Geometric Algorithms for Scheduling, Coordination and Motion Planning

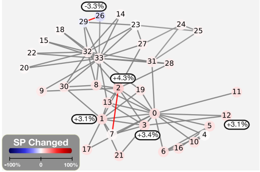
Jie Gao

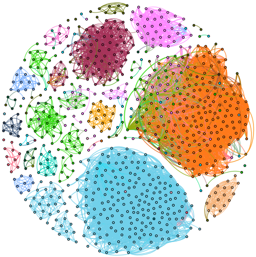
Stony Brook University

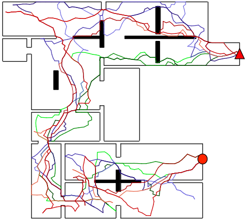
Northeastern Robotics Colloquium (NERC VII)
October 20th, 2018.

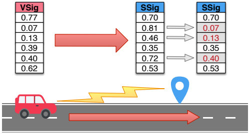
Research Interest and Projects

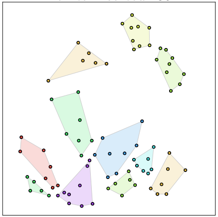
Algorithm, Computational Geometry, Wireless Networks, Social Networks, Trajectories, Privacy

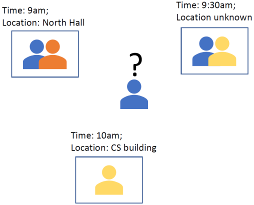












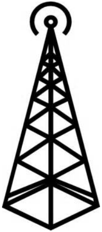
Resource Scheduling in Space Time Domain

Considerations:

- Efficiency – energy, storage, bandwidth.
- Performance – coverage, detection, connectivity.

Constraint dimensions:

- Spatial – visibility, proximity.
- Temporal – mobility.



Application Scenarios

How to schedule and allocate resources in spatial and temporal domains?

- Guarding: optimize resource usage, improve safety & security.

Application Scenarios

How to schedule and allocate resources in spatial and temporal domains?

- Guarding: optimize resource usage, improve safety & security.
- Mobile networks: design paths for mobile nodes to collect data from sensors with storage constraints.

Application Scenarios

How to schedule and allocate resources in spatial and temporal domains?

- Guarding: optimize resource usage, improve safety & security.
- Mobile networks: design paths for mobile nodes to collect data from sensors with storage constraints.
- Delivery: deliver packages to residents during their specified time windows.

Application Scenarios

How to schedule and allocate resources in spatial and temporal domains?

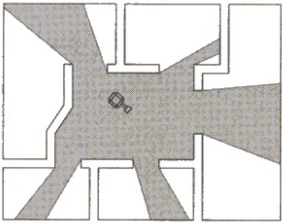
- Guarding: optimize resource usage, improve safety & security.
- Mobile networks: design paths for mobile nodes to collect data from sensors with storage constraints.
- Delivery: deliver packages to residents during their specified time windows.
- Sweeping multi-robots: collectively cover a terrain.

Outline

1. Collaborative monitoring and scheduling.
2. Path planning for mobile nodes.
3. Domain sweeping by multi-robots.

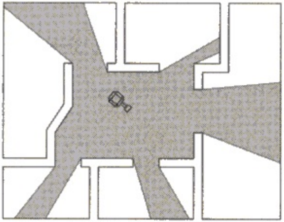
Coverage in a Building

Given n guarding nodes and m target nodes, the set of targets covered by guarding site g_i is $P(g_i)$, how to schedule the guarding nodes?



Coverage in a Building

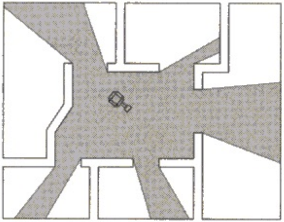
Given n guarding nodes and m target nodes, the set of targets covered by guarding site g_i is $P(g_i)$, how to schedule the guarding nodes?



- If guarding nodes are turned on all the time: Art Gallery Problem;

Coverage in a Building

Given n guarding nodes and m target nodes, the set of targets covered by guarding site g_i is $P(g_i)$, how to schedule the guarding nodes?



- If guarding nodes are turned on all the time: Art Gallery Problem;
- Insufficient guarding nodes: Duty Cycle Scheduling.

Coverage in a Building

Given n guarding nodes and m target nodes, the set of targets covered by guarding site g_i is $P(g_i)$, how to schedule the guarding nodes?

- Time is slotted.
- At each time slot at most k guarding nodes are turned on.
- Periodic schedule.

Coverage in a Building

Given n guarding nodes and m target nodes, the set of targets covered by guarding site g_i is $P(g_i)$, how to schedule the guarding nodes?

- Time is slotted.
- At each time slot at most k guarding nodes are turned on.
- Periodic schedule.

Question: when and which set of guarding nodes to turn on?

Coverage in a Building

Given n guarding nodes and m target nodes, the set of targets covered by guarding site g_i is $P(g_i)$, how to schedule the guarding nodes?

- Time is slotted.
- At each time slot at most k guarding nodes are turned on.
- Periodic schedule.

Question: when and which set of guarding nodes to turn on?

- Minimize the maximum or average *dark duration*.

Coverage in a Building

Given n guarding nodes and m target nodes, the set of targets covered by guarding site g_i is $P(g_i)$, how to schedule the guarding nodes?

- Time is slotted.
- At each time slot at most k guarding nodes are turned on.
- Periodic schedule.

Question: when and which set of guarding nodes to turn on?

- Minimize the maximum or average *dark duration*.
- Or, meet specific target coverage frequency requirements.

Scheduling for Minimizing Dark Duration [MobiHoc'16]

Given n guarding sites and m target nodes, the set of targets covered by guarding site g_i is $P(g_i)$, suppose at any slot only k guards are turned on, how to schedule them such that no target stays 'in dark' for too long.

Scheduling for Minimizing Dark Duration [MobiHoc'16]

Given n guarding sites and m target nodes, the set of targets covered by guarding site g_i is $P(g_i)$, suppose at any slot only k guards are turned on, how to schedule them such that no target stays 'in dark' for too long.

- Define **max dark duration T(p)** for $p \in D$: p is lighted up at least once every $T(p)$ slots.

Scheduling for Minimizing Dark Duration [MobiHoc'16]

Given n guarding sites and m target nodes, the set of targets covered by guarding site g_i is $P(g_i)$, suppose at any slot only k guards are turned on, how to schedule them such that no target stays 'in dark' for too long.

- Define **max dark duration T(p)** for $p \in D$: p is lighted up at least once every $T(p)$ slots.

$S:$	g_1	g_2	g_3	g_2	g_1	g_2	g_3	g_2	$\cdots$
	p_1				p_1				
	p_2	p_2		p_2	p_2	p_2		p_2	
		p_3		p_3		p_3		p_3	
			p_4				p_4		

Scheduling for Minimizing Dark Duration

Given n guarding sites and m target nodes, the set of targets covered by guarding site g_i is $P(g_i)$, suppose at any slot only k guards are turned on, how to schedule them such that no target stays 'in dark' for too long.

- **Min Max Dark Duration Scheduling**:

$$\min \max_{p \in D} T(p)$$

Scheduling for Minimizing Dark Duration

Given n guarding sites and m target nodes, the set of targets covered by guarding site g_i is $P(g_i)$, suppose at any slot only k guards are turned on, how to schedule them such that no target stays 'in dark' for too long.

- **Min Max Dark Duration Scheduling**:

$$\min \max_{p \in D} T(p)$$

- Round robin on a **minimum cover**, i.e., minimum number of guards that cover all targets, is optimal.

Scheduling for Minimizing Dark Duration

Given n guarding sites and m target nodes, the set of targets covered by guarding site g_i is $P(g_i)$, suppose at any slot only k guards are turned on, how to schedule them such that no target stays 'in dark' for too long.

- **Min Max Dark Duration Scheduling**:

$$\min \max_{p \in D} T(p)$$

- Round robin on a **minimum cover**, i.e., minimum number of guards that cover all targets, is optimal.

- Min Max problem is tailored towards worst case & sensitive to outliers.

Min Max Dark Duration Scheduling

- Three guards g_1, g_2, g_3 and six targets $p_1, p_2, p_3, p_4, p_5, p_6$.

Min Max Dark Duration Scheduling

- Three guards g_1, g_2, g_3 and six targets $p_1, p_2, p_3, p_4, p_5, p_6$.
- $P(g_1) = \{p_1\}$, $P(g_2) = \{p_2\}$, $P(g_3) = \{p_3, p_4, p_5, p_6\}$,

Min Max Dark Duration Scheduling

- Three guards g_1, g_2, g_3 and six targets $p_1, p_2, p_3, p_4, p_5, p_6$.
- $P(g_1) = \{p_1\}$, $P(g_2) = \{p_2\}$, $P(g_3) = \{p_3, p_4, p_5, p_6\}$,
- Min Max: round robin on g_1, g_2, g_3, each target has a dark duration 3.

Min Max Dark Duration Scheduling

- Three guards g_1, g_2, g_3 and six targets $p_1, p_2, p_3, p_4, p_5, p_6$.
- $P(g_1) = \{p_1\}$, $P(g_2) = \{p_2\}$, $P(g_3) = \{p_3, p_4, p_5, p_6\}$,
- Min Max: round robin on g_1, g_2, g_3, each target has a dark duration 3.
- Min Average: it is beneficial to repeat g_3 more often.

Min Max Dark Duration Scheduling

- Three guards g_1, g_2, g_3 and six targets $p_1, p_2, p_3, p_4, p_5, p_6$.
- $P(g_1) = \{p_1\}$, $P(g_2) = \{p_2\}$, $P(g_3) = \{p_3, p_4, p_5, p_6\}$,
- Min Max: round robin on g_1, g_2, g_3, each target has a dark duration 3.
- Min Average: it is beneficial to repeat g_3 more often.
- E.g., repeating g_1, g_3, g_2, g_3 yields average dark duration $8/3 < 3$.

Min Max Dark Duration Scheduling

- Three guards g_1, g_2, g_3 and six targets $p_1, p_2, p_3, p_4, p_5, p_6$.
- $P(g_1) = \{p_1\}$, $P(g_2) = \{p_2\}$, $P(g_3) = \{p_3, p_4, p_5, p_6\}$,
- Min Max: round robin on g_1, g_2, g_3, each target has a dark duration 3.
- Min Average: it is beneficial to repeat g_3 more often.
- E.g., repeating g_1, g_3, g_2, g_3 yields average dark duration $8/3 < 3$.

- **Min Average Dark Duration Scheduling**:

$$\min \sum_{p \in D} w(p) \cdot T(p)$$

where $w(p)$ is a weight parameter.

Algorithms for Min Average Dark Duration Scheduling

Min average optimization: use the weights $w(\cdot)$ to allow more flexibility to adjust to varying guarding requirements.

Algorithms for Min Average Dark Duration Scheduling

Min average optimization: use the weights $w(\cdot)$ to allow more flexibility to adjust to varying guarding requirements.

- Targets with higher importance have higher weights.

Challenge:

- Which guard to repeat, and how many times?
- How to schedule them?

Algorithms for Min Average Dark Duration Scheduling

Min average optimization: use the weights $w(\cdot)$ to allow more flexibility to adjust to varying guarding requirements.

- Targets with higher importance have higher weights.

Challenge:

- Which guard to repeat, and how many times?
- **How to schedule them?**

Special Case: Round Robin On a Permutation π

Find the optimal one π^* that minimizes the min average dark duration.

- Take a target $p \in D$, consider all guards that cover p. We wish them to spread uniformly in π.

Special Case: Round Robin On a Permutation π

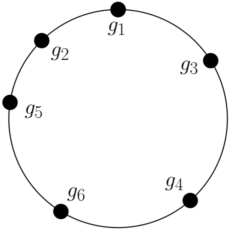
Find the optimal one π^* that minimizes the min average dark duration.

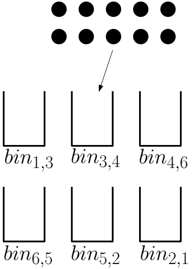
- Take a target $p \in D$, consider all guards that cover p. We wish them to spread uniformly in π.

$G(p) \subseteq V$, $|V| = 16$ $\qquad$ $\ell = |G(p)|$ bins

$G(p) = \{g_1, g_2, g_3, g_4, g_5, g_6\}$ $\qquad$ $m = |G \setminus G(p)|$ balls

Assume $G(p)$ appears in the permutation with this order $g_1, g_3, g_4, g_6, g_5, g_2$





Special Case: Round Robin On a Permutation π

How about a **random** permutation?

Special Case: Round Robin On a Permutation π

How about a **random** permutation?

- Throw m balls randomly into ℓ bins, how many balls do we have in the largest bin?

Special Case: Round Robin On a Permutation π

How about a **random** permutation?

- Throw m balls randomly into ℓ bins, how many balls do we have in the largest bin?
- Compared to the optimal (uniform), the ratio is $\alpha = O(1)$ if $k \geq \log n / \log \log n$, and $\alpha = O(\log n / \log \log n)$ otherwise.

Special Case: Round Robin On a Permutation π

How about a **random** permutation?

- Throw m balls randomly into ℓ bins, how many balls do we have in the largest bin?
- Compared to the optimal (uniform), the ratio is $\alpha = O(1)$ if $k \geq \log n / \log \log n$, and $\alpha = O(\log n / \log \log n)$ otherwise.

What about the general case? How many times shall we repeat a guard? — Could be optimized by a convex program.

Algorithm

Three steps:

- Any optimal schedule can be turned into a periodic schedule with a factor 2 approximation.

Algorithm

Three steps:

- Any optimal schedule can be turned into a periodic schedule with a factor 2 approximation.
- Run a convex optimization algorithm to find $f(g)$ – frequency of repeats for guard g

$$\min \sum_p \frac{w(p)}{\sum_{g \text{ covers } p} f(g)}$$

subject to $\sum_g f(g) = 1,\ f(g) \geq 0.$

Algorithm

Three steps:

- Any optimal schedule can be turned into a periodic schedule with a factor 2 approximation.
- Run a convex optimization algorithm to find $f(g)$ – frequency of repeats for guard g

  $$\min \sum_{p} \frac{w(p)}{\sum_{g \text{ covers } p} f(g)}$$

  subject to $\sum_g f(g) = 1$, $f(g) \geq 0$.
- Turn $f(g)$ into a nearby rational number and find # repeats $\tau(g)$ and length of the schedule T s.t. $\tau(g)/(kT) \approx f(g)$.
- Repeating g_i $\tau(g_i)$ times, and choose a random permutation on them.

Algorithm

Three steps:

- Any optimal schedule can be turned into a periodic schedule with a factor 2 approximation.
- Run a convex optimization algorithm to find $f(g)$ – frequency of repeats for guard g

$$\min \sum_{p} \frac{w(p)}{\sum_{g \text{ covers } p} f(g)}$$

  subject to $\sum_g f(g) = 1$, $f(g) \geq 0$.
- Turn $f(g)$ into a nearby rational number and find # repeats $\tau(g)$ and length of the schedule T s.t. $\tau(g)/(kT) \approx f(g)$.
- Repeating g_i $\tau(g_i)$ times, and choose a random permutation on them.

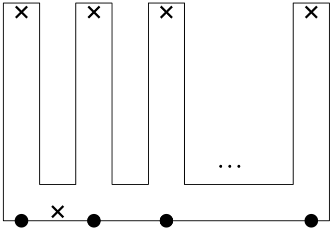
This algorithm gives $(2+\varepsilon)\alpha$ approximation in expectation, $\alpha = O(1)$ if $k \geq \log n / \log\log n$, and $\alpha = O(\log n / \log\log n)$ otherwise.

Min Energy Scheduling with Target Coverage Frequency [INFOCOM'17]

Min Energy Scheduling with Target Coverage Frequency: Suppose target j needs to be covered every f_j slots, how to schedule guards to meet the requirement such that at each slot only k guards are turned on? Minimize k.

Min Energy Scheduling with Target Coverage Frequency [INFOCOM'17]

Min Energy Scheduling with Target Coverage Frequency: Suppose target j needs to be covered every f_j slots, how to schedule guards to meet the requirement such that at each slot only k guards are turned on? Minimize k.



Bottom target must be covered every slot while each of the top target must be covered every m slots.

Min Resouce Scheduling with Target Coverage Frequency

Use set multi-cover & randomization. Details skipped.

- $O(\log n + \log m)$ approximation.
- Geometric setting: better approximation.

Part II: Collaborative Robot Path Planning

Problem: Given a set of n sites $\{p_1, p_2, \cdots, p_n\}$, schedule mobile nodes (vehicles, robots) to serve them (collection, delivery). Suppose the mobile nodes travel with unit speed.

Part II: Collaborative Robot Path Planning

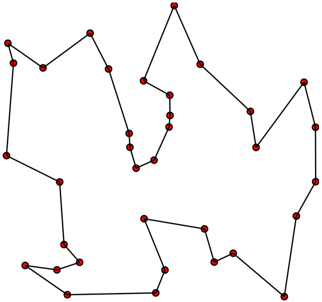
Problem: Given a set of n sites $\{p_1, p_2, \cdots, p_n\}$, schedule mobile nodes (vehicles, robots) to serve them (collection, delivery). Suppose the mobile nodes travel with unit speed.

- Minimize total travel time: traveling salesman problem.

Part II: Collaborative Robot Path Planning

Problem: Given a set of n sites $\{p_1, p_2, \cdots, p_n\}$, schedule mobile nodes (vehicles, robots) to serve them (collection, delivery). Suppose the mobile nodes travel with unit speed.

- Minimize total travel time: traveling salesman problem.



Additional constraints: node capacity, time-window.

Mobile Collection Problem [ALGOSENSORS'15]

Consider sites that collect mail/donations/data. Each site has accumulation **rate** r_i and **capacity** c_i, if the capacity is reached additional items are lost. Schedule the path for k mobile nodes to maximize items collected.

Mobile Collection Problem [ALGOSENSORS'15]

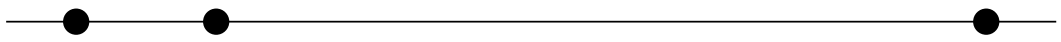
Consider sites that collect mail/donations/data. Each site has accumulation **rate** r_i and **capacity** c_i, if the capacity is reached additional items are lost. Schedule the path for k mobile nodes to maximize items collected.

- Even for $k = 1$ and sites in 1D with unit rates and capacities, the optimal solution is not TSP anymore.

Mobile Collection Problem [ALGOSENSORS'15]

Consider sites that collect mail/donations/data. Each site has accumulation **rate** r_i and **capacity** c_i, if the capacity is reached additional items are lost. Schedule the path for k mobile nodes to maximize items collected.

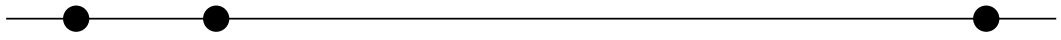
- Even for $k = 1$ and sites in 1D with unit rates and capacities, the optimal solution is not TSP anymore.



Mobile Collection Problem [ALGOSENSORS'15]

Consider sites that collect mail/donations/data. Each site has accumulation **rate** r_i and **capacity** c_i, if the capacity is reached additional items are lost. Schedule the path for k mobile nodes to maximize items collected.

- Even for $k = 1$ and sites in 1D with unit rates and capacities, the optimal solution is not TSP anymore.



- Theorem: The optimal solution is a zig-zag tour on an interval.
- Run Dynamic Programming in $O(n^2)$ time.

Mobile Collection Problem

Approximation factors:

With Sites	Single robot	k-robots	No Loss
on a Line	in P	$1/3$	exact
on a Tree	pseudo-poly	$\frac{1}{3}(1 - 1/e^{\frac{1}{2+\varepsilon}})$	12
General Metric	$1/6 - \varepsilon$		
Euclidean	$1/3 - \varepsilon$	$\frac{1}{3}(1 - 1/e^{1-\varepsilon})$	
Diff Capacities	$O(1/m)$		$O(m)$

Table: $m \leq \log(\frac{c_{max}}{c_{min}})$ where c_{max} is the largest capacity and c_{min} is the smallest capacity. For the results in the first four rows, we assume that the sensor capacities are all the same. ε is any positive constant.

Remark: nodes might be starved.

Time-Window Path Planning

UPS package delivery: deliver packages to user v_i, during the specified time window $[r_i, d_i]$.

- **Time-Window Prize Collecting Problem**: maximize the number of sites visited within their time windows.
- **Time-Window TSP**: minimize the length of path visiting all sites within their time windows, if possible.

Time-Window Path Planning

UPS package delivery: deliver packages to user v_i, during the specified time window $[r_i, d_i]$.

- **Time-Window Prize Collecting Problem**: maximize the number of sites visited within their time windows.
- **Time-Window TSP**: minimize the length of path visiting all sites within their time windows, if possible.

Open for ≥ 20 years even for 1D: best approximation is $O(\log n)$.

Mobile Collection Problem [WAFR'16]

TWPC, TWTSP in 1D: optimal with **relaxed time window** to $[r_i - \varepsilon L_i, d_i + \varepsilon L_i]$, for $L_i = d_i - r_i$.

TWTSP in 2D (general metric): (α, β)-**dual appoxiamtion**: speed $\leq \alpha$, travel distance $\leq \beta \cdot OPT$.

- $\alpha, \beta = O(1)$, unit time windows.
- $\alpha, \beta = O(\log L_{\max})$, when all window size are power of two.
- $\beta = O(\log n)$ if robot travels at possibly infinity speed.

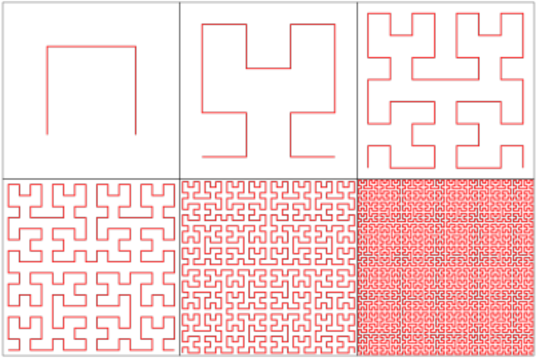
Part III: Robot Sweeping a Complex Domain

Part III: Robot Sweeping a Complex Domain

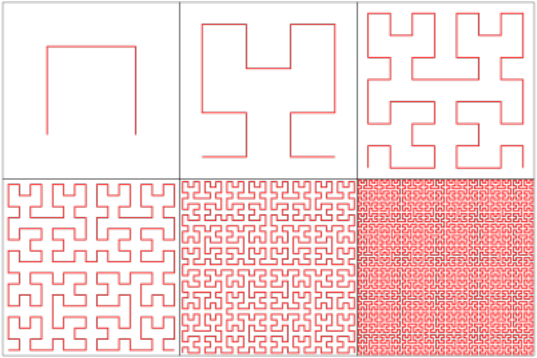
Two challenges:

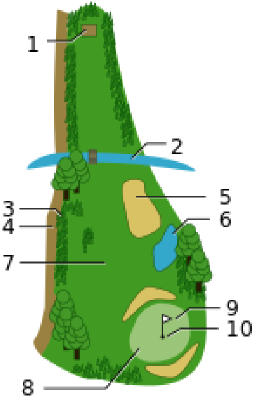
- How to handle complex geometry/topology?
- How to make coordination of multiple robots easy?

Space Filling Curves



Space Filling Curves

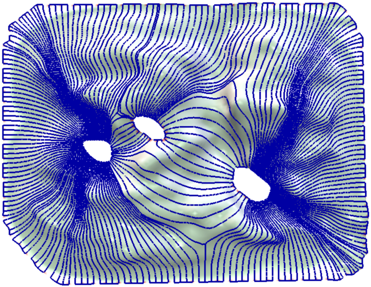


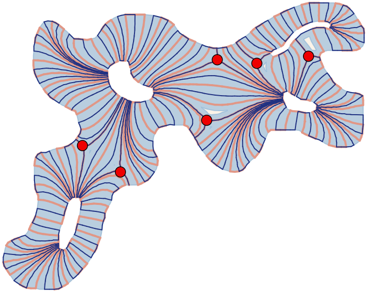


Question: what if the domain shape/topology is complicated?

Space Filling Curve for Complex Geometry [ICRA'17]

Use surface patrametrization to create topology-aware space filling curves.





Conclusion and Ongoing Work

- Classical problems revisited; performance guarantees.
- Security/safety applications: defend against adversarial, strategic agents.

Acknowledgement

- Esther Arkin, Joe Mitchell, Shan Lin, David Gu, Matthew Johnson, Nirman Kumar
- Kin Sum Liu, Hua Huang, Brent Schiller, Tyler Mayer, Hao Tsung Yang, Mayank Goswami, Gui Citovsky, Jiemin Zeng, Su Jia, Yu-Yao Lin, Chien-Chun Ni.

- http://www3.cs.stonybrook.edu/∼jgao/
- Questions and comments?