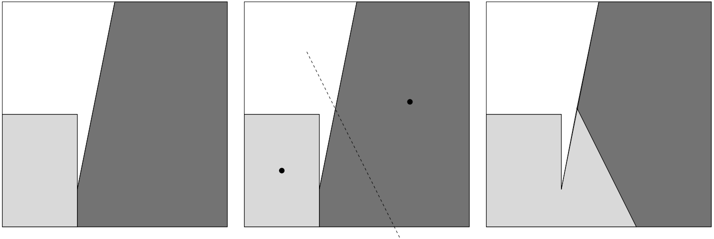
1. *at fixed positions, optimal partition is Voronoi*
2. *at fixed partition, optimal positions are "generalized centers"*
3. *alternate v-p optimization*

$\implies$ *local optimum = center Voronoi partition*



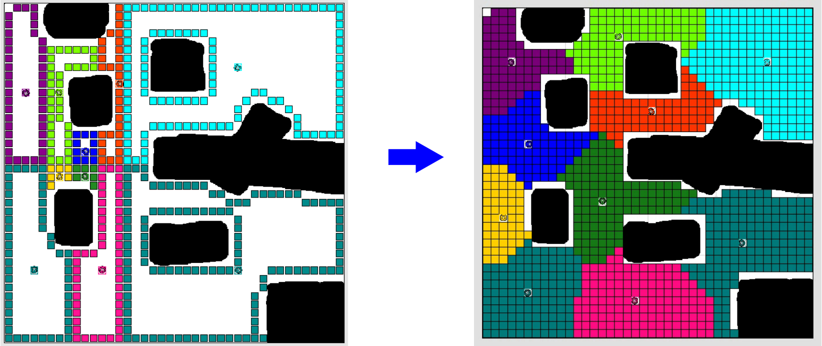
Gossip partitioning policy

1. Random communication between two regions
2. Compute two centers
3. Compute bisector of centers
4. Partition two regions by bisector



F. Bullo, R. Carli, and P. Frasca. Gossip coverage control for robotic networks: Dynamical systems on the the space of partitions. *SIAM Review*, January 2010. Submitted

Gossip partitioning policy: sample implementation



- Player/Stage platform
- realistic robot models in discretized environments
- integrated wireless network model & obstacle-avoidance planner

J. W. Durham, R. Carli, P. Frasca, and F. Bullo. Discrete partitioning and coverage control with gossip communication. In *ASME Dynamic Systems and Control Conference*, Hollywood, CA, October 2009

Gossip partitioning policy: analysis results

1. class of dynamical systems on space of partitions

   i.e., study evolution of the regions rather of the agents

2. convergence to centroidal Voronoi partitions (under mild conditions)

3. novel results in topology, analysis and geometry:
   1. **compactness** of space of finitely-convex partitions with respect to the symmetric difference metric
   2. **continuity** of various geometric maps (Voronoi as function of generators, centroid location as function of set, multicenter functions)
   3. **LaSalle convergence theorems** for dynamical systems on metric spaces with deterministic and stochastic switches

conjectures about topology of space of partitions

asymmetric gossip algorithms, akin to stigmergy

tolerance to failures, arrivals, and dynamic environments

Today's Outline

1. Robotic Coordination: Brief Review
2. Dynamic Vehicle Routing (DVR)
3. Extensions
   - DVR for Nonholonomic Vehicles
   - DVR for Moving Demands
   - DVR with heterogeneous demands requiring teams
   - DVR with priority levels
4. DVR Load Balancing via Territory Partitioning
5. Conclusions