Stochastic Event Capture Using Mobile Sensors Subject to a Quality Metric

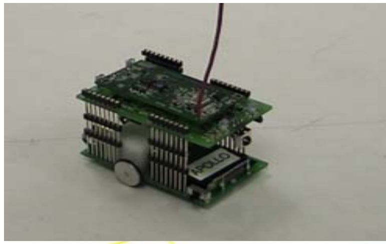
Nabhendra Bisnik, Alhussein A. Abouzeid, and Volkan Isler

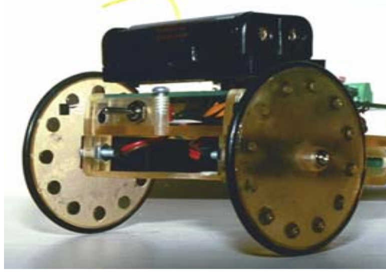
Rensselaer Polytechnic Institute (RPI)

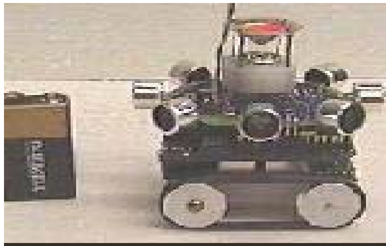
Troy, NY

Mobile Sensors

- Advances in robotics and sensor technology has enabled deployment of smart mobile sensors
- Advantages of mobile sensors:
  - An adversary has to always guess
  - All points can be eventually covered
  - Sensors may settle in “good” positions
  - Move around obstructions
  - Number of sensors required may be reduced





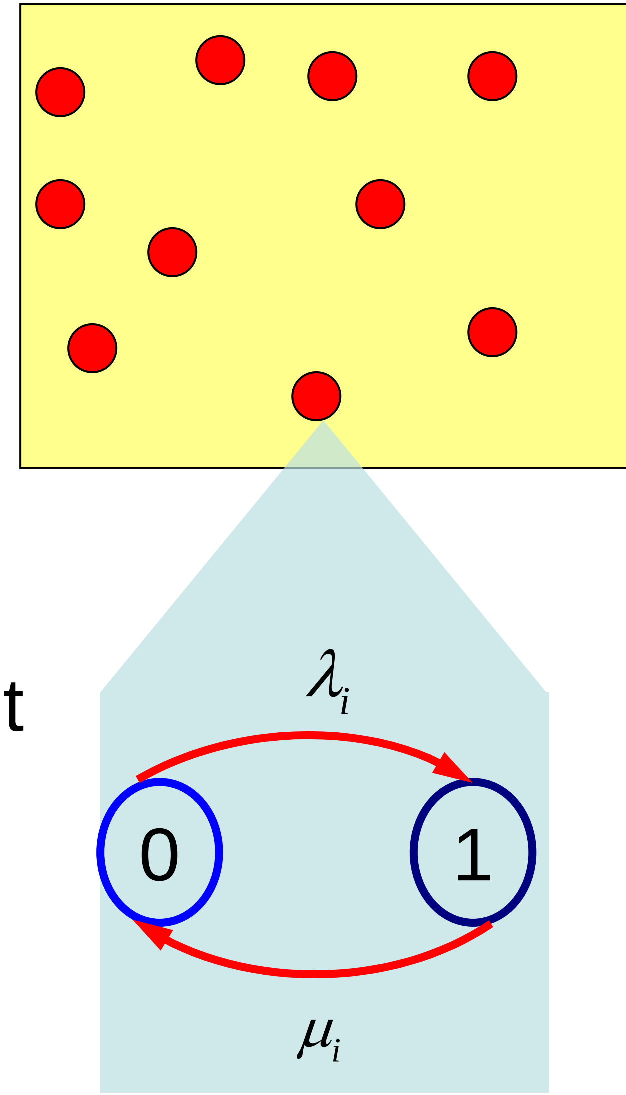


Does Mobility Always Increase Coverage?

- The answer is no!!
- It depends on the phenomena
- Stationary coverage is binary, while mobile coverage is fuzzy
- For random mobility, probabilistic notion of coverage
- Mobility useful in covering events that last over a large time periods
- May not be useful for covering events that are short lived

The Event Capture Problem

- Events appear and disappear at certain points called Points of Interest (PoI)
- The event dynamics at each PoI is known
- An event is *captured* if a sensor visits the PoI when the event is present
- Quality of coverage (QoC) metrics
  - Fraction of events captured
  - Probability that an event is lost

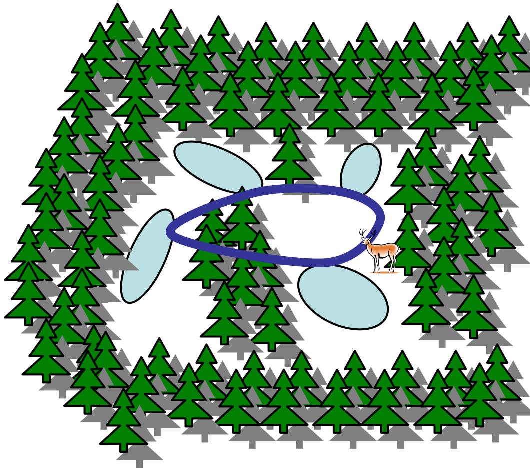


Our Contributions

- **Analytical study** of how quality of coverage scales with parameters such as velocity, number of sensors and event dynamics
- **Algorithms** for Bound Event Loss Probability (***BELP***) Problem
  - **Minimum Velocity BELP (MV-BELP):** What is the minimum velocity with which a sensor may satisfy the required QoC
  - **Minimum Sensor BELP (MS-BELP):** If v fixed what is the minimum number of sensors required
- The problems can be optimally solved for special cases, general problem NP-hard

Applications of our Work

- Habitat Monitoring: PoIs – points frequented by animals, Event – arrival of an animal



- Surveillance: PoIs – vulnerable points, Event – arrival of adversary

- Hybrid Sensor Network: PoIs – stationary sensors, Event – arrival of data
- Supply Chain: PoI – Factories, Event – Arrival of new batch

Talk Outline

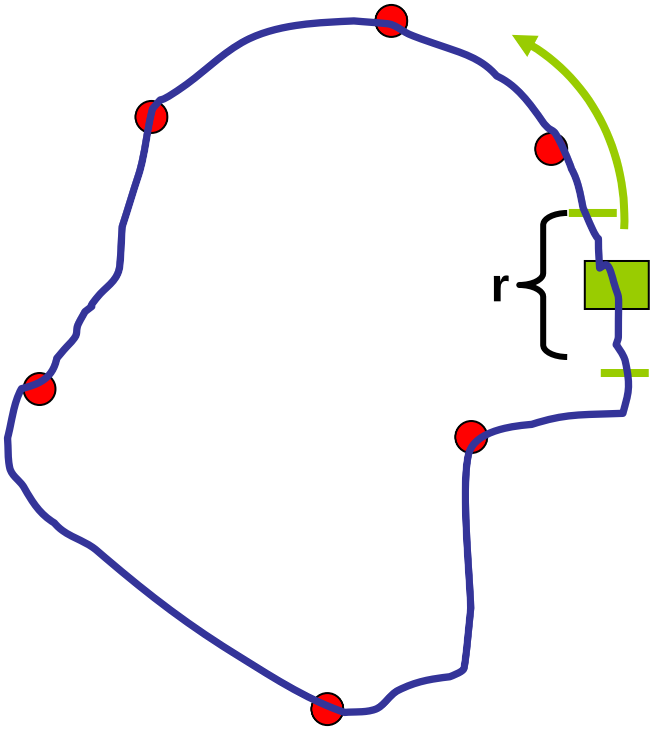
- Analytical results: When is mobility useful?
- BELP Problem
- Algorithms for MV-BELP problem
  - Restricted motion case
  - Unrestricted motion case
- Algorithms for MS-BELP problem
  - Restricted motion case
  - Unrestricted motion case
- Summary and Future Works

Talk Outline

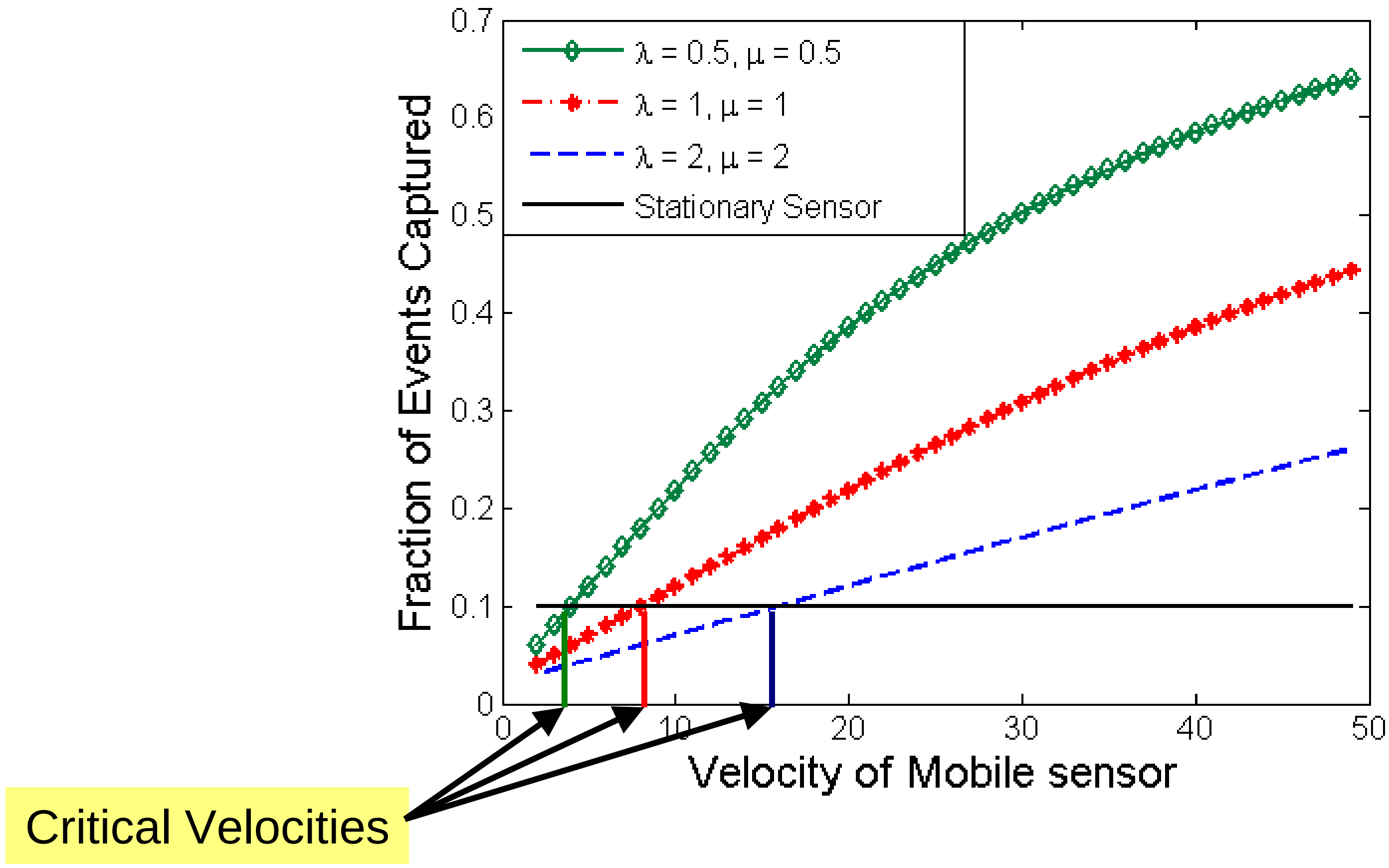
- Analytical results: When is mobility useful?
- BELP Problem
- Algorithms for MV-BELP problem
  - Restricted motion case
  - Unrestricted motion case
- Algorithms for MS-BELP problem
  - Restricted motion case
  - Unrestricted motion case
- Summary and Future Works

A Mobile Coverage Scenario

- n PoIs have to be covered using a mobile sensor
- Events arrive at rate λ and depart at rate μ
- Velocity of mobile sensor is v and sensing range is r
- The mobile sensor moves along a closed curve of length D to cover the PoIs
- We evaluate the fraction of events captured

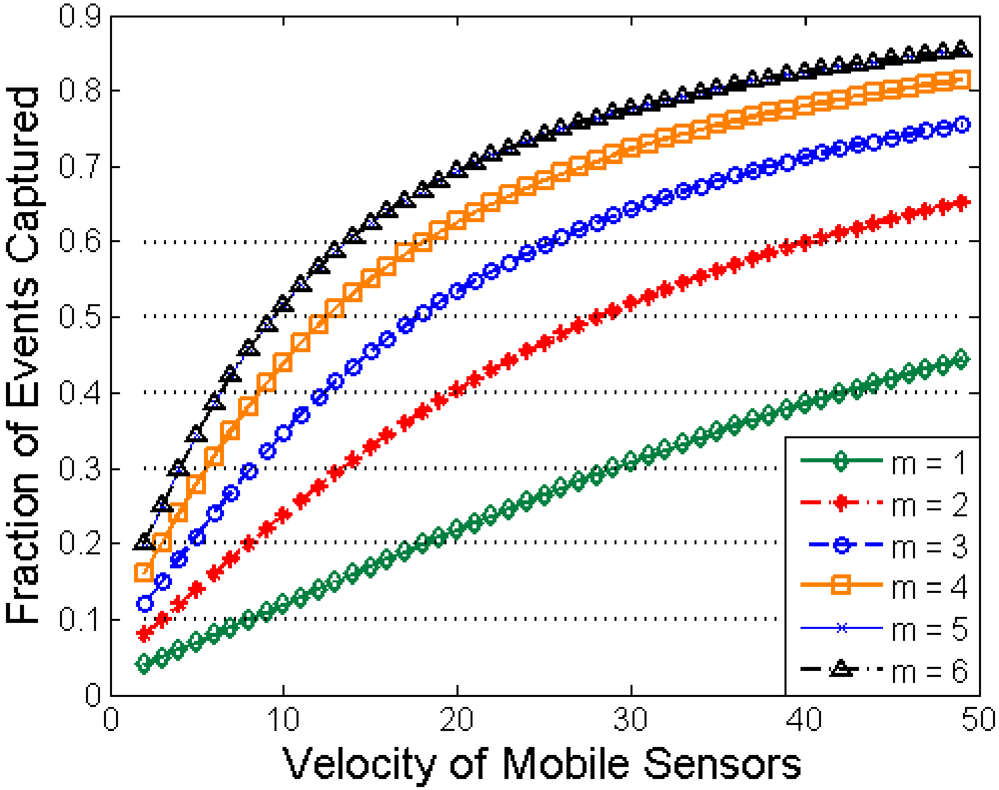


Fraction of Events Captured



If the velocity of the sensor less than the critical velocity, the coverage worse than that achieved by a stationary sensor

Multiple Sensors Case

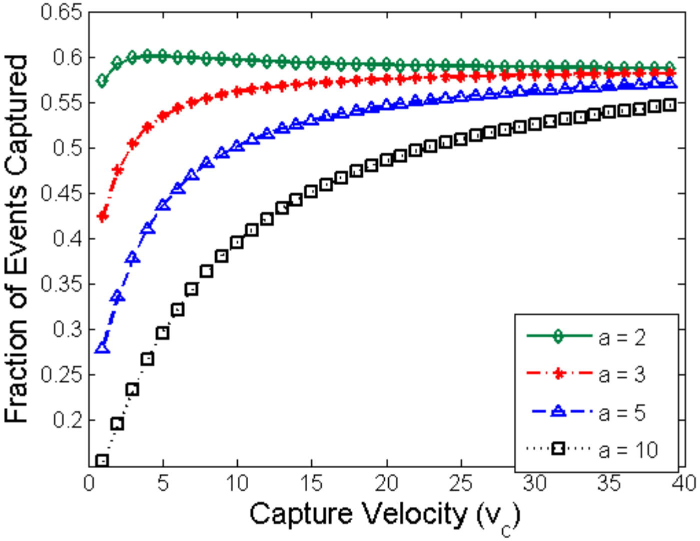


As the number of mobile sensors increase, the critical velocities required for improvements in coverage initially decreases, then starts to increase

Variable Velocity Case

- Intuitively it might be useful to slow down while visiting the PoIs and move at highest possible velocity when no PoIs are visible

- That is, move with velocity v_{max} when no PoIs are visible, move with $v_c \cdot v_{max}$ when a PoI is visible



Slowing down during a visit, in order to spend more fraction of time observing the PoIs does not help either

The solution therefore is to choose “good” paths to move along

Talk Outline

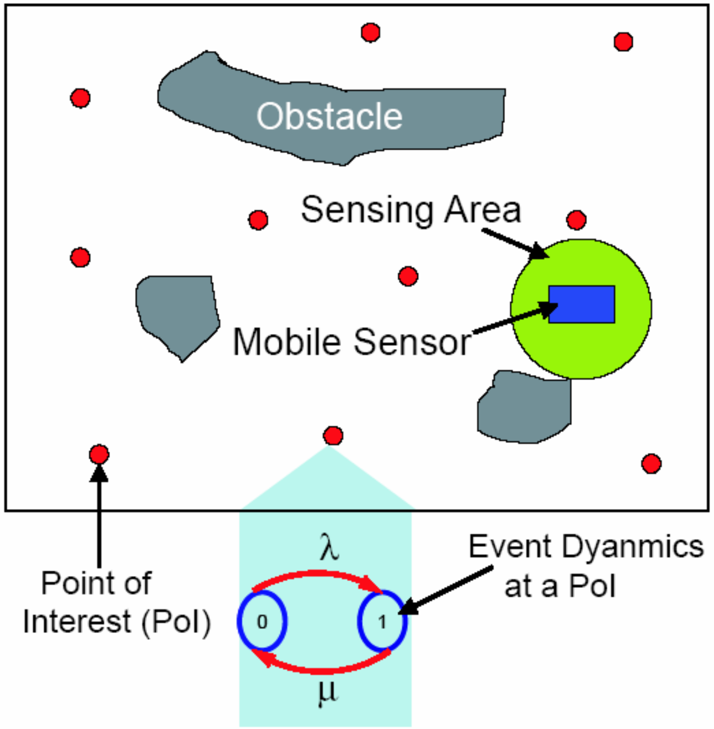
- Analytical results: When is mobility useful?
- BELP Problem
- Algorithms for MV-BELP problem
  - Restricted motion case
  - Unrestricted motion case
- Algorithms for MS-BELP problem
  - Restricted motion case
  - Unrestricted motion case
- Summary and Future Works

BELP Problem

- Bounded event loss probability (BELP) problem: Given a set of PoIs and the event dynamics, plan the motion of sensors such that

$$P[\mathcal{E}_i] < \epsilon \ \forall \ \text{PoIs} \ i$$

- Two optimization goals
  - Single sensor, minimize velocity (MV-BELP)
  - Fix velocity, minimize number of sensors (MS-BELP)



Probability of Event Loss

- Probability of event loss depends on event dynamics and time between two consecutive visits to a PoI

$$\mathcal{P}(T,\lambda,\mu) = 1 - \frac{\mu}{\mu^2-\lambda^2}(\mu e^{-\lambda T}) - \lambda e^{-\mu T}) - \frac{\lambda e^{-\mu T}}{\lambda+\mu} - \frac{\mu\lambda}{\mu^2-\lambda^2}\left(\frac{\mu}{\mu-\lambda}e^{-\lambda T} - \frac{\lambda}{\mu-\lambda}e^{-\mu T} - \lambda T e^{-\mu T}\right)$$

- There exists a T_{crit_i} such that

$$\mathcal{P}(T,\lambda_i,\mu_i) < \epsilon \ \forall \ T < T_{\mathsf{crit}_i}$$

- Thus BELP problem boils down to finding a mobility schedule such that the time between two consecutive visits to PoI i is less than T_{crit_i}

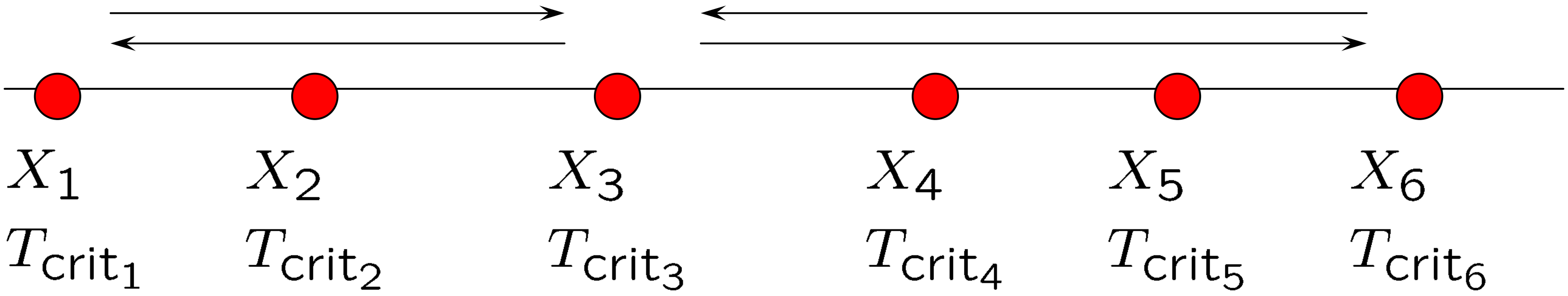
Talk Outline

- Analytical results: When is mobility useful?
- BELP Problem
- Algorithms for MV-BELP problem
  - Restricted motion case
  - Unrestricted motion case
- Algorithms for MS-BELP problem
  - Restricted motion case
  - Unrestricted motion case
- Summary and Future Works

Restricted Motion

- The sensors are restricted to move along a line or a closed curve, along which all the PoIs are located
- Such scenario may arise in cases such as
  - The PoIs are located on road side
  - Trusted paths are created so that sensors do not get lost or stuck
- Restriction of motion to a given path simplifies the BELP problem

MV-BELP: Restricted Motion



- For line case, optimal velocity is given by

$$v_{\text{min}} = \max_{1 \le i \le n} \max \left(\frac{2(X_i - X_1 - 2r)}{T_{\text{crit}_i}}, \frac{2(X_6 - X_i - 2r)}{T_{\text{crit}_i}} \right)$$

- For the closed curved case, optimal velocity obtained by *n* iteration of the procedure for the linear case

MV-BELP: Unrestricted Motion

- Heuristic algorithm
  1. Calculate TSPN path for the set of PoIs
  2. Set $v = \frac{|TSPN(\mathcal{S})|}{T_{\text{min}}}$, $T_{\text{min}} = \min_{1 \le i \le n} T_{\text{crit}_i}$
- If $v^\star$ is the optimal velocity the

$$\frac{v}{v^\star} \le \frac{T_{\text{max}}}{T_{\text{min}}} f(n)$$

  where $T_{\text{max}} = \max_{1 \le i \le n} T_{\text{crit}_i}$ and *f(n)* is approximation ratio of the TSPN algorithm
- If $T_{min} = T_{max}$, then $v/v^\star = f(n)$

Talk Outline

- Analytical results: When is mobility useful?
- BELP Problem
- Algorithms for MV-BELP problem
  - Restricted motion case
  - Unrestricted motion case
- Algorithms for MS-BELP problem
  - Restricted motion case
  - Unrestricted motion case
- Summary and Future Works

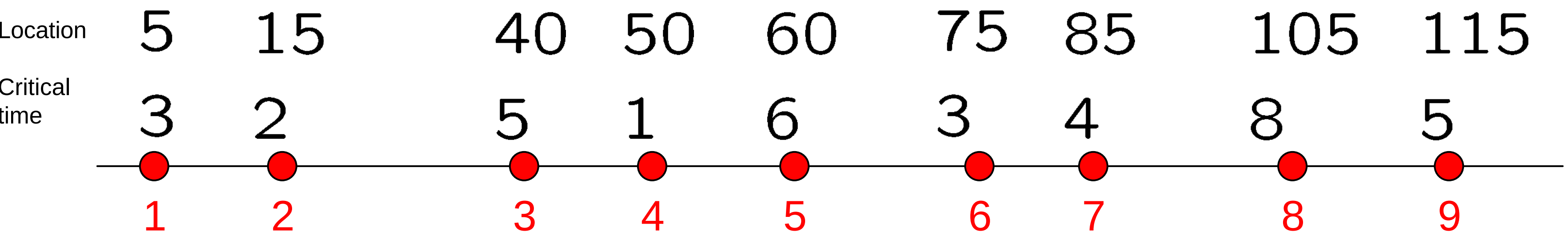
MS-BELP: Restricted Motion

- We propose a greedy heuristic algorithm for line case

```
While all sensors not assigned
      Assign the left-most unassigned PoI to a new sensor
      For all unassigned PoIs
            If QoC at the PoI can be satisfied while
            satisfying QoC at all PoIs in the cover set
                  Add PoI to the cover set of the current
                  sensor
```

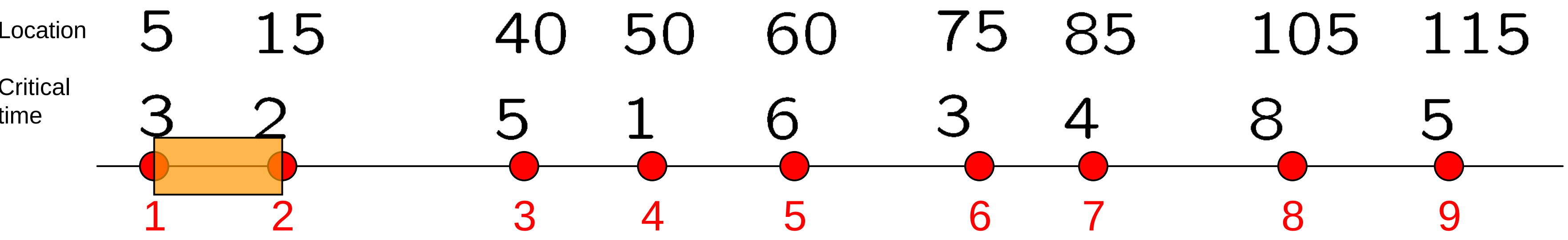
- Use *n+1* iteration of line algorithm to solve the closed curve case
- The greedy heuristic algorithm is within a factor **two** of the optimal

MS-BELP: Restricted Motion



Greedy algorithm for MS-BELP on a line

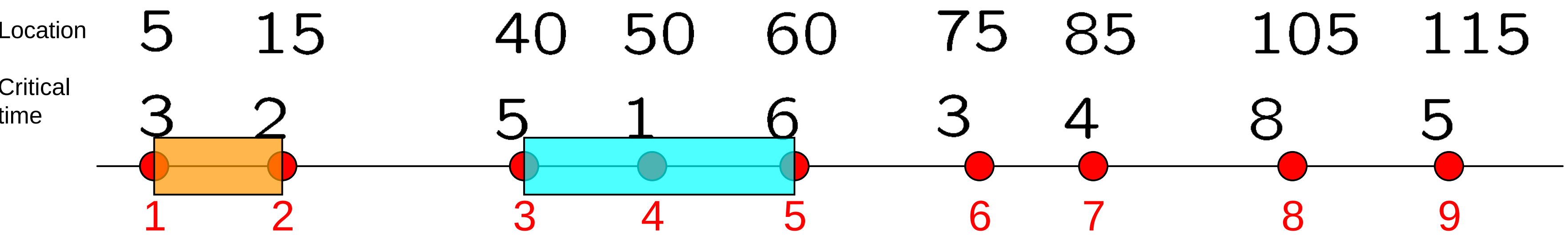
MS-BELP: Restricted Motion



$$\Gamma_1 = \{1, 2\}$$

Greedy algorithm for MS-BELP on a line

MS-BELP: Restricted Motion

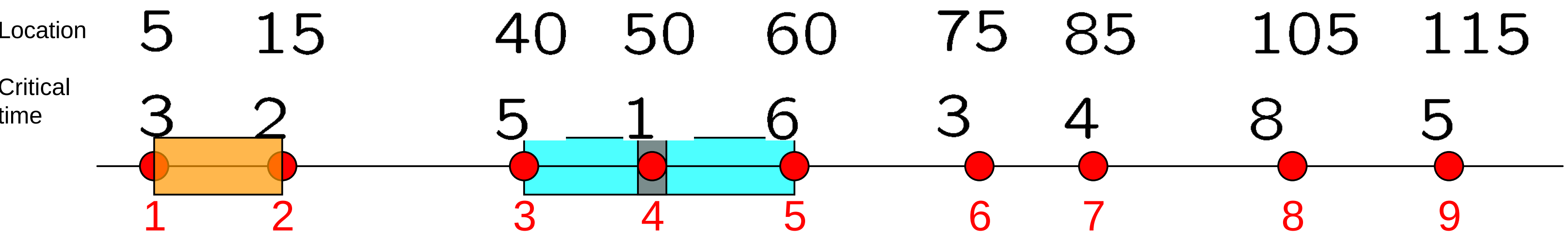


$$\Gamma_1 = \{1, 2\}$$

$$\Gamma_2 = \{3, 5\}$$

Greedy algorithm for MS-BELP on a line

MS-BELP: Restricted Motion



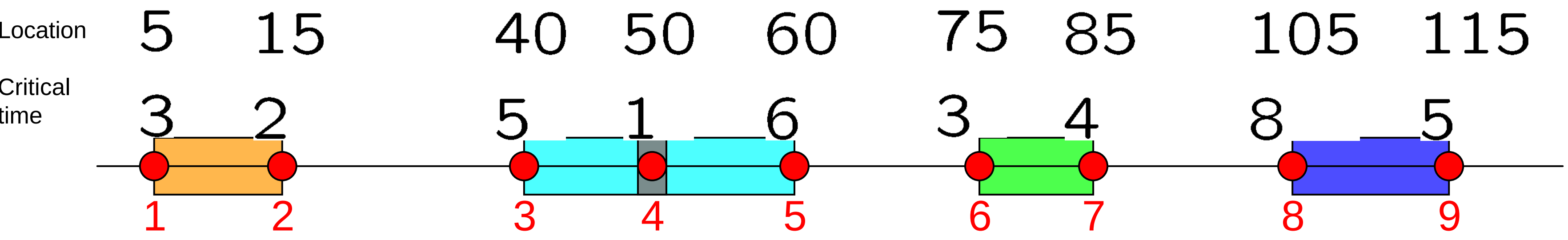
$\Gamma_1 = \{1, 2\}$

$\Gamma_2 = \{3, 5\}$

$\Gamma_3 = \{4\}$, Stationary

Greedy algorithm for MS-BELP on a line

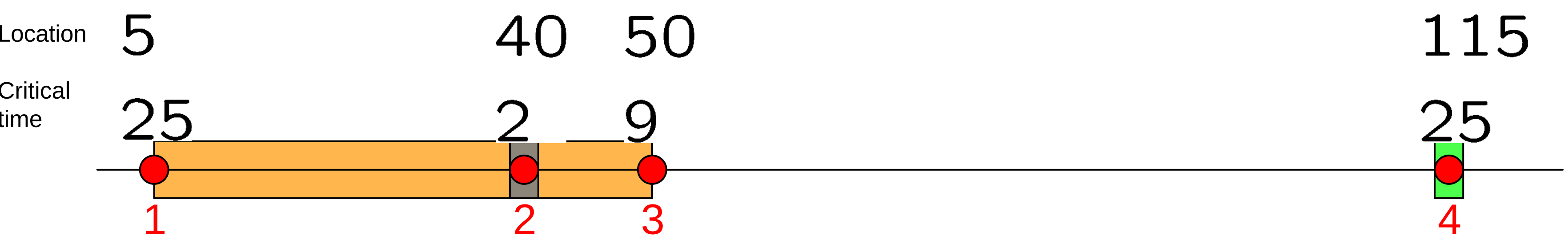
MS-BELP: Restricted Motion



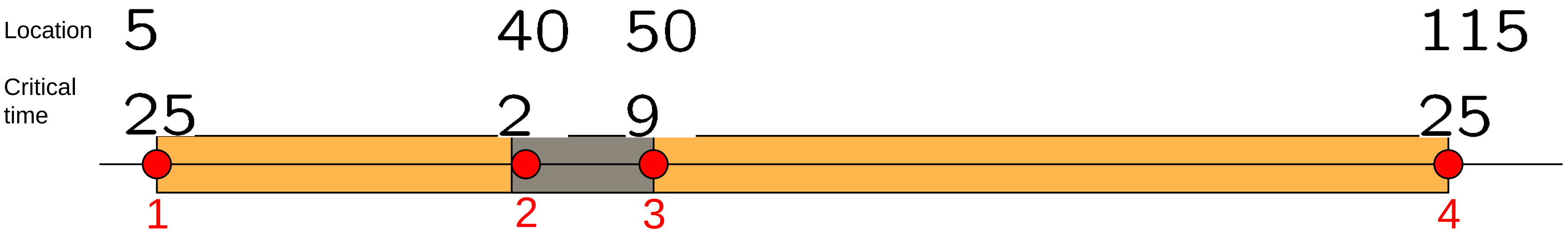
$$\Gamma_1 = \{1, 2\}$$
$$\Gamma_2 = \{3, 5\}$$
$$\Gamma_3 = \{4\}, \text{ Stationary}$$
$$\Gamma_4 = \{6, 7\}$$
$$\Gamma_5 = \{8, 9\}$$

Greedy algorithm for MS-BELP on a line

Sub-Optimality of the Greedy Algorithm



Sensor assignment by the greedy algorithm (v = 10m/s)



The optimal sensor assignment (v = 10m/s)

Here the OPT uses 2 sensors, while the greedy algorithm uses 3 sensors

MS-BELP: Unrestricted Motion

- Heuristic algorithm
  1. Calculate TSPN path for the set of PoIs
  2. Use greedy algorithm for closed curve to solve MS-BELP over the TSPN path
- If $k^\star$ is the optimal number of sensors, then

$$\frac{k}{k^\star} \le \frac{2T_{\mathrm{max}}}{T_{\mathrm{min}}} + \frac{2r_{\mathrm{max}}}{vT_{\mathrm{min}}}$$

- The performance ratio also depends on location of the PoIs

Talk Outline

- Analytical results: When is mobility useful?
- BELP Problem
- Algorithms for MV-BELP problem
  - Restricted motion case
  - Unrestricted motion case
- Algorithms for MS-BELP problem
  - Restricted motion case
  - Unrestricted motion case
- Summary and Future Works

Summary

- Characterized the scenarios where mobility improves the quality of coverage
- Formulate the bounded event loss probability (BELP) problem
- For restricted motion cases, we propose optimal and 2-approximate algorithms for MV-BELP and MS-BELP respectively
- For unrestricted motion cases, we propose heuristic algorithms and bound their performance with respect to the optimal

Future Work

- Develop approximate algorithms whose performance ratio is constant or depends on number of PoIs only

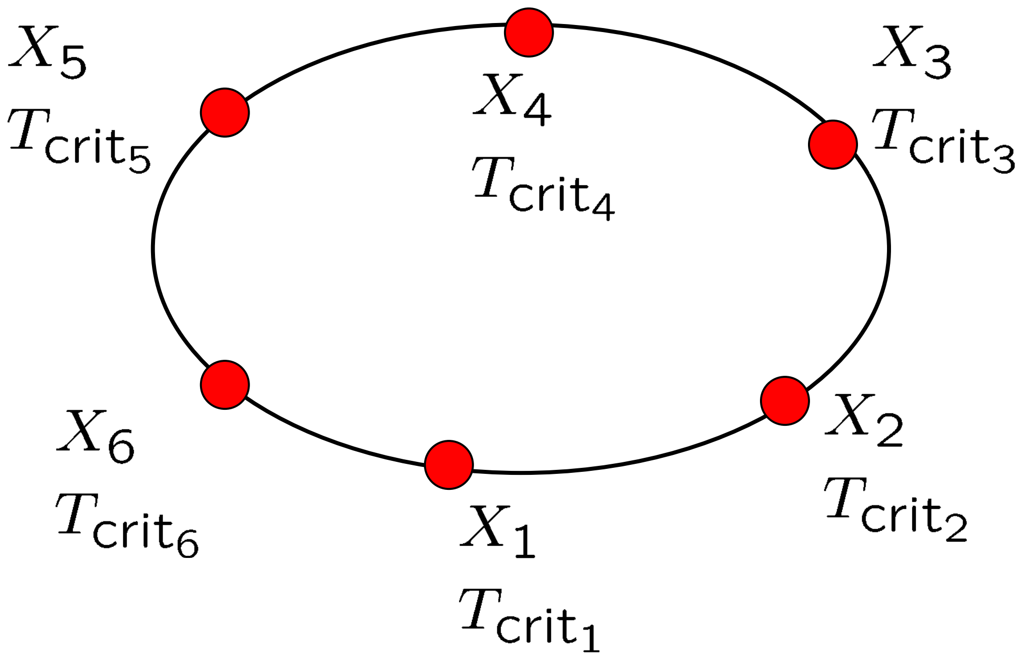
- Take communication requirements into accounts and develop path planning algorithms that satisfy communication constraints as well

Thank You

MV-BELP on a Curve

- Mobile sensor is restricted to move along a simple closed curve joining all PoIs

- Two Options
  - Sensor circles around the curve
  - Sensor moves to and fro between two neighboring nodes (n such cases)
- In all n+1 cases
- Minimum velocity for each case can be calculated

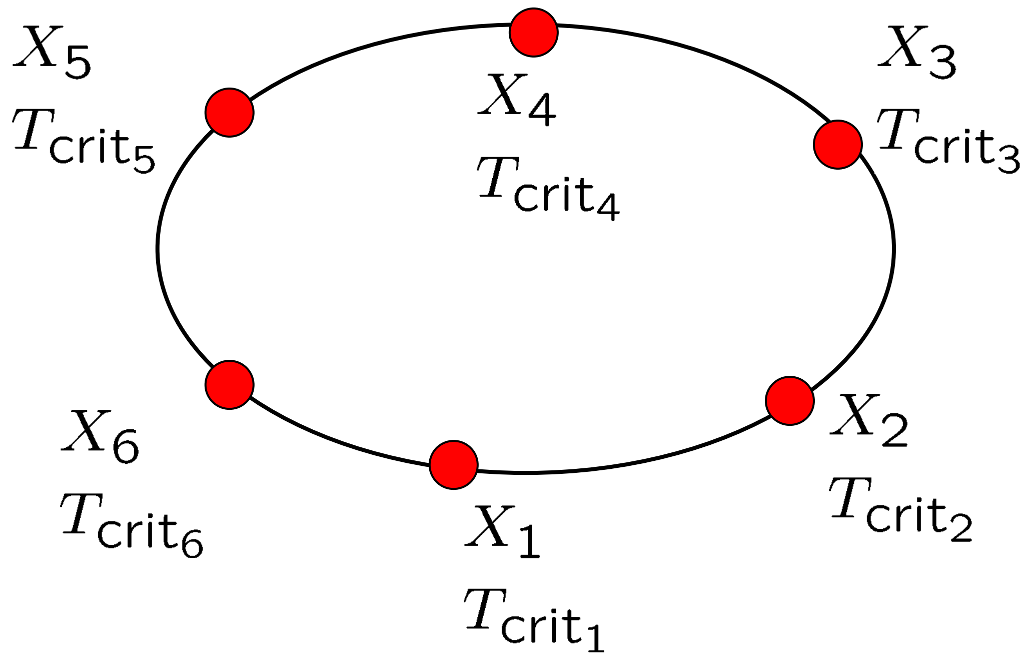


MV-BELP on a Curve

- Mobile sensor is restricted to move along a simple closed curve joining all PoIs

If sensor circles around, minimum velocity required:

$$v_0 = \frac{D}{\min_i T_{\mathrm{crit}_i}}$$

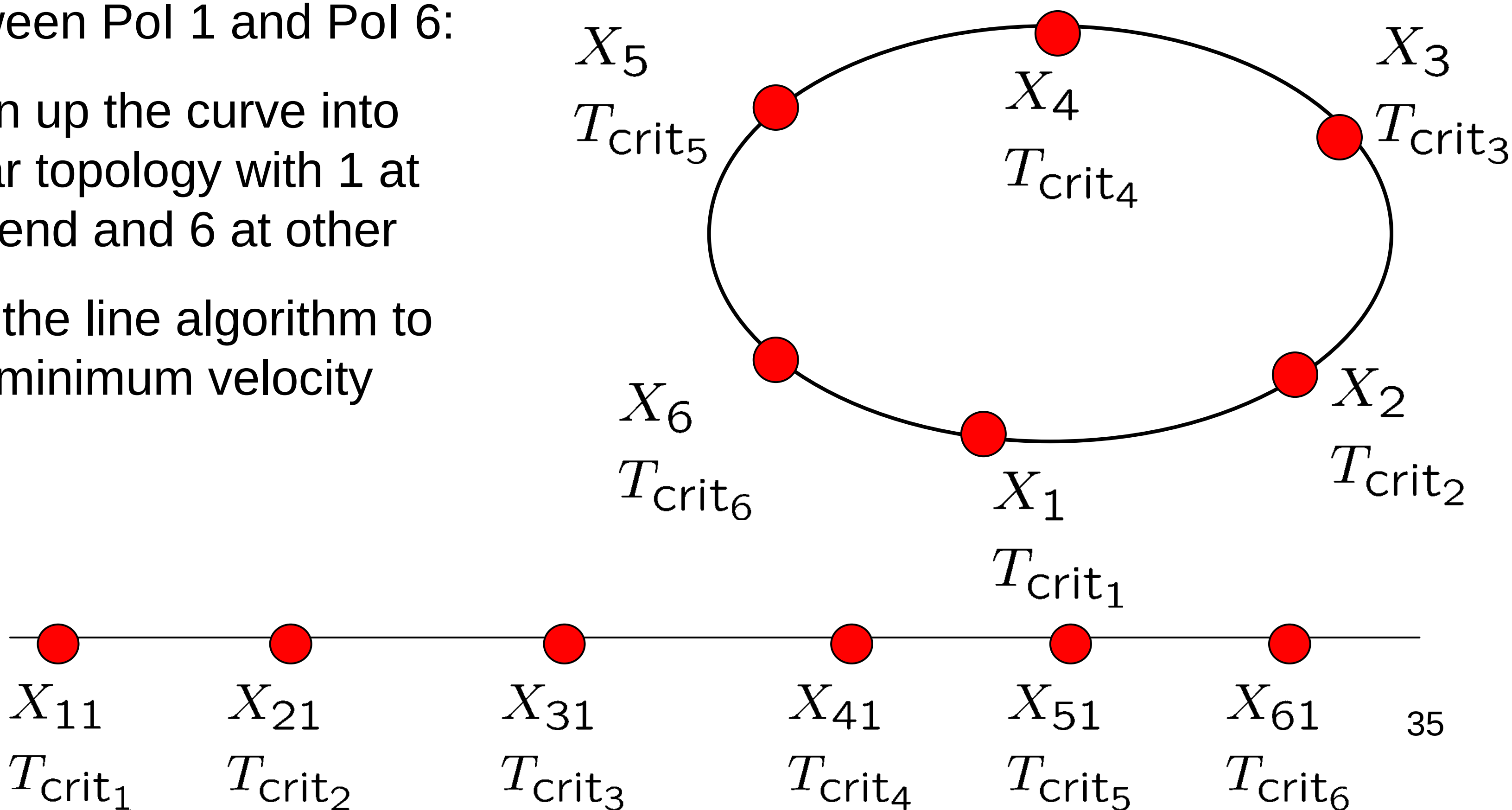


MV-BELP on a Curve

- Mobile sensor is restricted to move along a simple closed curve joining all PoIs

If sensor moves to and fro between PoI 1 and PoI 6:

1. Open up the curve into linear topology with 1 at one end and 6 at other
2. Use the line algorithm to find minimum velocity

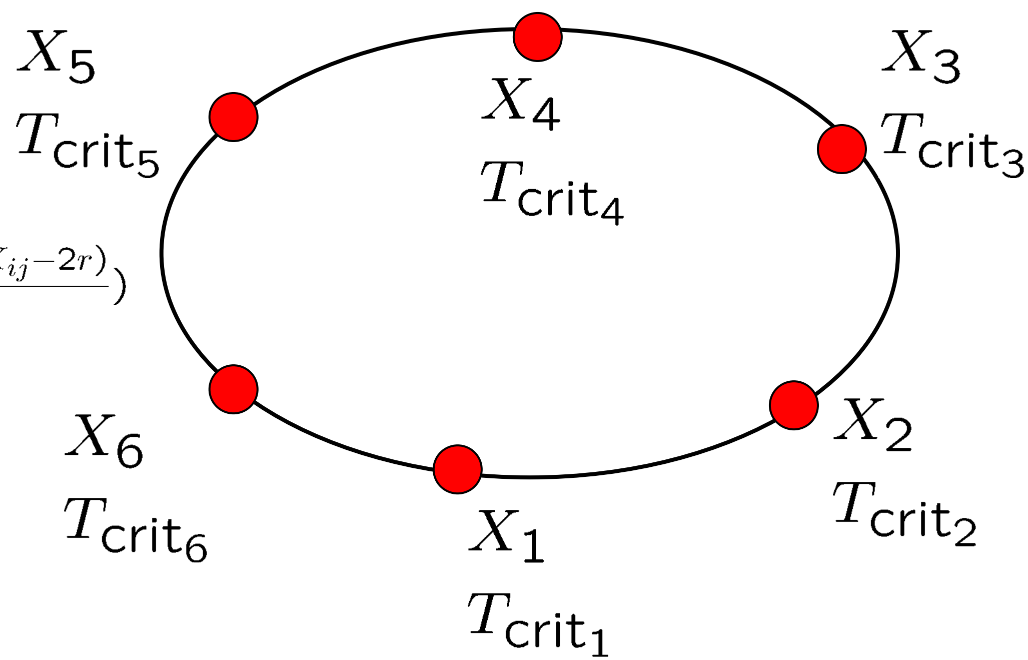


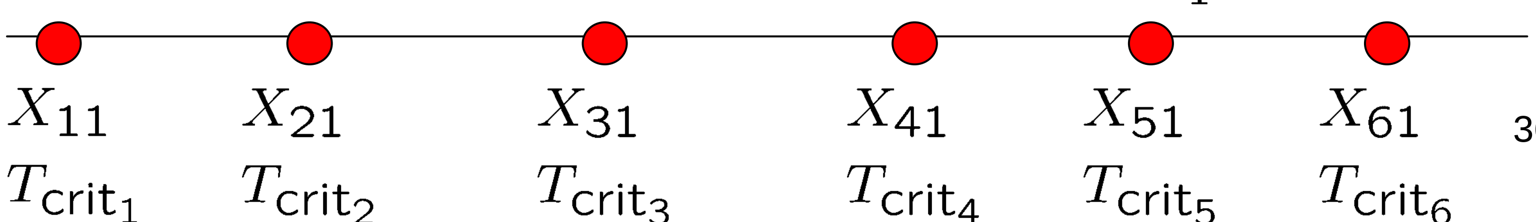
MV-BELP on a Curve

- Mobile sensor is restricted to move along a simple closed curve joining all PoIs

Minimum velocity required for to and fro motion between PoI and its neighbor:

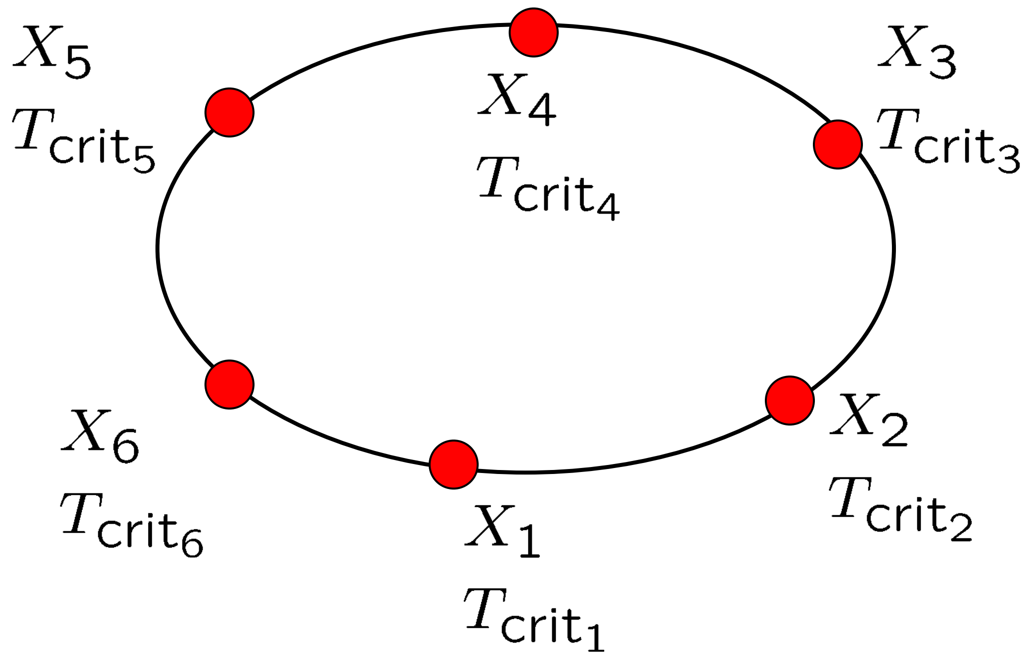
$$v_j = \max_i \max\left(\frac{2(X_{ij}-X_{ii}-2r)}{T_{\text{crit}_i}}, \frac{2(X_{\text{mod}_n(j+n)j}-X_{ij}-2r)}{T_{\text{crit}_i}}\right)$$





MV-BELP on a Curve

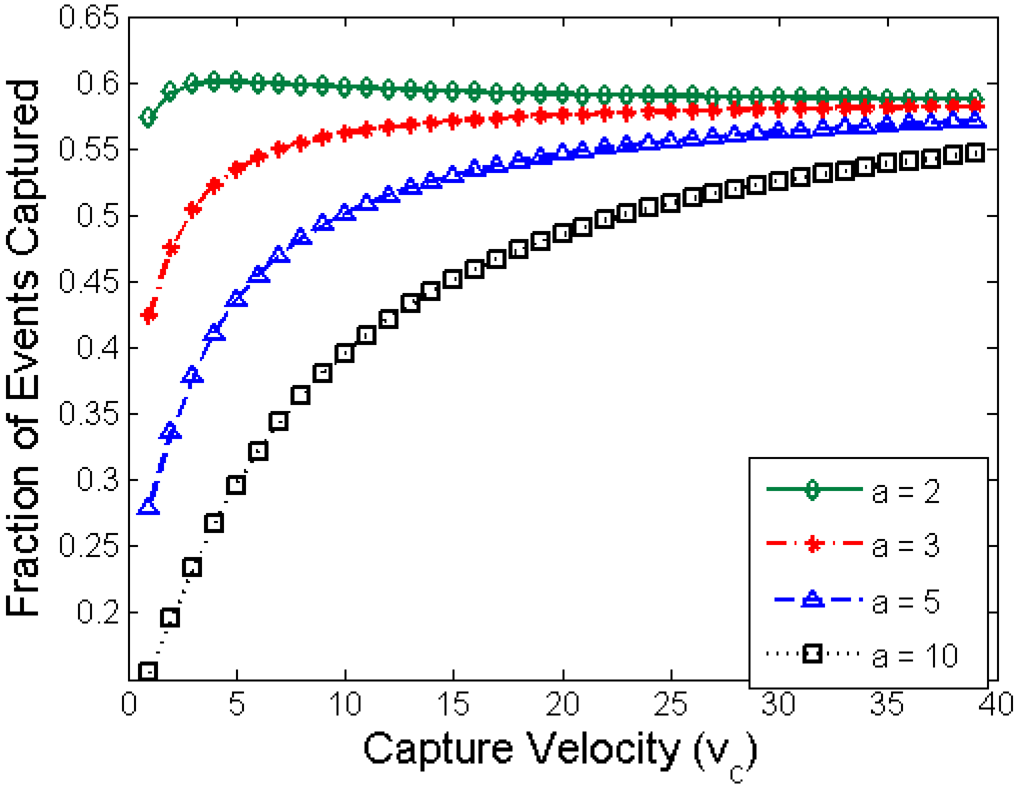
- Mobile sensor is restricted to move along a simple closed curve joining all PoIs



Minimum velocity required for to and fro motion between PoI and its neighbor:

$$v_{\min} = \min_{0 \le j \le n} v_j$$

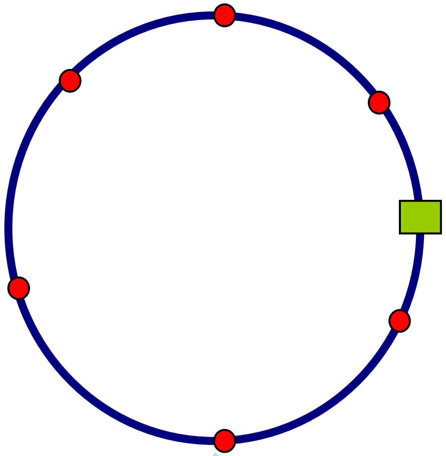
Variable Velocity Case



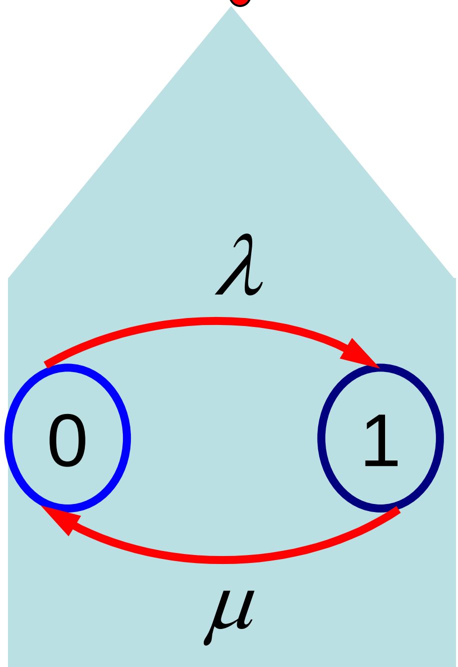
Slowing down during a visit, in order to spend more fraction of time observing the PoIs does not help either

The solution therefore is to choose "good" paths to move along

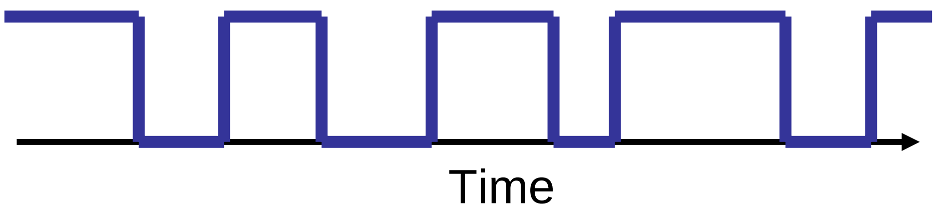
The Event Model



- The PoIs have states 0 and 1
- State 1 corresponds to event to be "captured"
- The time spent in each state is exponentially distributed with means $1/\lambda$ and $1/\mu$

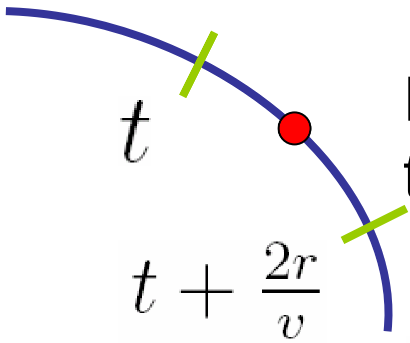


The states of PoIs may be represented as a Markov chain



The state vs. time plot

Analysis



Each time the sensor "visits" a PoI it observes the point for time 2r/v

$S_i(t)$ = state of PoI i at time t

$N_i(t, t + \frac{2r}{v})$ = Total number of distinct events detected in a visit to PoI i

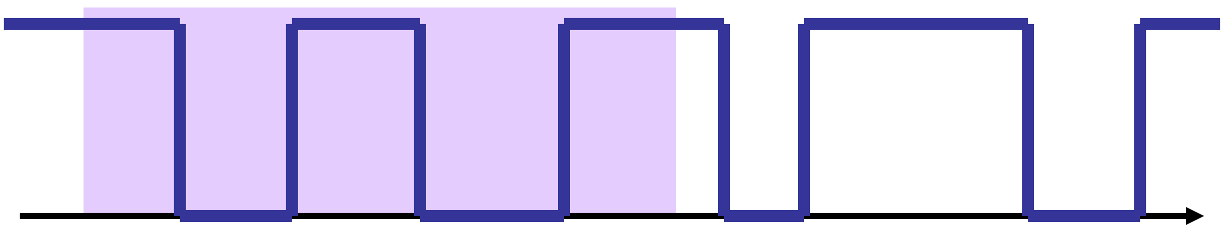
$$P[S_i(t) = 1] = \frac{\lambda}{\mu + \lambda} \qquad P[S_i(t) = 0] = \frac{\mu}{\mu + \lambda}$$

$$E[N_i(t, t + \frac{2r}{v})] = P[S_i(t) = 1] \times E[N_i(t, t + \frac{2r}{v}) | S_i(t) = 1] + P[S_i(t) = 0] \times E[N_i(t, t + \frac{2r}{v}) | S_i(t) = 0]$$

Suppose that the sensor starts observing a PoI when its state is 1, then

$$N_i(t, t+\frac{2r}{v}) = 1 + C(\frac{2r}{v})$$

Where C(t) = number of 1 => 0 => 1 cycles in time t



Since expected duration of one cycle is $\frac{1}{\lambda} + \frac{1}{\mu}$ expected number of cycles in time τ equals

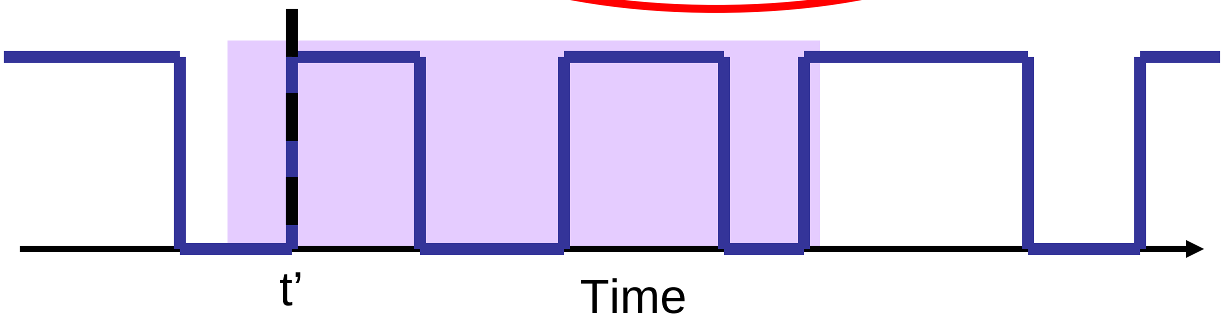
$$E[C(\tau)] = \frac{\lambda\mu}{\lambda+\mu}(\tau - \frac{1}{\lambda+\mu}(1 - e^{-(\lambda+\mu)\tau}))$$

So expected number of distinct events captured, given state of the point was one when the sensor arrived equals

$$E[N_i(t, t+\frac{2r}{v}) \mid S_i(t) = 1] = 1 - e^{-\mu(\frac{D-2r}{v})} + E[C(\frac{2r}{v})] = 1 + \frac{\lambda\mu}{\lambda+\mu}\frac{2r}{v}$$

Now suppose that the sensor starts observing a PoI when its state is 0, then

$$E[N_i(t, t+\frac{2r}{v})|S_i(t)=0] = \int_{t'=t}^{t'=t+\frac{2r}{v}} P[S_i(t')=0, S_i(t'+\mathrm{d}t')=1] \cdot \left(E[N_i(t', t+\frac{2r}{v})|S_i(t')=1]\right)$$



1st Term: Probability that state flips from 0 to 1 at t', t < t' < t+2r/v

2nd Term: Expected number events captured between t' and t+2r/v given state at t' is 1, already known

$$E[N_i(t, t+\frac{2r}{v})|S_i(t)=0] = \left(1-\exp\left(-\lambda\frac{2r}{v}\right)\right)\left[1+\frac{\lambda\mu}{\lambda+\mu}\right] - \frac{\lambda\mu}{\lambda+\mu}\left[\frac{1}{\lambda} - \left(\frac{2r}{v}+\frac{1}{\lambda}\right)\exp\left(-\lambda\frac{2r}{v}\right)\right]$$

Now that $E[N_i(t,t+\frac{2r}{v}) \mid S_i(t)=1]$ and $E[N_i(t,t+\frac{2r}{v}) \mid S_i(t)=0]$ are known $E[N_i(t,t+\frac{2r}{v})]$ can be determined

Let T_∞ be a large time duration, N'_{T_∞} be the number of events captured by the sensor and N_{T_∞} be the total number of events that occur, then

$$N'_{T_\infty} = \frac{vT_\infty}{D} \cdot k \cdot E[N_i(t,t+\frac{2r}{v})]$$

$$N_{T_\infty} = \frac{T_\infty}{1/\lambda + 1/\mu} \cdot k = \frac{\lambda\mu T_\infty}{\lambda+\mu} \cdot k$$

Therefore the fraction of events captured by the sensor equals

$$\frac{N'_{T_\infty}}{N_{T_\infty}} = \frac{v(\lambda+\mu)}{\lambda\mu D} E[N_i(t,t+\frac{2r}{v})]$$

Variable Velocity Case

- Suppose the sensor can move at all velocities between 0 and v_{max}
- How should sensor adjust its speed during the journey
- Move with v_{max} when no PoI visible
- With what speed to move when it sees a PoI
  - Too small => miss events at other PoIs
  - Too large => miss potential events at this PoI
- What is the optimal speed to move with during a visit?