Scheduling and Motion Planning for Wireless Sensors and Mobile Networks

Jie Gao

Stony Brook University

September 11th, 2017.

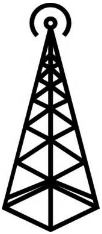
Research Interest and Projects

Computational Geometry, Algorithm Design and Analysis, Wireless Networks, Social Networks

- Geometric Methods for Network Analysis.
- Location and Trajectory Privacy.
- Social Influence and Contagions
- Scheduling Algorithms.

Scheduling Wireless Devices





Considerations:

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

Constraint dimensions:

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

Two Application Scenarios

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

- Smart building: optimize energy usage, improve safety & security.

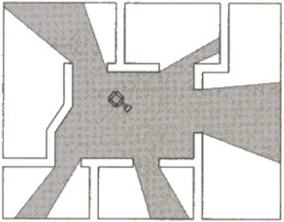
Two Application Scenarios

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

- Smart building: optimize energy usage, improve safety & security.
- Mobile networks: assign mobile nodes to collect data from sensors with storage capacities.

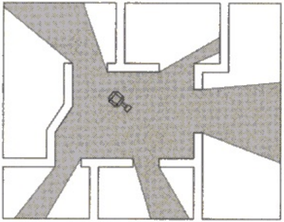
Coverage in Smart Building

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



Coverage in Smart Building

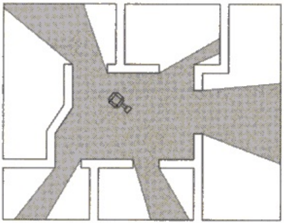
Given n camera nodes and m target nodes, the set of targets covered by sensors g_i is $P(g_i)$, how to schedule the camera nodes?



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

Coverage in Smart Building

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



- If sensors are turned on all the time: Art Gallery Problem;
- Sensors are not turned on all the time: Duty Cycle Scheduling.

Coverage in Smart Building

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

- Targets are frequently covered;

Coverage in Smart Building

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

- Targets are frequently covered;
- Energy usage is reduced.

Coverage in Smart Building

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

- Targets are frequently covered;
- Energy usage is reduced.
- Generic coverage assumption.

Coverage in Smart Building

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

- Targets are frequently covered;
- Energy usage is reduced.
- Generic coverage assumption.

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

Coverage in Smart Building

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

- Targets are frequently covered;
- Energy usage is reduced.
- Generic coverage assumption.

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

- Time is slotted.

Coverage in Smart Building

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

- Targets are frequently covered;
- Energy usage is reduced.
- Generic coverage assumption.

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

- Time is slotted.
- At each time slot at most k cameras are turned on.

Coverage in Smart Building

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

- Targets are frequently covered;
- Energy usage is reduced.
- Generic coverage assumption.

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

- Time is slotted.
- At each time slot at most k cameras are turned on.
- Minimize the maximum or average *dark duration*.

Coverage in Smart Building

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

- Targets are frequently covered;
- Energy usage is reduced.
- Generic coverage assumption.

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

- Time is slotted.
- At each time slot at most k cameras are turned on.
- Minimize the maximum or average *dark duration*.
- Or, meet specific target coverage frequency requirements.

Scheduling for Minimizing Dark Duration

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

Scheduling for Minimizing Dark Duration

Given n camera nodes and m target nodes, the set of targets covered by sensors g_i is $P(g_i)$, suppose at any slot only k sensors are turned on, how to schedule sensors 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

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

- **Min Max Dark Duration Scheduling**:

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

- **Min Average Dark Duration Scheduling**:

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

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

Min Max Dark Duration Scheduling

Assume that G is a **minimal cover**, i.e., removing any sensor renders targets not fully covered, then round robin on G is optimal.

Min Max Dark Duration Scheduling

Assume that G is a **minimal cover**, i.e., removing any sensor renders targets not fully covered, then round robin on G is optimal.

- Max dark duration: $\lceil n/k \rceil$

Min Max Dark Duration Scheduling

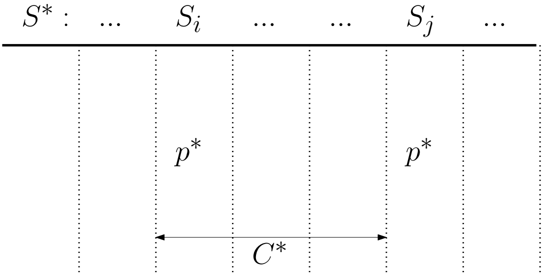
Assume that G is a **minimal cover**, i.e., removing any sensor renders targets not fully covered, then round robin on G is optimal.

- Max dark duration: $\lceil n/k \rceil$
- Claim: OPT $C^* \geq \lceil n/k \rceil$.

Min Max Dark Duration Scheduling

Assume that G is a **minimal cover**, i.e., removing any sensor renders targets not fully covered, then round robin on G is optimal.

- Max dark duration: $\lceil n/k \rceil$
- Claim: OPT $C^* \geq \lceil n/k \rceil$.

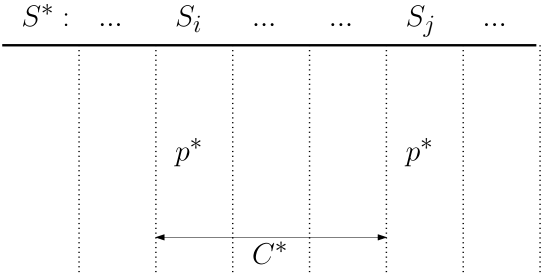


Assume $C^* < \lceil n/k \rceil$ realized by $p^* \in D$.

Min Max Dark Duration Scheduling

Assume that G is a **minimal cover**, i.e., removing any sensor renders targets not fully covered, then round robin on G is optimal.

- Max dark duration: $\lceil n/k \rceil$
- Claim: OPT $C^* \geq \lceil n/k \rceil$.

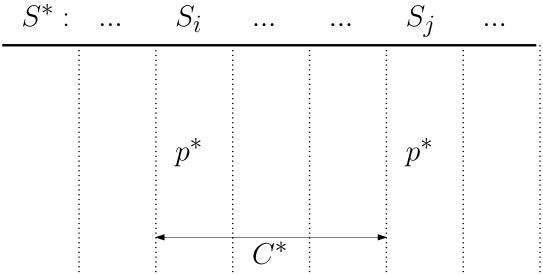


Assume $C^* < \lceil n/k \rceil$ realized by $p^* \in D$.
Then at least one guard g does not appear during the interval of length C^* as shown below.

Min Max Dark Duration Scheduling

Assume that G is a **minimal cover**, i.e., removing any sensor renders targets not fully covered, then round robin on G is optimal.

- Max dark duration: $\lceil n/k \rceil$
- Claim: OPT $C^* \geq \lceil n/k \rceil$.



Assume $C^* < \lceil n/k \rceil$ realized by $p^* \in D$.
Then at least one guard g does not appear during the interval of length C^* as shown below.
$\rightarrow$ the point that is only guarded by g has dark duration $> C^*$.

Min Max Dark Duration Scheduling

If G is **not** a minimal cover, then round robin on a **minimum cover** (a subset of G) is optimal.

Min Max Dark Duration Scheduling

If G is **not** a minimal cover, then round robin on a **minimum cover** (a subset of G) is optimal.

NP-hard: reduction from set cover.

Min Max Dark Duration Scheduling

If G is **not** a minimal cover, then round robin on a **minimum cover** (a subset of G) is optimal.

NP-hard: reduction from set cover.

- Any optimal schedule can be turned into an optimal periodic schedule.

Min Max Dark Duration Scheduling

If G is **not** a minimal cover, then round robin on a **minimum cover** (a subset of G) is optimal.

NP-hard: reduction from set cover.

- Any optimal schedule can be turned into an optimal periodic schedule.
- The set of sensors used in one cycle of the periodic schedule is the minimum set cover.

Min Max Dark Duration Scheduling

If G is **not** a minimal cover, then round robin on a **minimum cover** (a subset of G) is optimal.

NP-hard: reduction from set cover.

- Any optimal schedule can be turned into an optimal periodic schedule.
- The set of sensors used in one cycle of the periodic schedule is the minimum set cover.

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

Min Average Dark Duration Scheduling: Motivation

Example:

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

Min Average Dark Duration Scheduling: Motivation

Example:

- 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 Average Dark Duration Scheduling: Motivation

Example:

- 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 Dark Duration Scheduling: Motivation

Example:

- 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 Average Dark Duration Scheduling: Motivation

Example:

- 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: Motivation

Example:

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

Min Average Dark Duration Scheduling: Motivation

Example:

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

- Targets with higher importance have higher weights.

Algorithms for Min Average Dark Duration Scheduling

Challenge:

- Which sensor 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 uniformly spread in π.

Special Case: Round Robin On a Permutation π

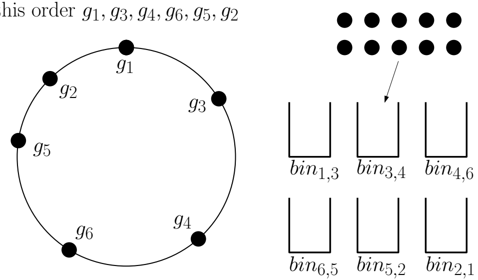
Find the optimal one π^* that minimizes the min average dark duration.

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

More General: Best Periodic Schedule

The guards in one full cycle T of a periodic schedule: each guard g appears $\tau(g)$ times. Total # guards in one cycle is $\sum_g \tau(g) = Tk$.

More General: Best Periodic Schedule

The guards in one full cycle T of a periodic schedule: each guard g appears $\tau(g)$ times. Total # guards in one cycle is $\sum_g \tau(g) = Tk$.

Goal: solve for $\tau(g)$, $\forall g$, and T.

- Define $f^*(g) = \tau^*(g)/(kT^*)$, frequency.

More General: Best Periodic Schedule

The guards in one full cycle T of a periodic schedule: each guard g appears $\tau(g)$ times. Total # guards in one cycle is $\sum_g \tau(g) = Tk$.

Goal: solve for $\tau(g)$, $\forall g$, and T.

- Define $f^*(g) = \tau^*(g)/(kT^*)$, frequency.
- Define $X^*(p) = \sum_{g \text{ covers } p} \tau^*(g)$, # guards that cover p.

More General: Best Periodic Schedule

The guards in one full cycle T of a periodic schedule: each guard g appears $\tau(g)$ times. Total # guards in one cycle is $\sum_g \tau(g) = Tk$.

Goal: solve for $\tau(g)$, $\forall g$, and T.

- Define $f^*(g) = \tau^*(g)/(kT^*)$, frequency.
- Define $X^*(p) = \sum_{g \text{ covers } p} \tau^*(g)$, # guards that cover p.
- Claim: $T^*(p) \geq T^*/X^*(p)$ – the best case is when all guards that cover p are uniformly spread in T^*.

More General: Best Periodic Schedule

The guards in one full cycle T of a periodic schedule: each guard g appears $\tau(g)$ times. Total # guards in one cycle is $\sum_g \tau(g) = Tk$.

Goal: solve for $\tau(g)$, $\forall g$, and T.

- Define $f^*(g) = \tau^*(g)/(kT^*)$, frequency.
- Define $X^*(p) = \sum_{g \text{ covers } p} \tau^*(g)$, # guards that cover p.
- Claim: $T^*(p) \geq T^*/X^*(p)$ – the best case is when all guards that cover p are uniformly spread in T^*.
- Objective function

$$A^* = \sum_p w(p)T^*(p) \geq B^* = \sum_p \frac{w(p)/k}{\sum_{g \text{ covers } p} f^*(g)}$$

More General: Best Periodic Schedule

The guards in one full cycle T of a periodic schedule: each guard g appears $\tau(g)$ times. Total # guards in one cycle is $\sum_g \tau(g) = Tk$.
Goal: solve for $\tau(g)$, $\forall g$, and T.

- Define $f^*(g) = \tau^*(g)/(kT^*)$, frequency.
- Define $X^*(p) = \sum_{g \text{ covers } p} \tau^*(g)$, # guards that cover p.
- Claim: $T^*(p) \geq T^*/X^*(p)$ – the best case is when all guards that cover p are uniformly spread in T^*.
- Objective function

$$A^* = \sum_p w(p) T^*(p) \geq B^* = \sum_p \frac{w(p)/k}{\sum_{g \text{ covers } p} f^*(g)}$$

B is a convex function of $f(g)$ – we can minimize the right hand side!

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 $f(g)$

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

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 $f(g)$

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

  subject to $\sum_g f(g) = 1$, $f(g) \geq 0$.
- Turn $f(g)$ into a nearby rational number and find $\tau(g)$, T such that $\tau(g)/(kT) \approx f(g)$.
- 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 $f(g)$

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

  subject to $\sum_g f(g) = 1$, $f(g) \geq 0$.
- Turn $f(g)$ into a nearby rational number and find $\tau(g)$, T such that $\tau(g)/(kT) \approx f(g)$.
- Repeating g_i $\tau(g_i)$ times, and choose a random permutation on them.

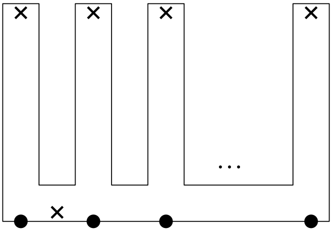
This algorithm gives $(2+\varepsilon)\alpha$ approximation in expectation, $\alpha = O(1)$ if $k \geq \log n / \log \log n$, and $\alpha = O(\log n / \log \log n)$ otherwise.

Min Energy Scheduling with Target Coverage Frequency

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

Min Energy Scheduling with Target Coverage Frequency

Min Energy Scheduling with Target Coverage Frequency: Suppose target j needs to be covered every f_j slots, how to schedule sensors to meet the requirement such that at each slot only k sensors 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 Energy Scheduling with Target Coverage Frequency

Use set multi-cover & randomization. Details skipped.

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

Part II: Path Planning for Mobile Nodes

Problem: Given a set of n sensor nodes $\{p_1, p_2, \cdots, p_n\}$, schedule mobile nodes to serve them (data collection, battery recharging). Suppose the mobile node travels with unit speed.

Part II: Path Planning for Mobile Nodes

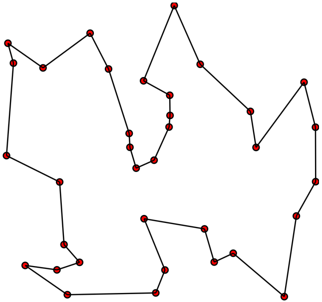
Problem: Given a set of n sensor nodes $\{p_1, p_2, \cdots, p_n\}$, schedule mobile nodes to serve them (data collection, battery recharging). Suppose the mobile node travels with unit speed.

- Minimize total travel time: traveling salesman problem.

Part II: Path Planning for Mobile Nodes

Problem: Given a set of n sensor nodes $\{p_1, p_2, \cdots, p_n\}$, schedule mobile nodes to serve them (data collection, battery recharging). Suppose the mobile node travels with unit speed.

- Minimize total travel time: traveling salesman problem.



Additional constraints: data rate/node capacity, time-window.

Data Collection Problem

Each node has data rate r_i and capacity c_i, if the capacity is reached additional data is lost. Schedule the path for k mobile nodes to maximize data collected.

Data Collection Problem

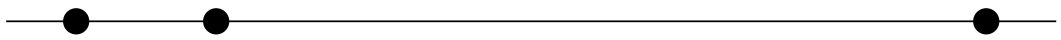
Each node has data rate r_i and capacity c_i, if the capacity is reached additional data is lost. Schedule the path for k mobile nodes to maximize data collected.

- Even for $k = 1$ and sensors with unit data rates and capacities on a line, the optimal solution is not TSP anymore.

Data Collection Problem

Each node has data rate r_i and capacity c_i, if the capacity is reached additional data is lost. Schedule the path for k mobile nodes to maximize data collected.

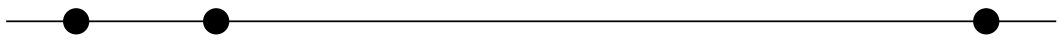
- Even for $k = 1$ and sensors with unit data rates and capacities on a line, the optimal solution is not TSP anymore.



Data Collection Problem

Each node has data rate r_i and capacity c_i, if the capacity is reached additional data is lost. Schedule the path for k mobile nodes to maximize data collected.

- Even for $k = 1$ and sensors with unit data rates and capacities on a line, 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.

Data Collection Problem

With Sensors	Single mule	k-mule	No Data Loss
on a Line	in P	$\frac{1}{3}$	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	$O(\frac{1}{m})$		$O(m)$

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.

Example: Single Mule, Capacity c, Euclidean Setting

Algorithm:

- Find a path of length $c/2$ w/ max # R of nodes.
- Travelling back and forth along it gives a tour.
- No data loss, data rate$=R$.

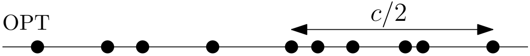
Example: Single Mule, Capacity c, Euclidean Setting

Algorithm:

- Find a path of length $c/2$ w/ max $\#$ R of nodes.
- Travelling back and forth along it gives a tour.
- No data loss, data rate$=R$.

Proof: OPT $R^* \leq 3R$.

- Find in the OPT tour the interval of length $c/2$ with max data collection rate (which is at least R^*).



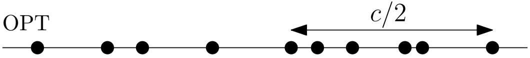
Example: Single Mule, Capacity c, Euclidean Setting

Algorithm:

- Find a path of length $c/2$ w/ max # R of nodes.
- Travelling back and forth along it gives a tour.
- No data loss, data rate$=R$.

Proof: OPT $R^* \leq 3R$.

- Find in the OPT tour the interval of length $c/2$ with max data collection rate (which is at least R^*).



- The interval has at most R *distinct* nodes.
- Total data collected from any sensor is c (on the first visit) and $c/2$ after each $c/2$ interval. Data rate is at most $3R$.

Conclusion

- Classical problems revisited.
- Worst-case performance guarantees.
- Optimization and scheduling are a crucial part of the machine intelligence era.

Acknowledgement

- Kin Sum Liu, Jie Gao, Shan Lin, Hua Huang, Brent Schiller, Joint Sensor Duty Cycle Scheduling with Coverage Guarantee, MobiHoc'2016.
- Kin Sum Liu, Tyler Mayer, Hao Tsung Yang, Esther Arkin, Jie Gao, Mayank Goswami, Matthew P. Johnson, Nirman Kumar, Shan Lin, Joint Sensing Duty Cycle Scheduling for Heterogeneous Coverage Guarantee, INFOCOM'2017.
- Gui Citovsky, Jie Gao, Joseph Mitchell, Jiemin Zeng, Exact and Approximation Algorithms for Data Mule Scheduling in a Sensor Network, ALGOSENSORS 2015.
- http://www3.cs.stonybrook.edu/~jgao/

- Questions and comments?