Resource Conservation in Sensor Networks

- Byers, J., and Nasser, G. "Utility-Based Decision-Making in Wireless Sensor Networks". Proceedings of IEEE MobiHOC 2000, Boston, MA, August 2000.
- Zhao, F., Shin, J., and Reich, J. "Information-Driven Dynamic Sensor Collaboration for Tracking Applications". Proceedings of IEEE Signal Processing Magazine, March 2002.

Overview

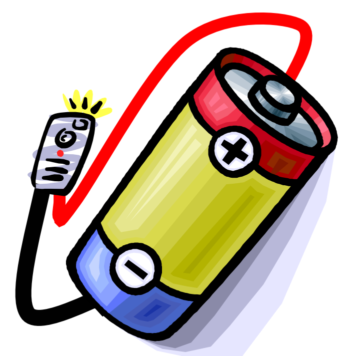
- Motivation
- Byers 2000
  - Cost-Utility Model
  - Algorithm
  - Experimental Results
- Zhao 2002
  - IDSQ Model
  - Algorithm
  - Experimental Results

Overview

- Motivation
- Byers 2000
  - Cost-Utility Model
  - Algorithm
  - Experimental Results
- Zhao 2002
  - IDSQ Model
  - Algorithm
  - Experimental Results

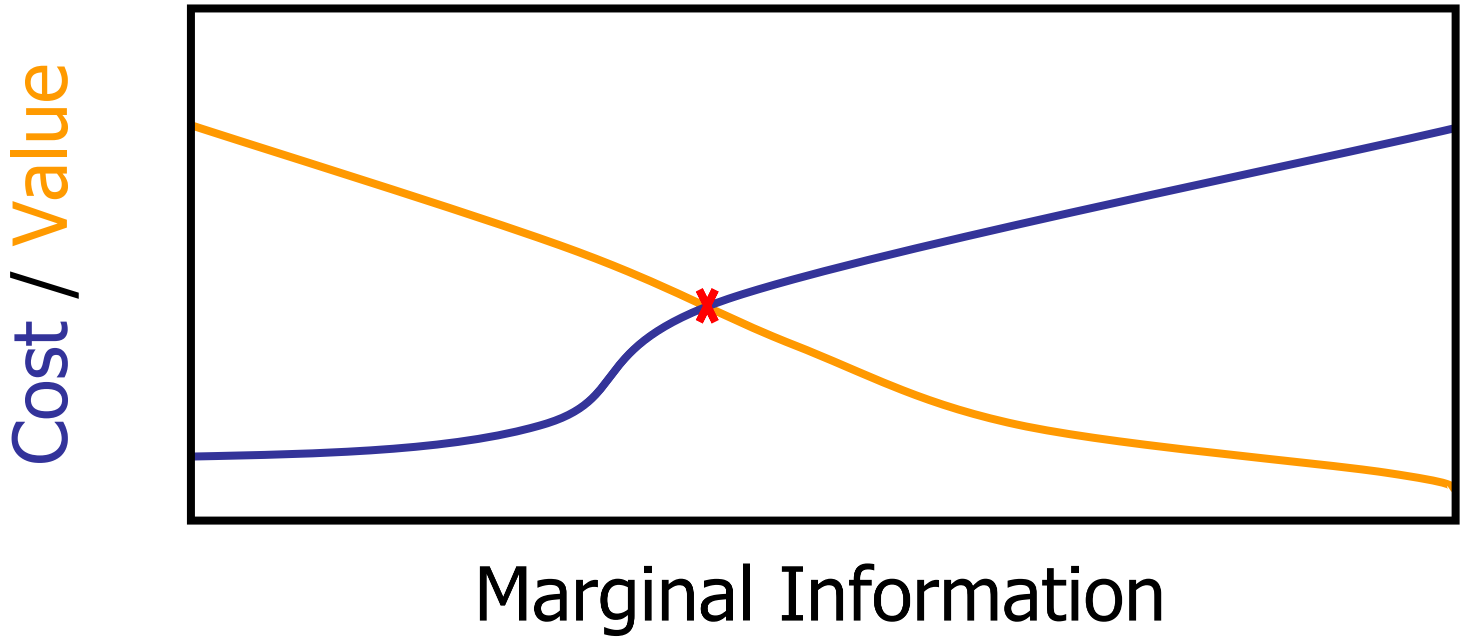
Power & Sensor Conservation

- Sensors have limited power
  - Reliant on non-renewable batteries
  - Battery technology not improving
- Conserving power
  - Increases sensor life
  - Makes sensor smaller
  - Decreases per sensor cost
- Also, minimize sensor count



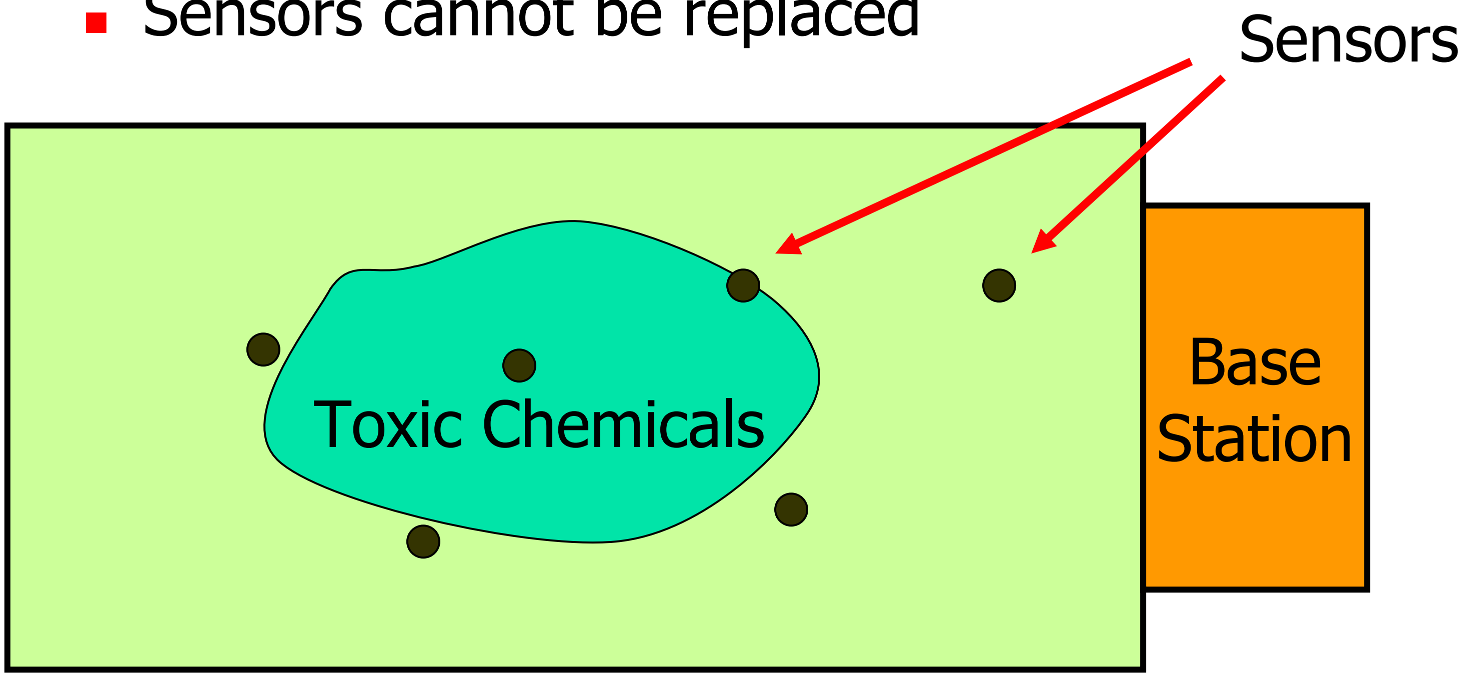
Information Utility

- Information has diminishing marginal returns



Example: Monitoring Toxicity

- Assumptions:
  - Sensors connected to base station
  - Sensors cannot be replaced



Overview

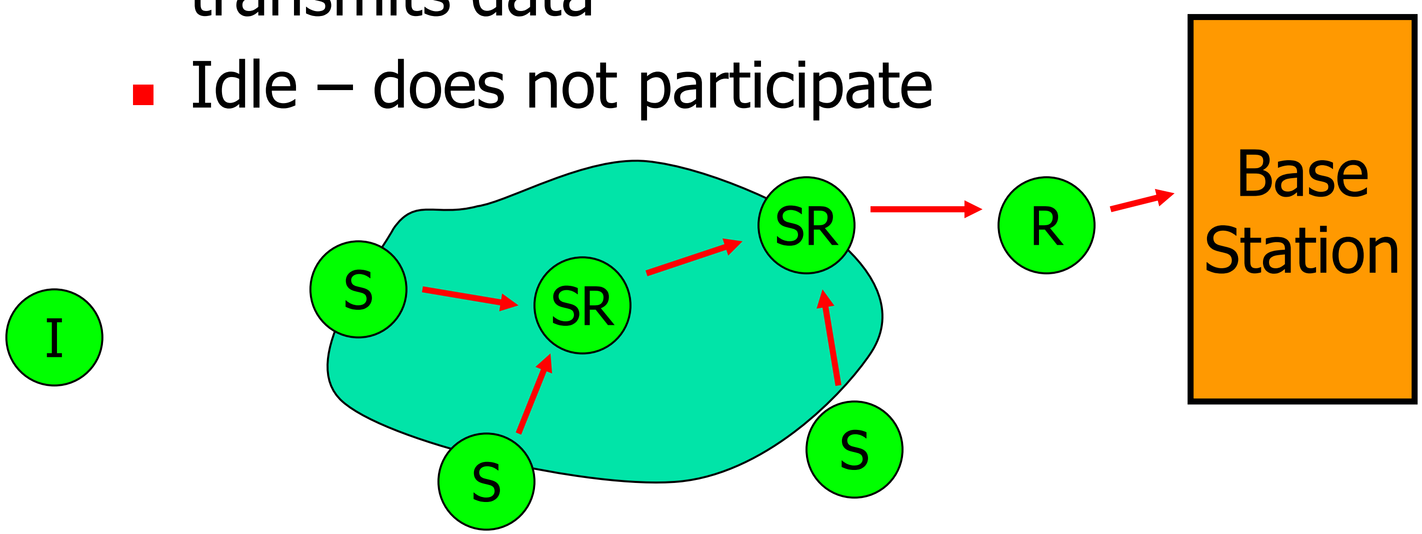
- Motivation
- Byers 2000
  - Cost-Utility Model
  - Algorithm
  - Experimental Results
- Zhao 2002
  - IDSQ Model
  - Algorithm
  - Experimental Results

Network Model

- Assumptions:
  - Isotropic sensor transmission with fixed range R
  - No communication medium conflict
- Let G = (V,E)
  - V = Set of all operational sensors
  - E = Communication links between sensors
- An edge exists when the physical distance between a pair of nodes is below R

Sensing Model

- Nodes can operate in one of four modes:
  - Only Routing – receives and transmits data
  - Only Sensing – senses and transmits data
  - Both Routing and Sensing – receives, senses, and transmits data
  - Idle – does not participate

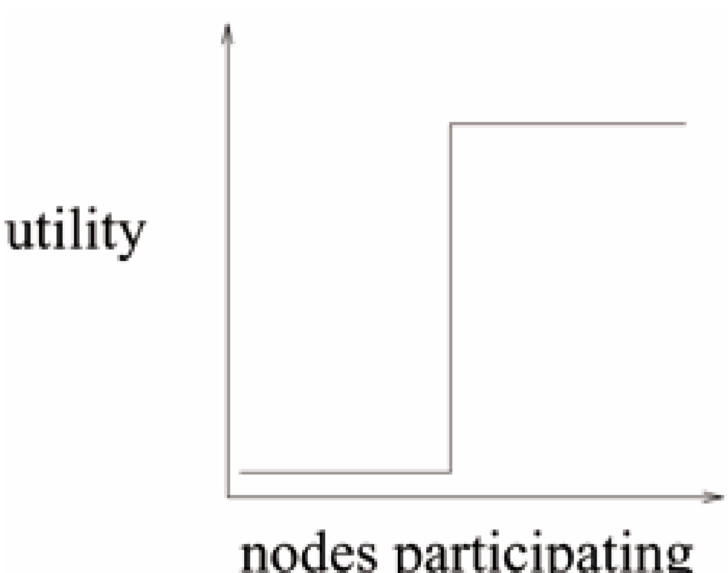


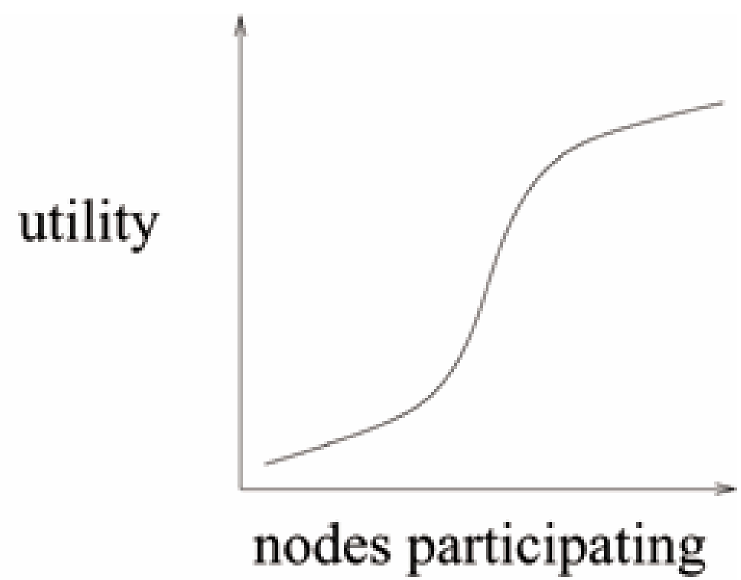
Data Aggregation

- Want to perform lossy compression of data to save communication costs
- Most useful when:
  - Information is of little value (e.g. little change)
  - Sensors provide redundant information
  - Large tree depth

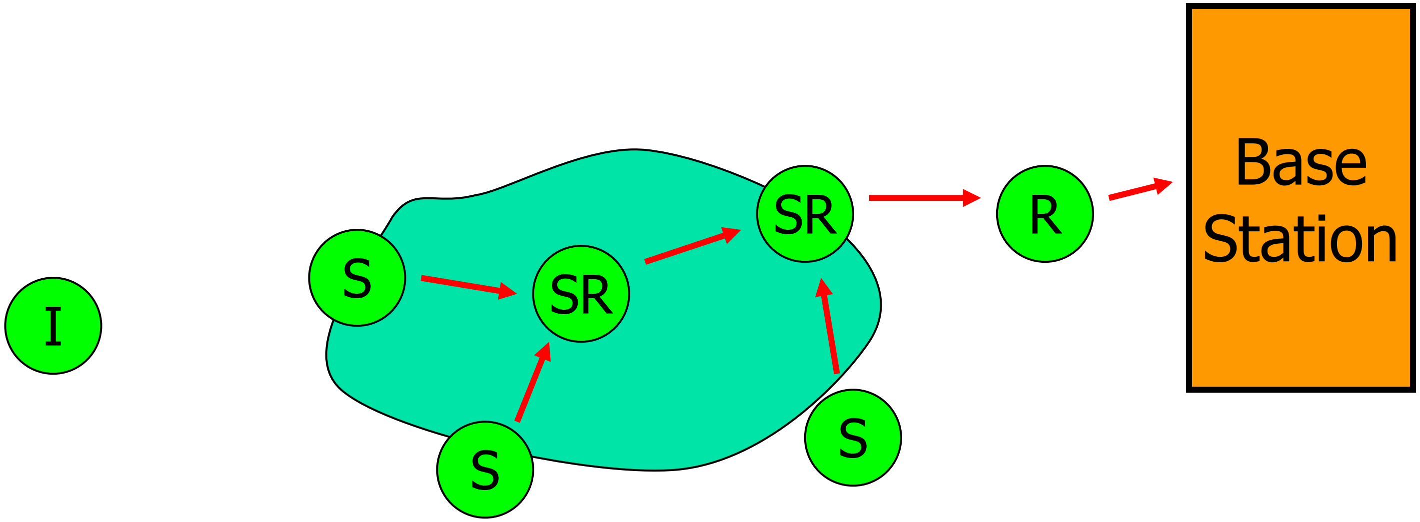
Information Utility

- Let S be the sensors that are chosen to sense
- Model utility based on S such that there exists a mapping U: $S^* \rightarrow [0,1]$
- Possible utilities:



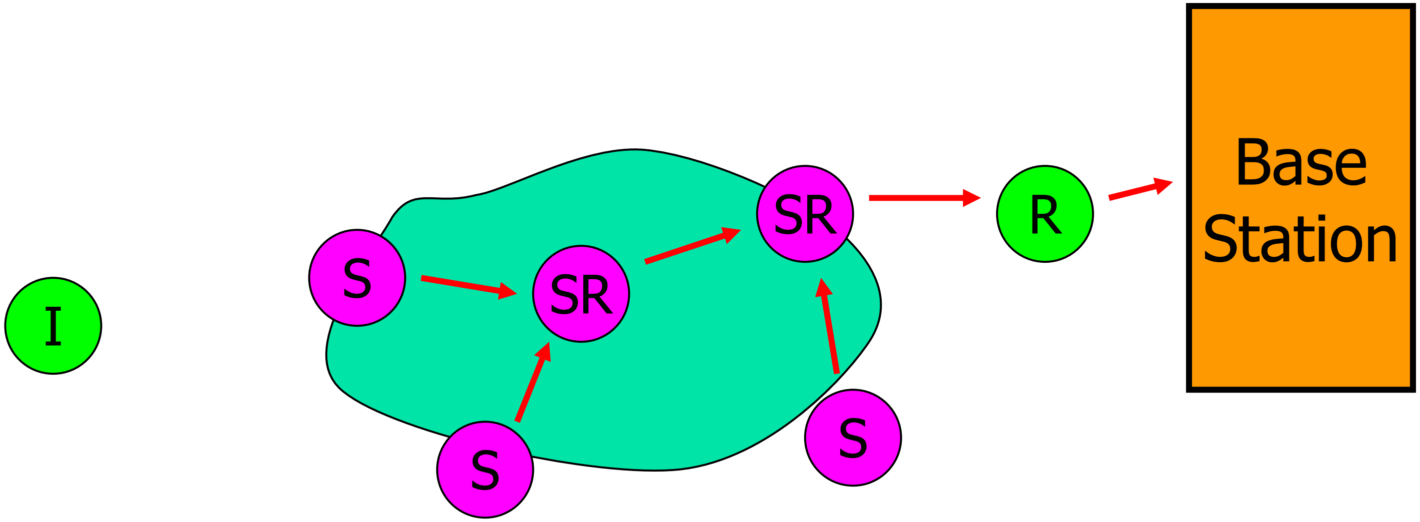


Costs



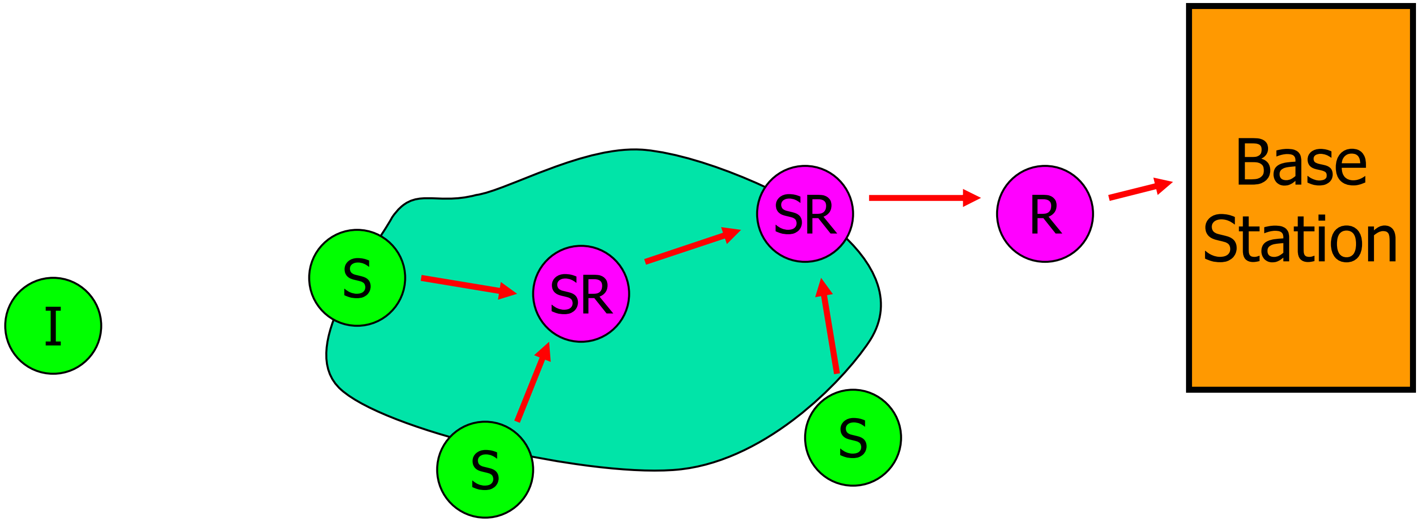
- S_t is the set of sensors sensing at time t
- $U(S_t)$ = utility from the sensors active at time t
- R_t is the set of sensors routing at time t
- c_s = sensing cost, c_t = transmitting cost, c_r = receiving cost, and c_a is aggregation cost

Costs



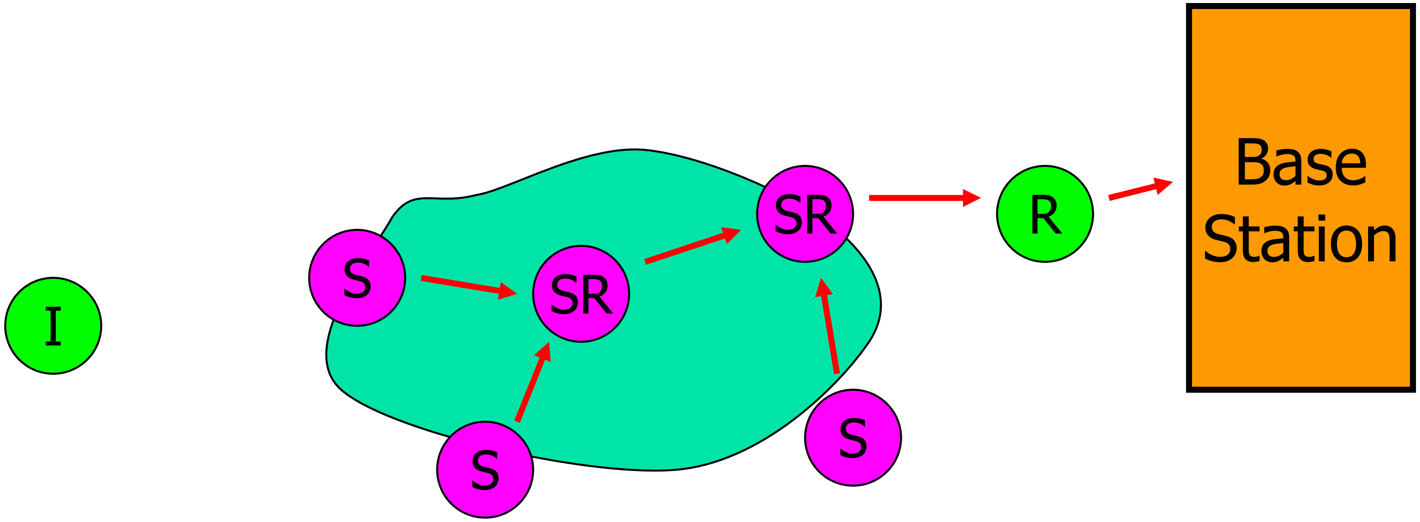
- S_t is the set of sensors sensing at time t
- $U(S_t)$ = utility from the sensors active at time t
- R_t is the set of sensors routing at time t
- c_s = sensing cost, c_t = transmitting cost, c_r = receiving cost, and c_a is aggregation cost

Costs



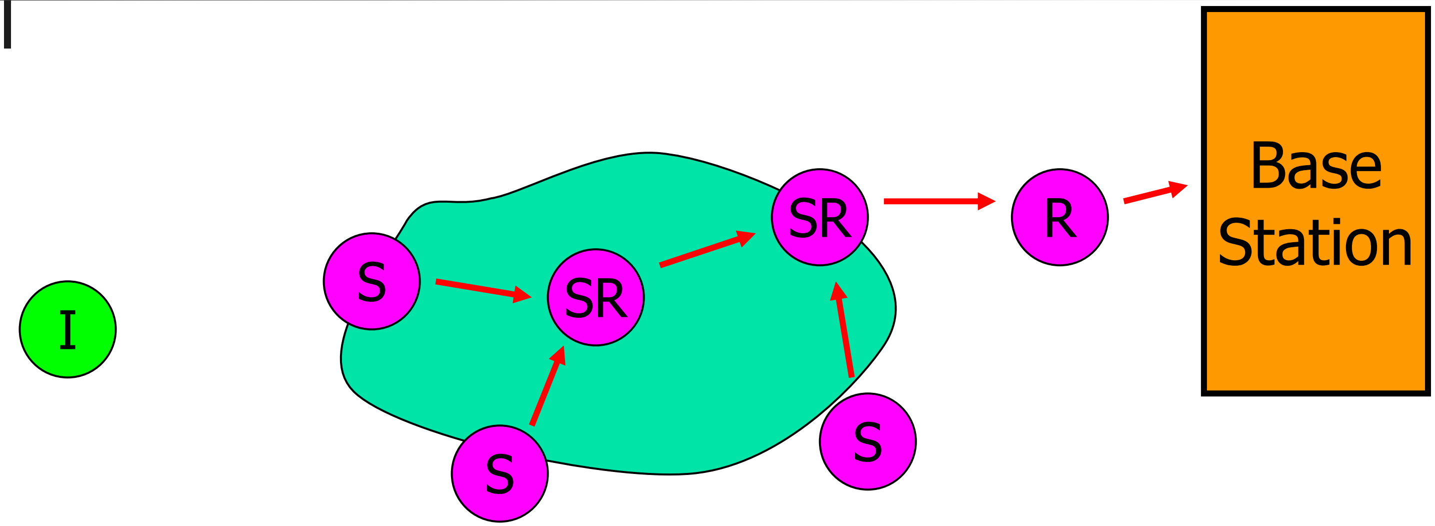
- S_t is the set of sensors sensing at time t
- $U(S_t)$ = utility from the sensors active at time t
- R_t is the set of sensors routing at time t
- c_s = sensing cost, c_t = transmitting cost, c_r = receiving cost, and c_a is aggregation cost

Costs



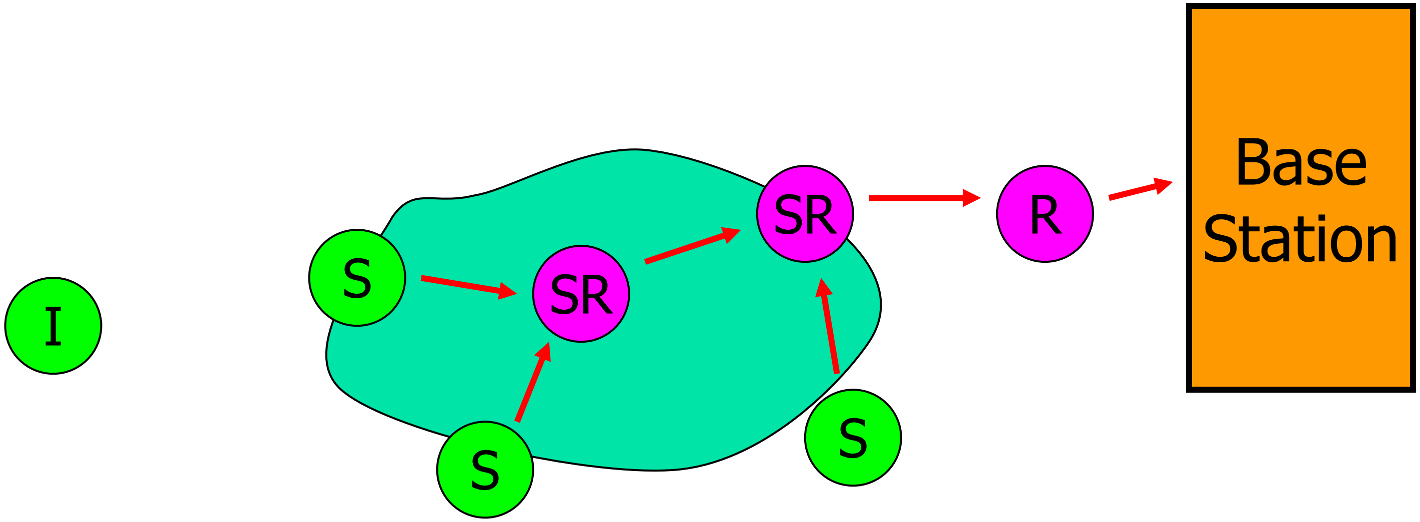
- S_t is the set of sensors sensing at time t
- $U(S_t)$ = utility from the sensors active at time t
- R_t is the set of sensors routing at time t
- c_s = sensing cost, c_t = transmitting cost, c_r = receiving cost, and c_a is aggregation cost

Costs



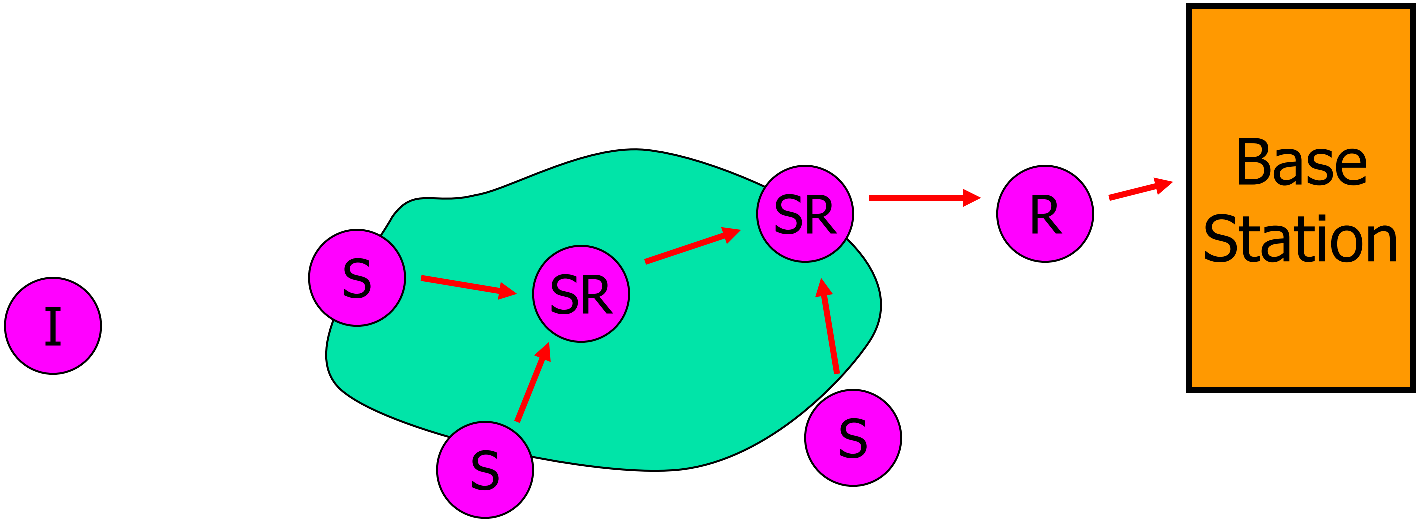
- S_t is the set of sensors sensing at time t
- $U(S_t)$ = utility from the sensors active at time t
- R_t is the set of sensors routing at time t
- c_s = sensing cost, c_t = transmitting cost, c_r = receiving cost, and c_a is aggregation cost

Costs



- S_t is the set of sensors sensing at time t
- $U(S_t)$ = utility from the sensors active at time t
- R_t is the set of sensors routing at time t
- c_s = sensing cost, c_t = transmitting cost, c_r = receiving cost, and c_a is aggregation cost

Costs



- S_t is the set of sensors sensing at time t
- $U(S_t)$ = utility from the sensors active at time t
- R_t is the set of sensors routing at time t
- c_s = sensing cost, c_t = transmitting cost, c_r = receiving cost, and c_a is aggregation cost

Objective Function

- Objective: maximize utility across all time
- Subject to:

  for each node across all time,

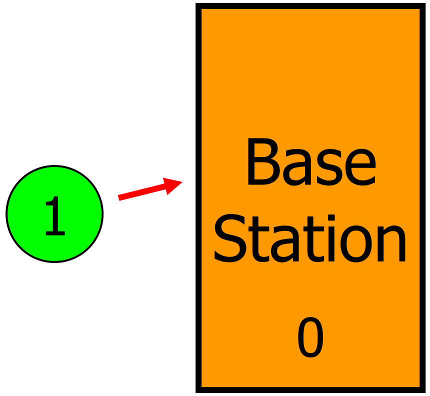
  all costs incurred by node $\leq$ power at node
- c_s = sensing cost
- c_t = transmitting cost
- c_r = receiving cost
- c_a is aggregation cost

Overview

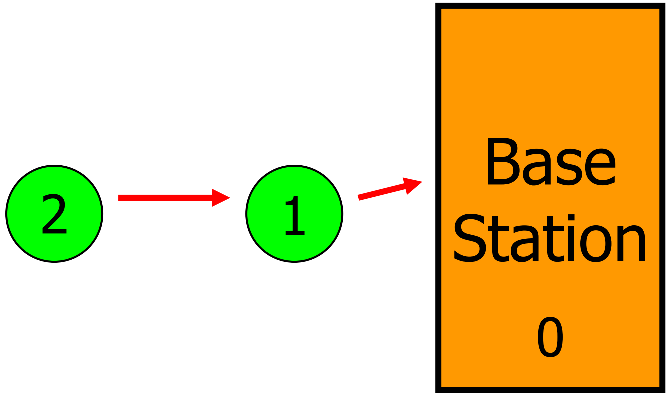
- Motivation
- Byers 2000
  - Cost-Utility Model
  - Algorithm
  - Experimental Results
- Zhao 2002
  - IDSQ Model
  - Algorithm
  - Experimental Results

Network Initialization

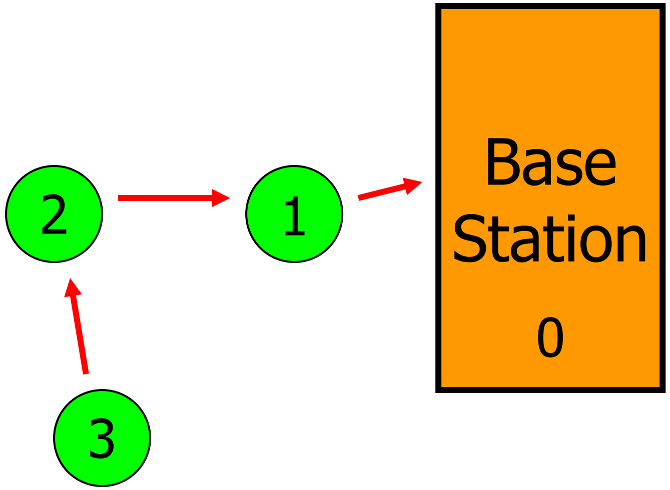
Network Initialization



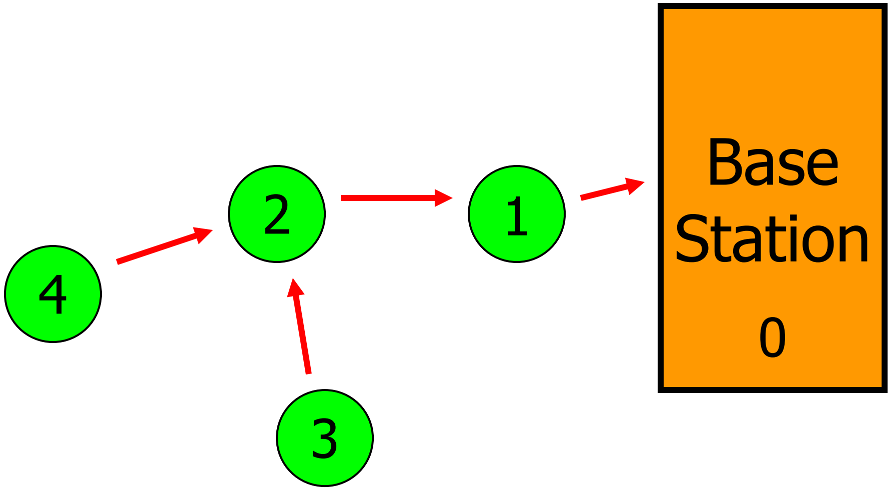
Network Initialization



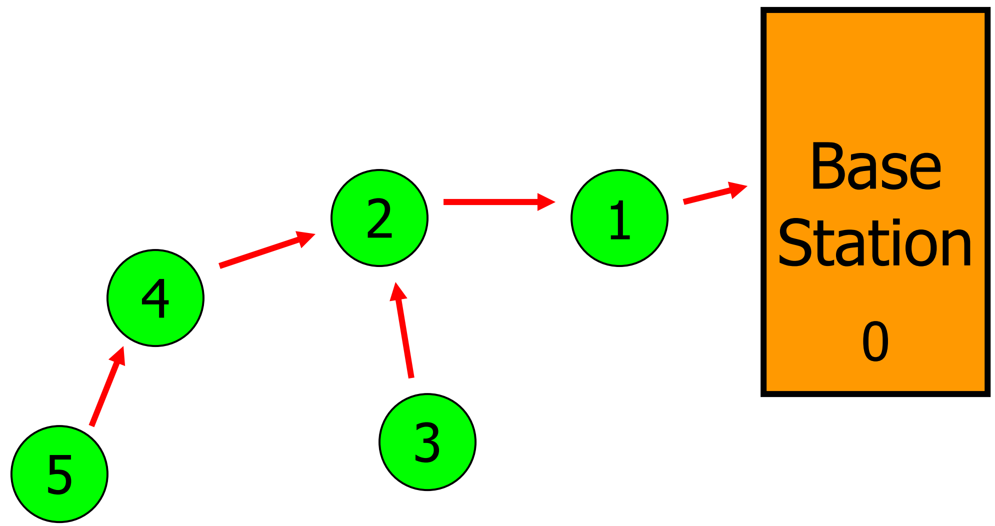
Network Initialization



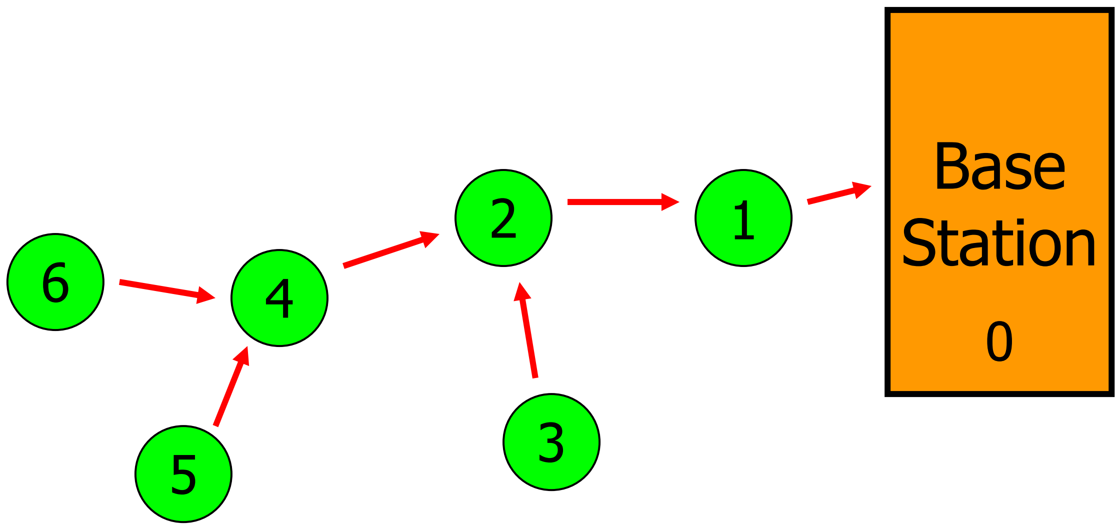
Network Initialization



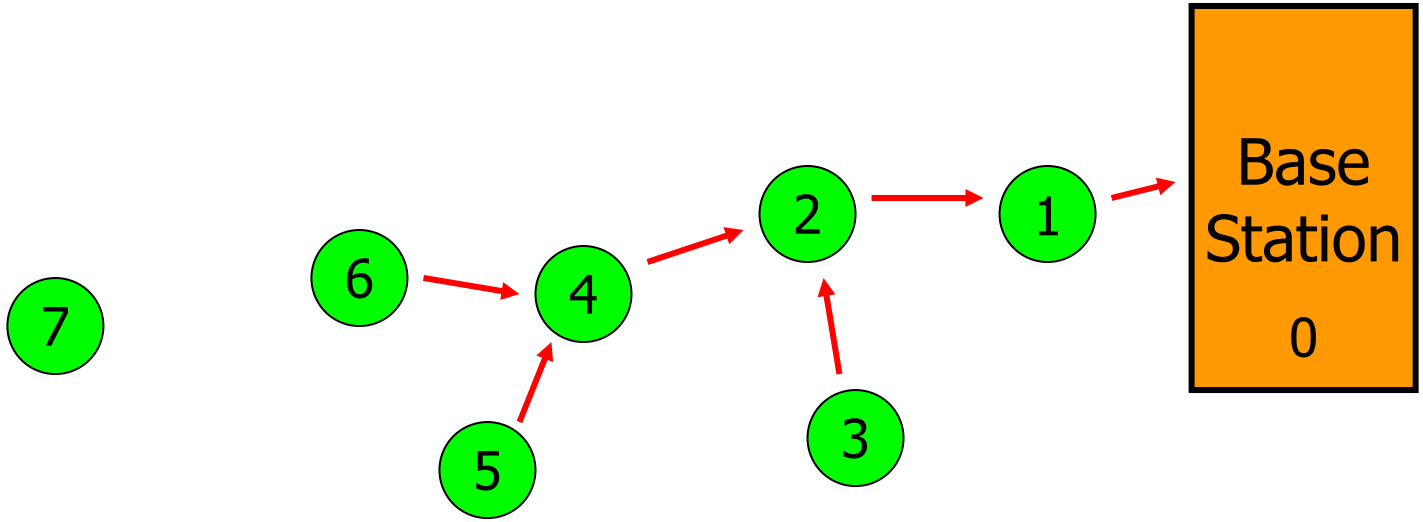
Network Initialization



Network Initialization

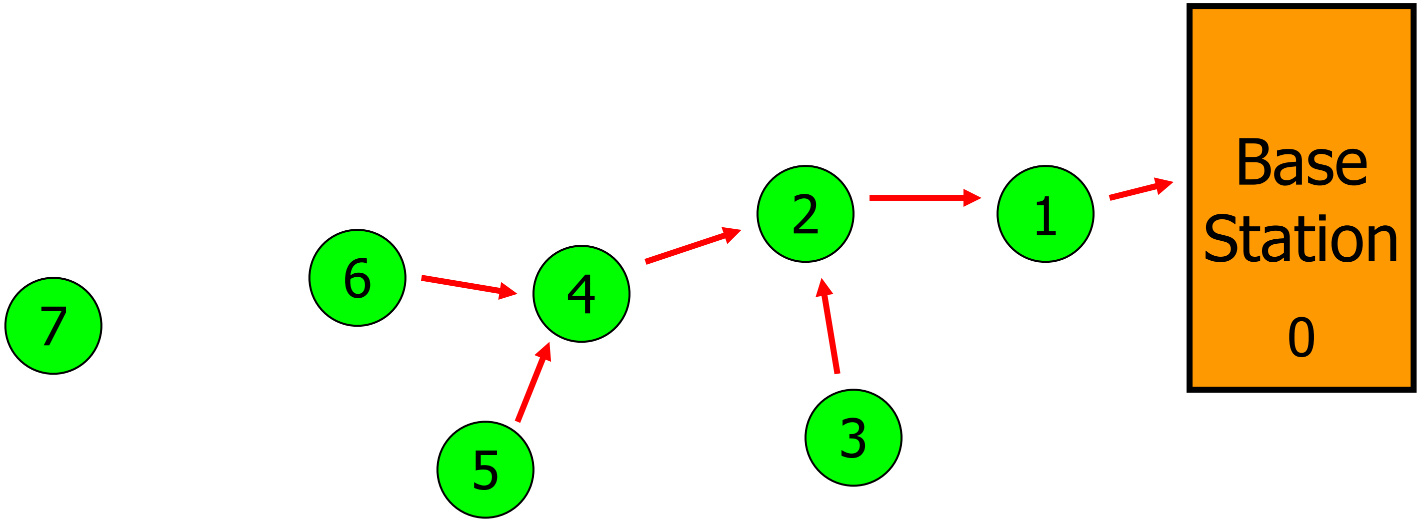


Network Initialization



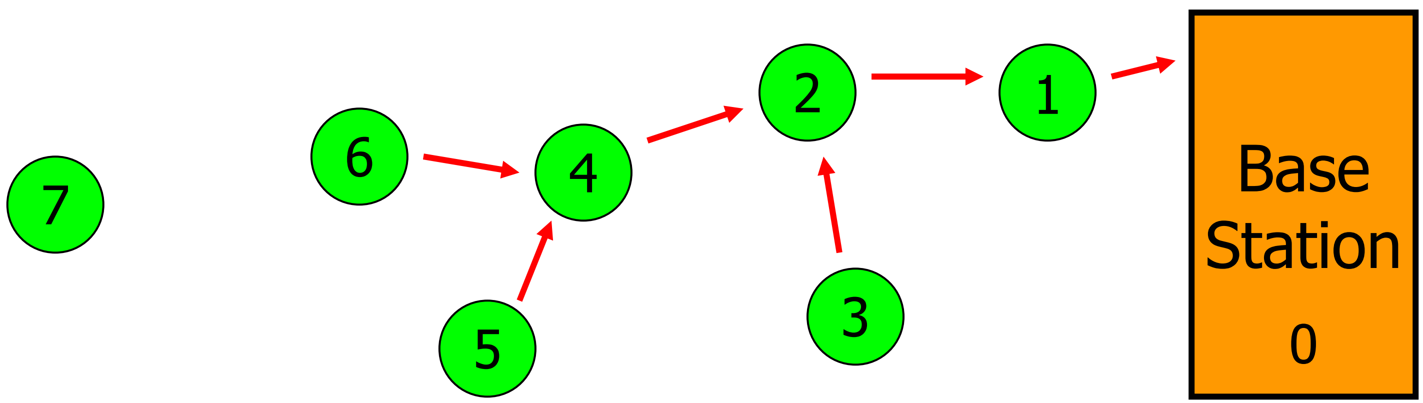
Adaptive Routing Algorithm

- Base station decides that N nodes must sense
- Variant of N Participants (Information Theory)
- Invariant: sense only if all children sensing



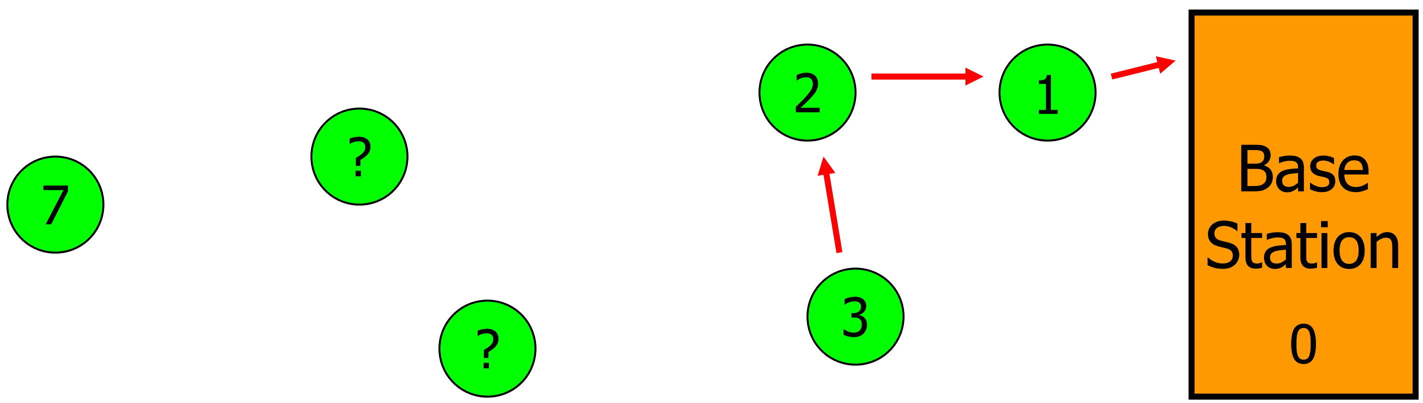
Inheriting Orphaned Subtrees

- Leaf failure is rectified via N Participants
- If routing node fails:
  - Each node independently emits search ping
  - Connected nodes within distance R become prospective parents
  - Orphans choose parent at minimum depth in tree



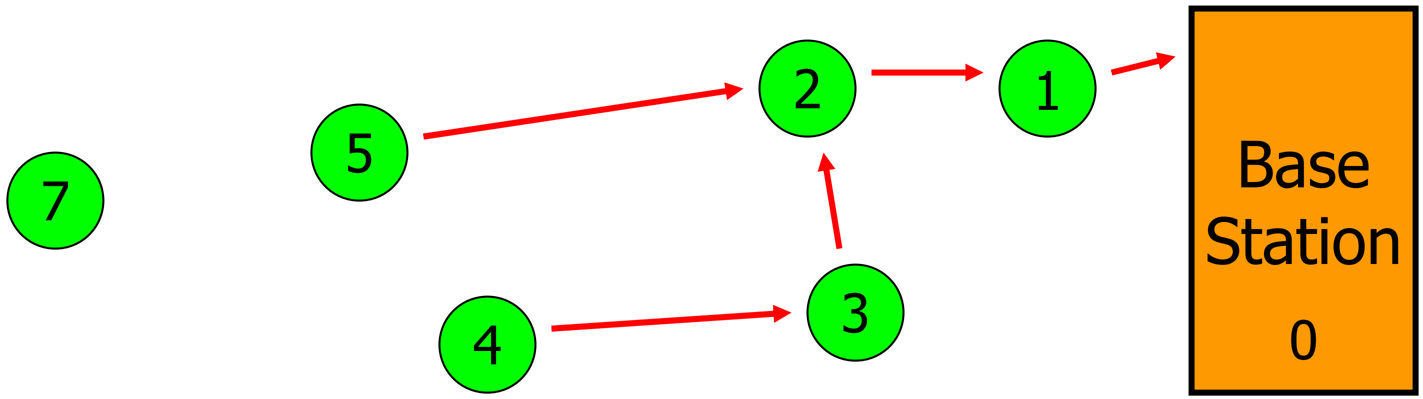
Inheriting Orphaned Subtrees

- Leaf failure is rectified via N Participants
- If routing node fails:
  - Each node independently emits search ping
  - Connected nodes within distance R become prospective parents
  - Orphans choose parent at minimum depth in tree



Inheriting Orphaned Subtrees

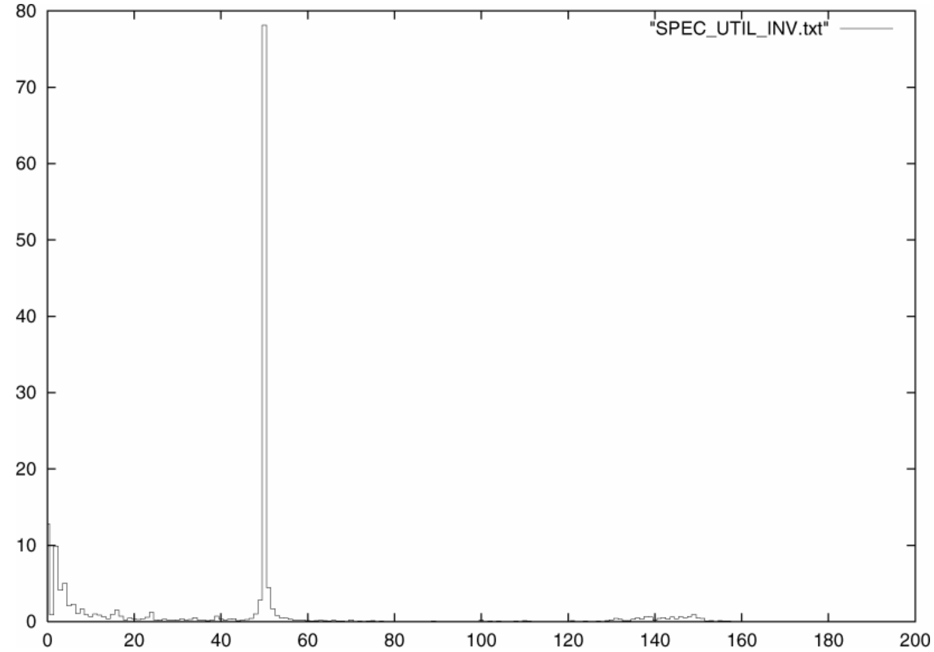
- Leaf failure is rectified via N Participants
- If routing node fails:
  - Each node independently emits search ping
  - Connected nodes within distance R become prospective parents
  - Orphans choose parent at minimum depth in tree



Overview

- Motivation
- Byers 2000
  - Cost-Utility Model
  - Algorithm
  - Experimental Results
- Zhao 2002
  - IDSQ Model
  - Algorithm
  - Experimental Results

Simulated Results

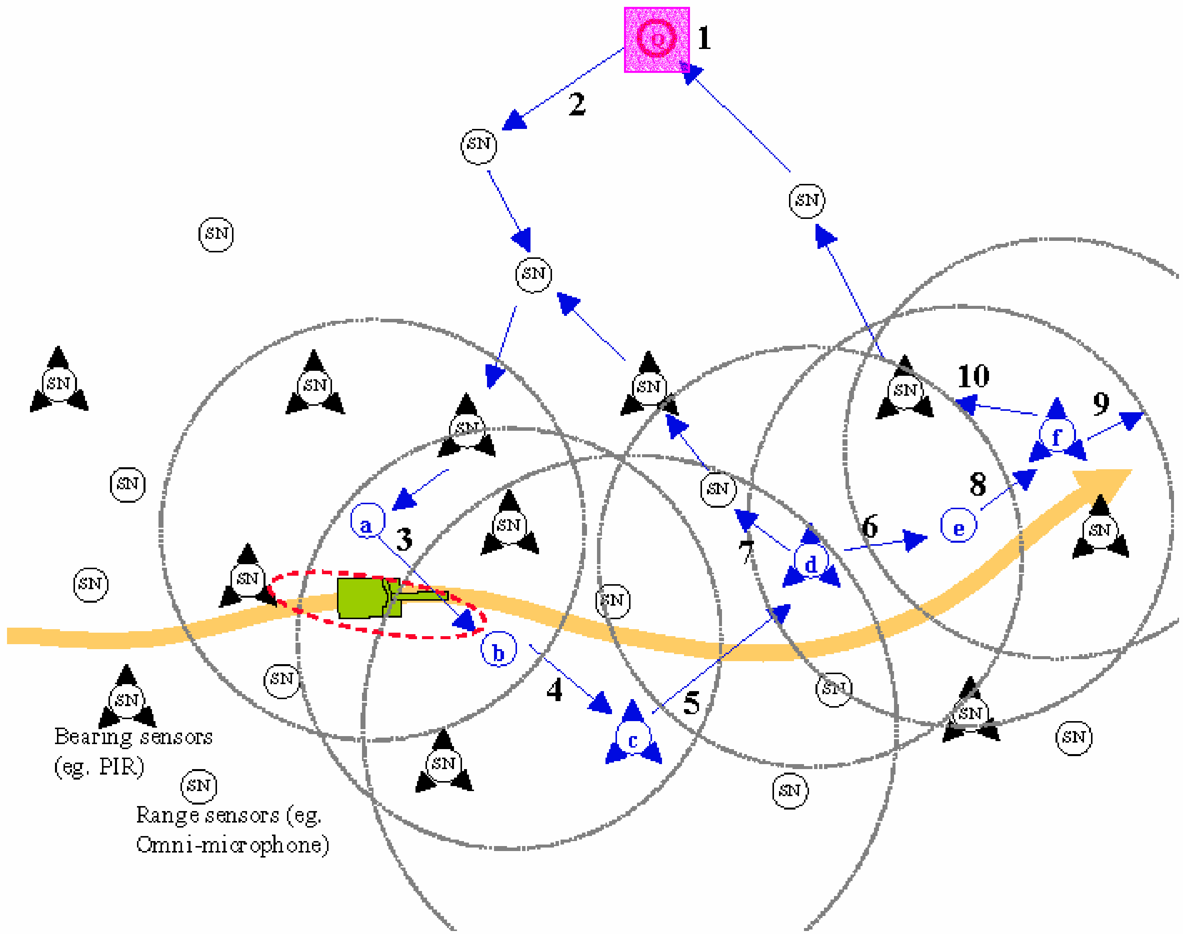


Heuristic	Lifetime	Total Utility	Total Energy Consumed	Utility/Energy
NAIVE	92.98	62.3	22039.3	.002827
SIMPLE_UTIL	349.633	64.855	15418.01	.004206
SPEC_INV	216	104.87	14505.54	.00723

Overview

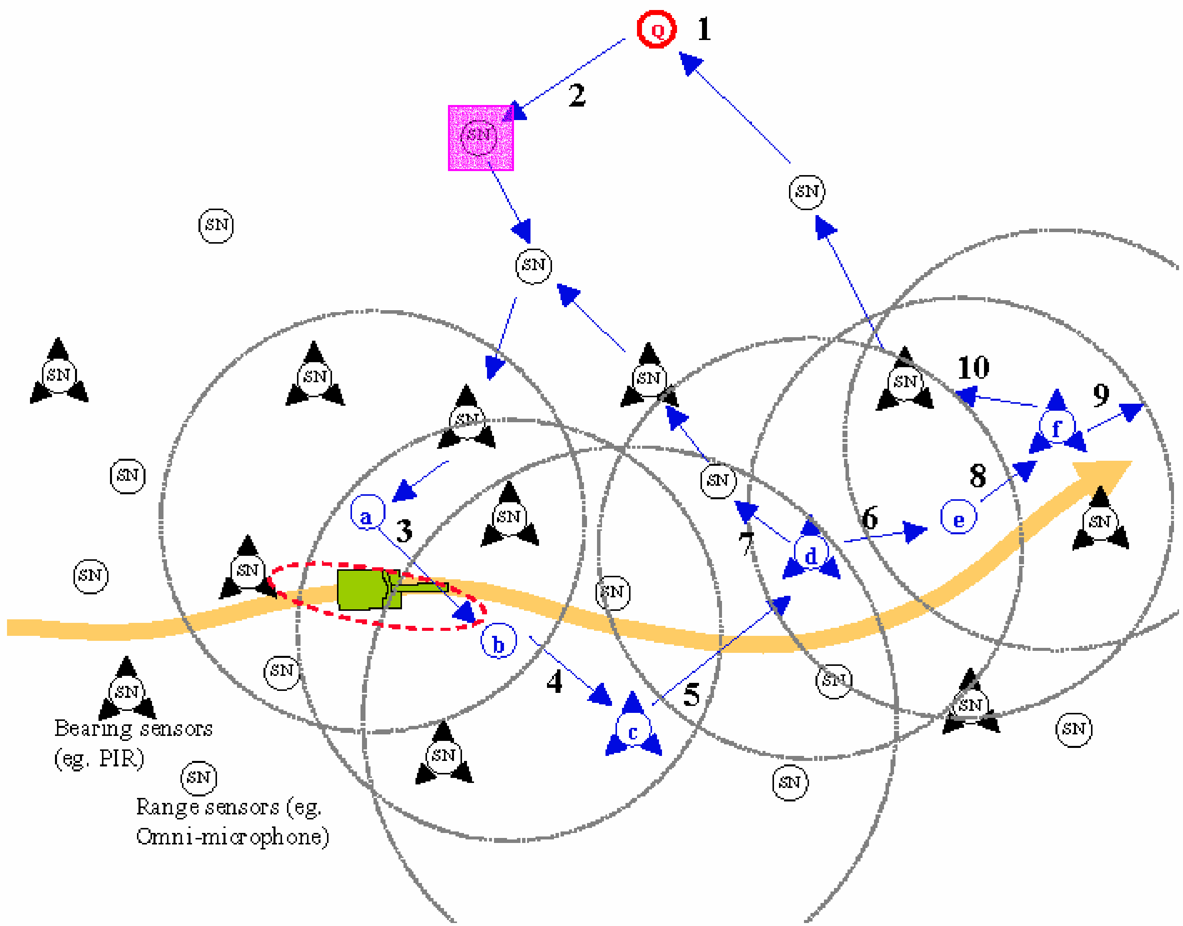
- Motivation
- Byers 2000
  - Cost-Utility Model
  - Algorithm
  - Experimental Results
- Zhao 2002
  - IDSQ Model
  - Algorithm
  - Experimental Results

Tracking Scenario

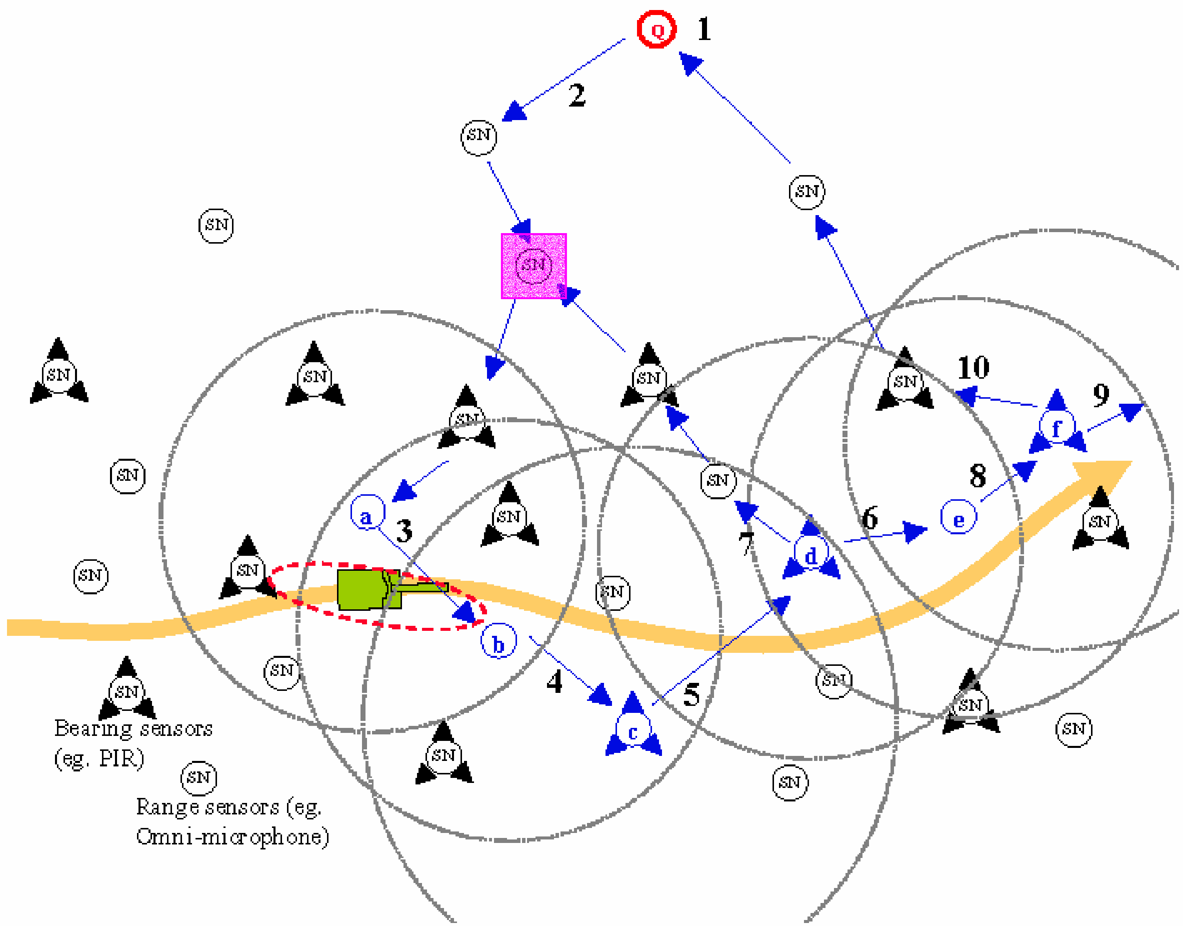


- Query must be routed to the node best able to answer it

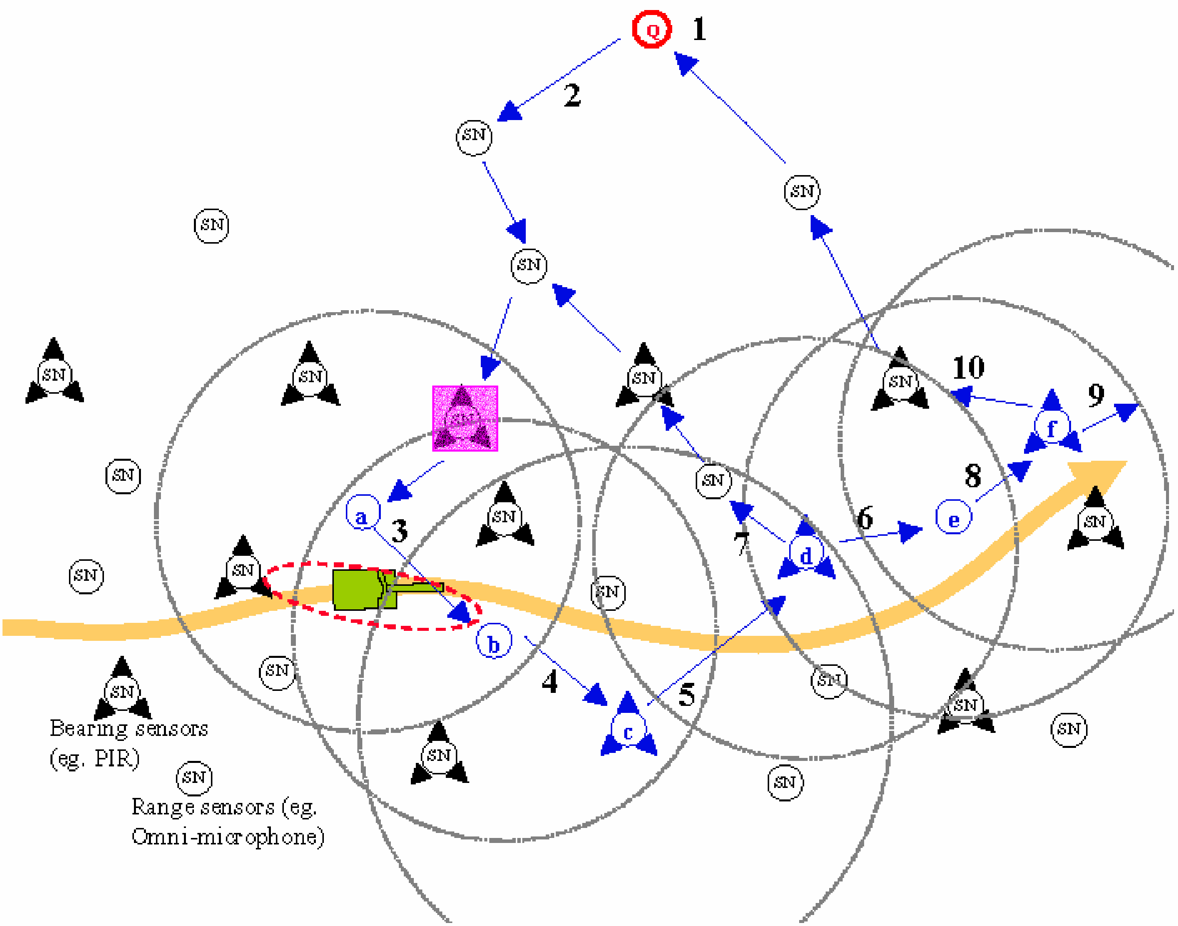
Tracking Scenario



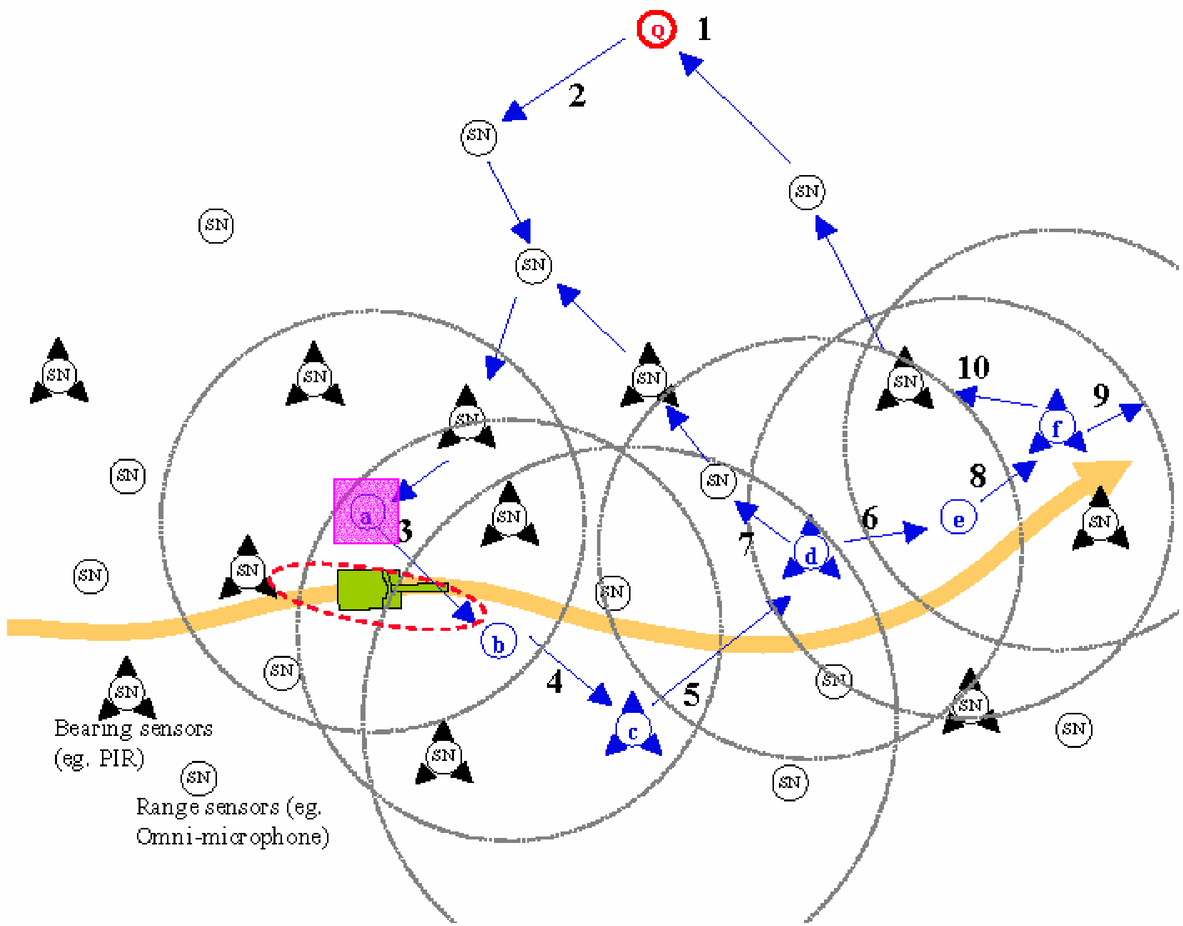
Tracking Scenario



Tracking Scenario

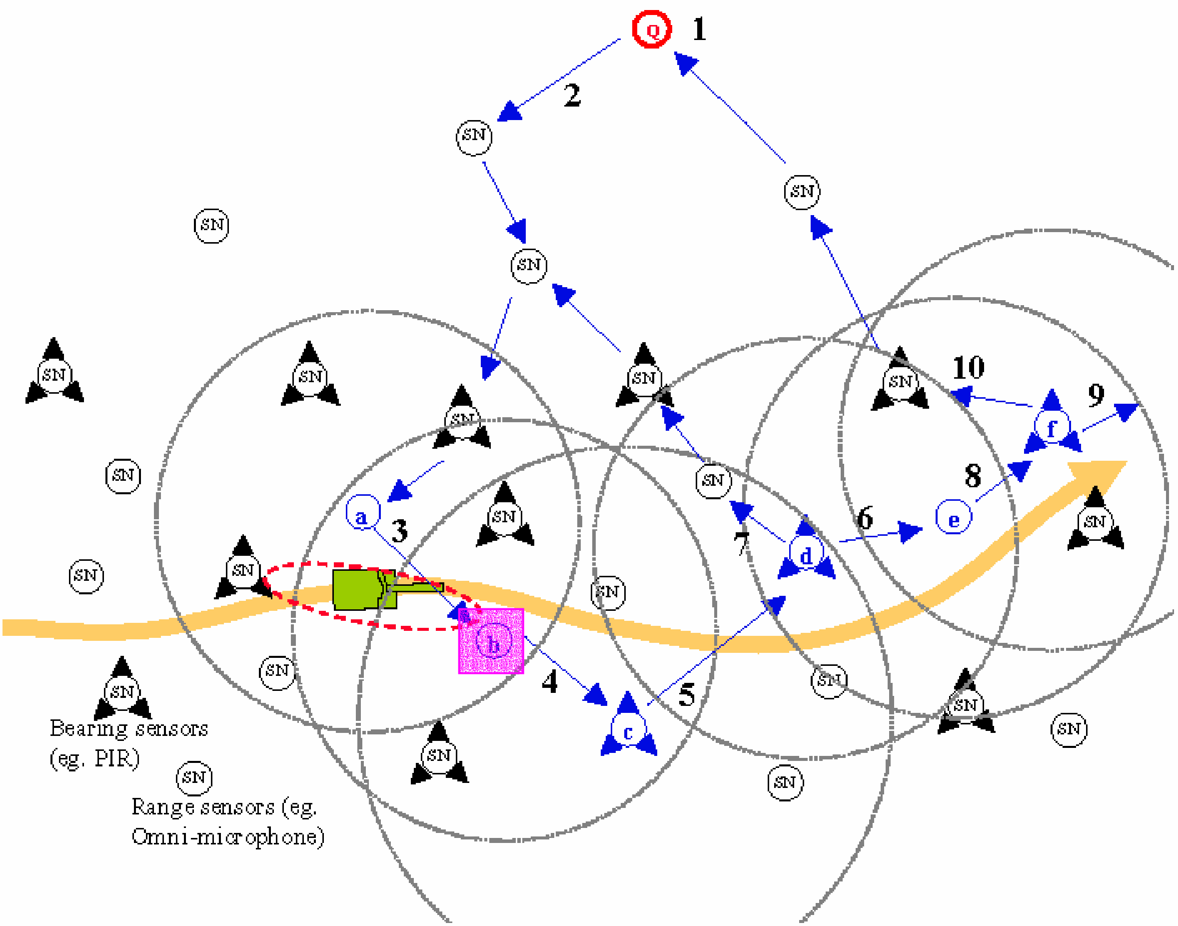


Tracking Scenario



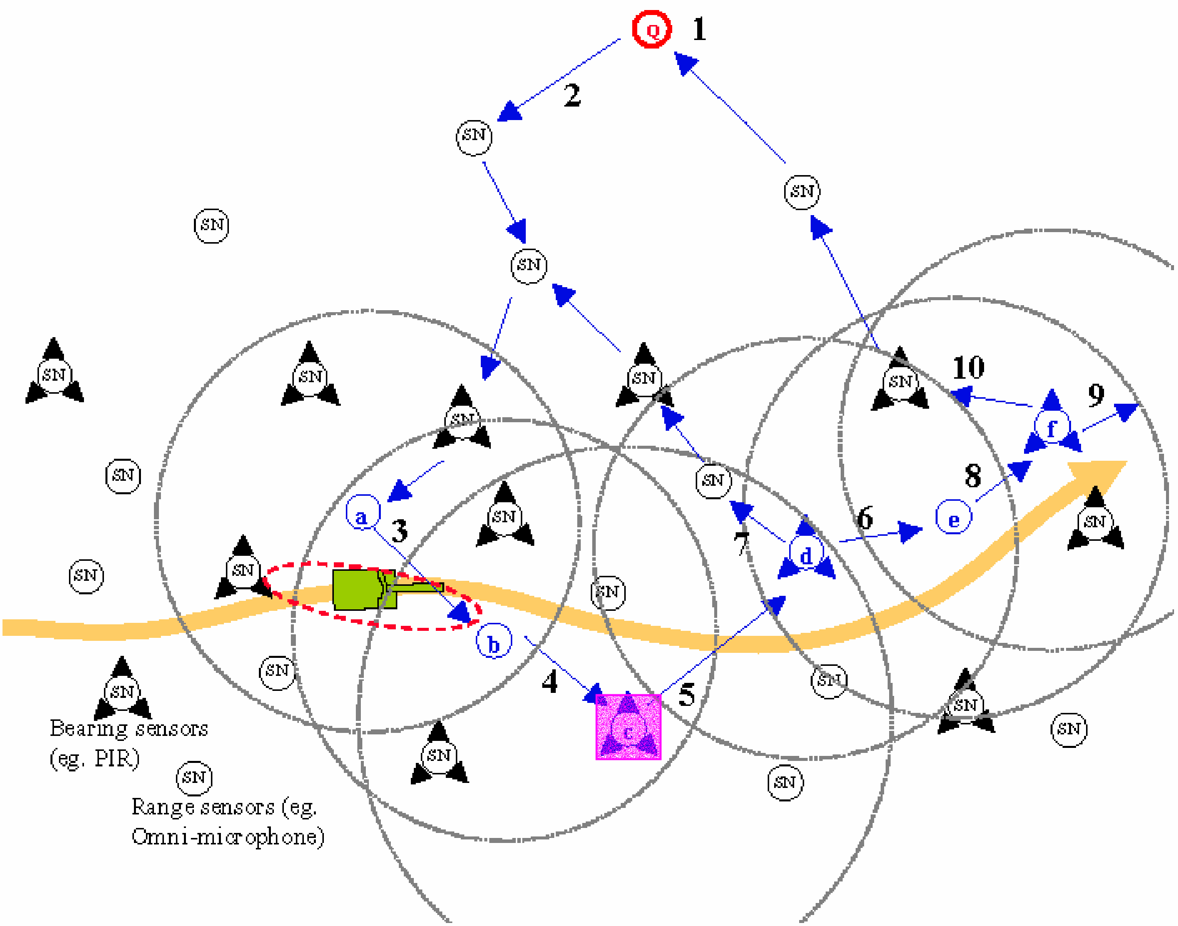
- Sensor a senses the location of the target and chooses the next best sensor

Tracking Scenario

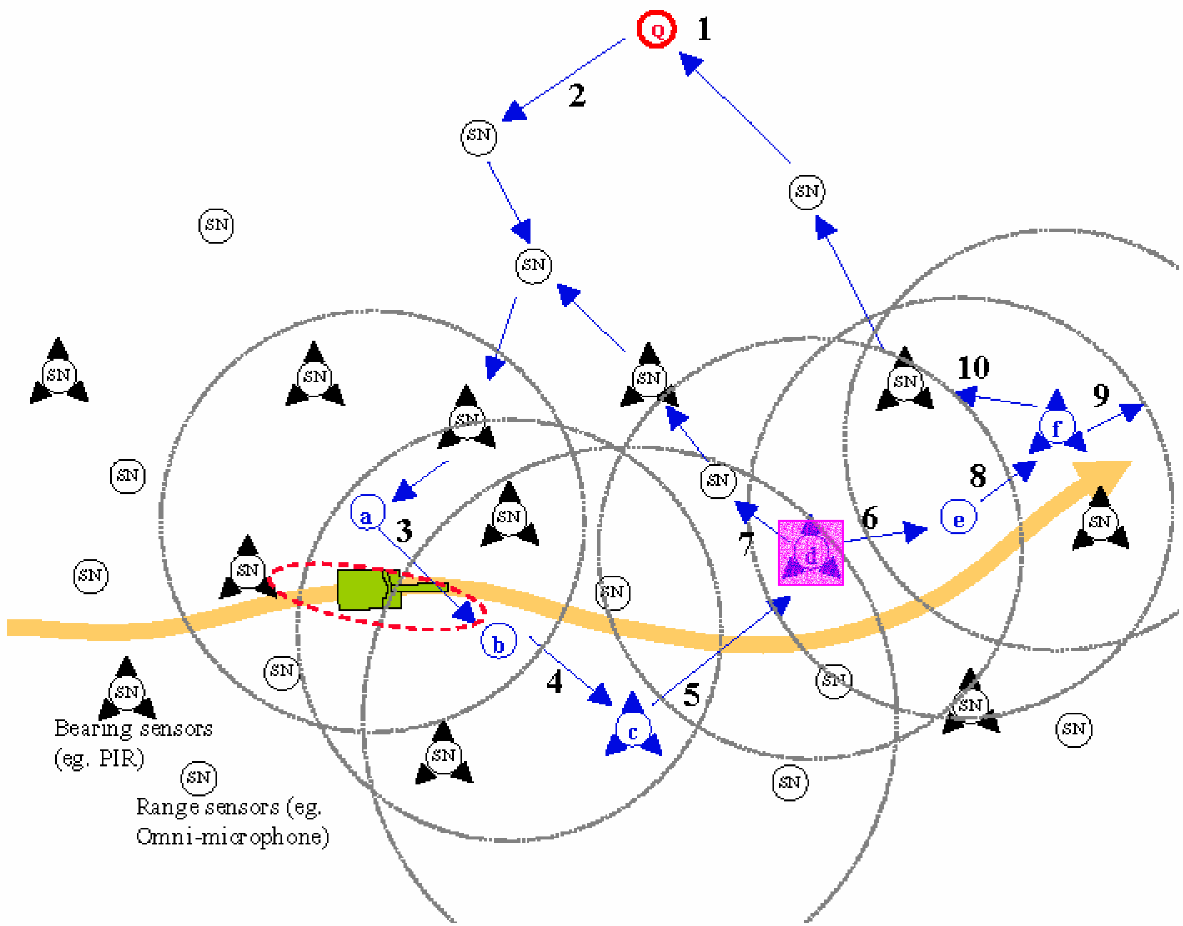


- Sensor b does the same

Tracking Scenario

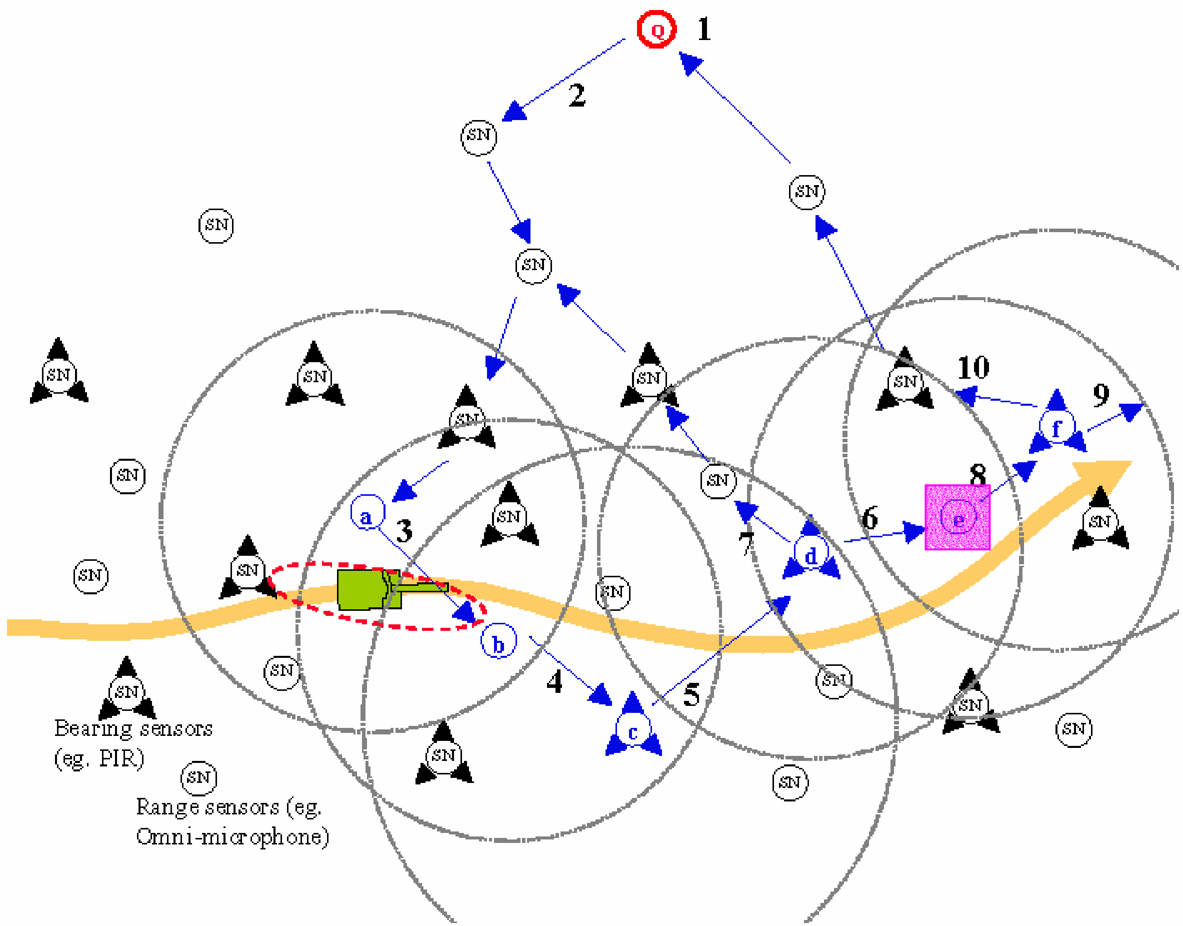


Tracking Scenario

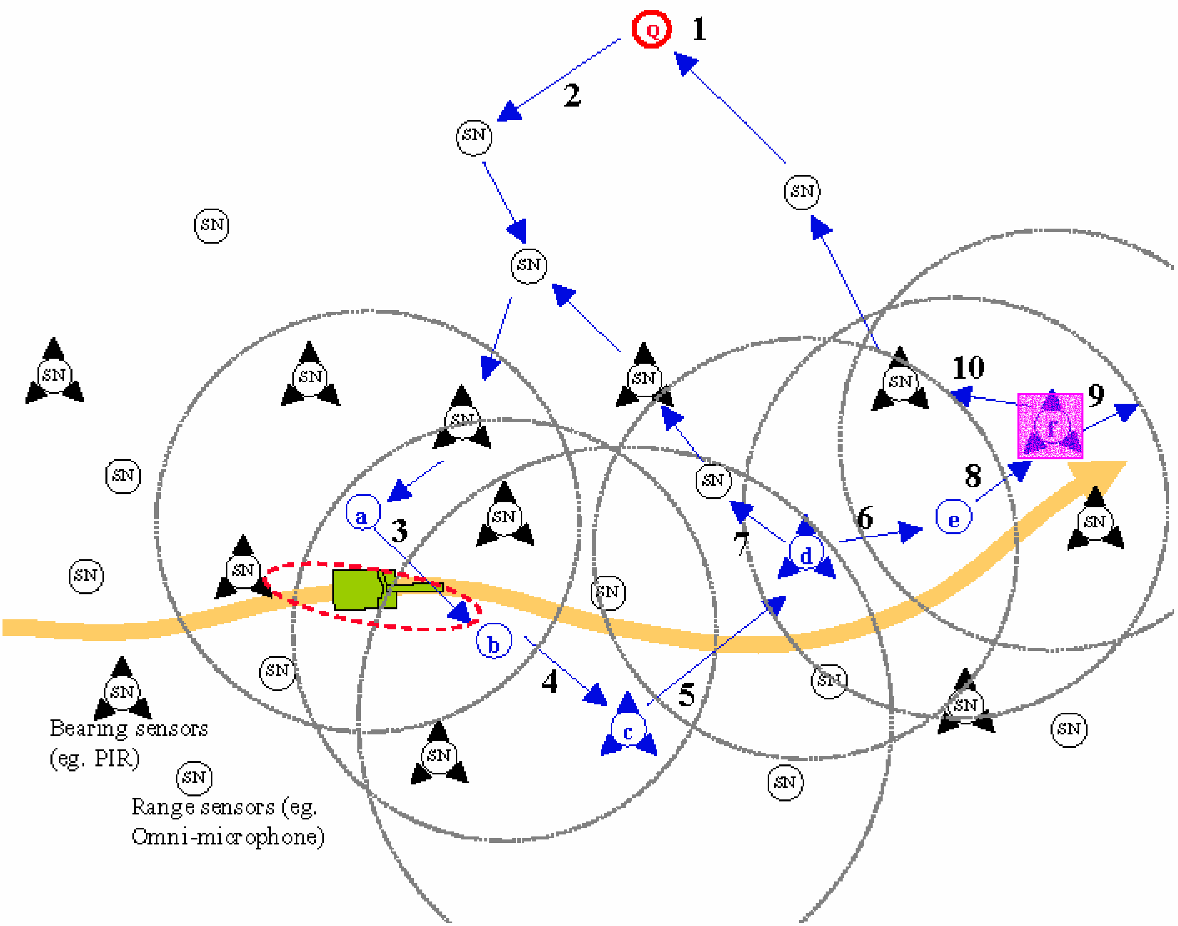


- Sensor d both chooses the next best sensor and also sends a reply to the query node

Tracking Scenario



Tracking Scenario



- Sensor f loses the target and sends the final response back to the query node

IDSQ: Information Driven Sensor Querying and Data Routing

- Algorithm to repeatedly choose next best sensor
- Objective:

Cost of Acquiring z_j

$$M(p(\mathbf{x} \mid \mathbf{z}_1, \cdots, \mathbf{z}_j)) = \alpha \cdot \varphi_{Utility}(p(\mathbf{x} \mid \mathbf{z}_1, \cdots, \mathbf{z}_j)) - (1-\alpha)\,\varphi_{Cost}(\mathbf{z}_j)$$

Net Value of Acquiring z_j

Value of Acquiring z_j

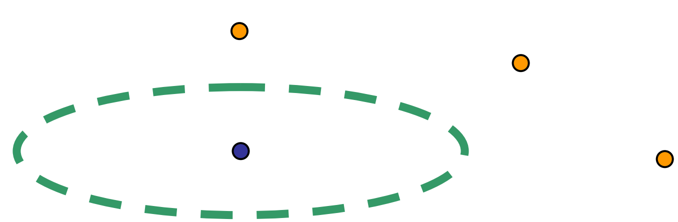
Relative weighting of utility and cost

Aside: Mahalanobis Distance

- The quantity r in

$$r^2 = (x - m_x)'\, C_X^{-1}\, (x - m_x)$$

is called the Mahalanobis distance from the feature vector **x** to the mean vector $\mathbf{m}_x$, where C_x is the covariance matrix for **x**



(all orange points are Mahalanobis equidistant)

Information Utility

- Target location estimated from observations z_1 through z_{j-1}
- Choose next observation providing maximal information utility when incorporated into belief:

$$\hat{j} = \operatorname{argmax}_{j \in V} \varphi_{Utility}\left(p\left(\mathbf{x} \mid \{\mathbf{z}_i\}_{i \in U} \cup \{\mathbf{z}_j\}\right)\right)$$

New Information

Value of Information

Current Information

Candidate Utility 1: Relative Entropy

- Information utility:

$$\varphi(p(\mathbf{x} \mid \{\mathbf{z}_i\}_{i \in U} \cup \{\mathbf{z}_j\})) \triangleq -H_p(x)$$

- Quick Definitions
  - Entropy is uncertainty
  - Relative Entropy is difference in entropy
- Best metric but impossible to implement

Candidate Utility 2: Mahalanobis Distance

- Information utility:

$$\varphi(\mathbf{x}_j, \hat{\mathbf{x}}, \hat{\Sigma}) = -(\mathbf{x}_j - \hat{\mathbf{x}})^T \hat{\Sigma}^{-1} (\mathbf{x}_j - \hat{\mathbf{x}})$$

Sensor Position

Mean, Covariance of Belief

- Works well for Gaussian distributions
- Does not generalize well to heterogeneous sensors

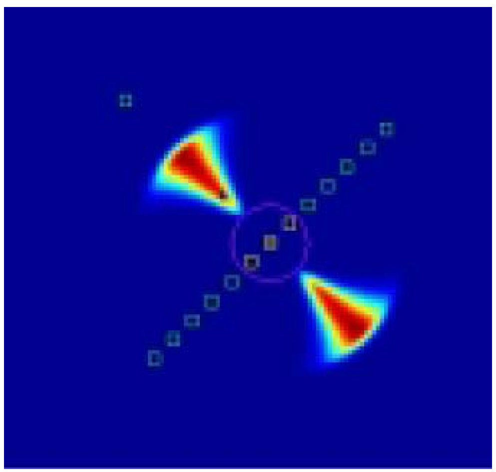
Candidate Utility 3: Expected Relative Entropy

- Observation at sensors unknown
- Need observations for relative entropy
- Posterior gives expected observations
- Useful for finding expected relative entropy

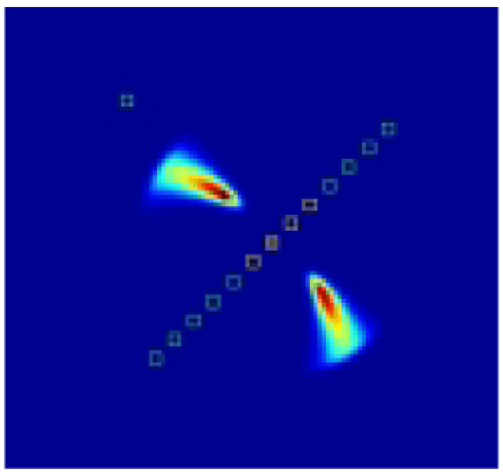
Overview

- Motivation
- Byers 2000
  - Cost-Utility Model
  - Algorithm
  - Experimental Results
- Zhao 2002
  - IDSQ Model
  - Algorithm
  - Experimental Results

Experimental Results: Sensor Choice



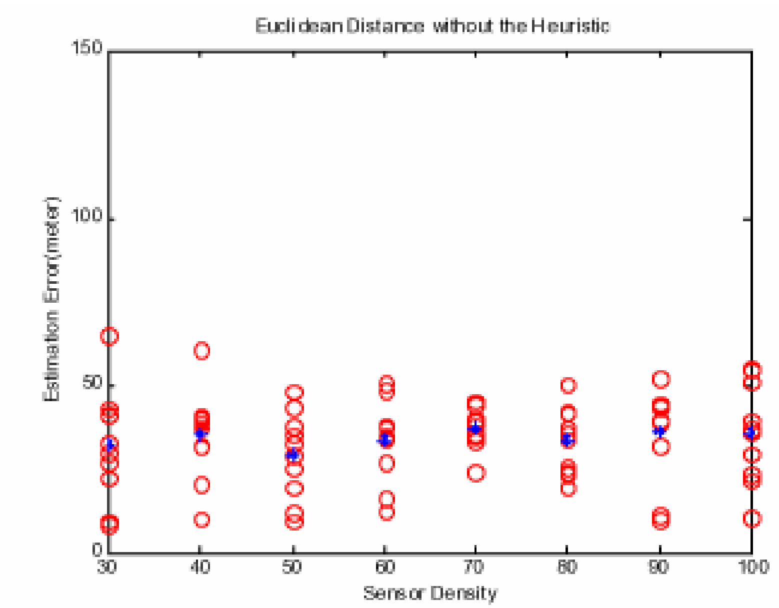
Original Belief

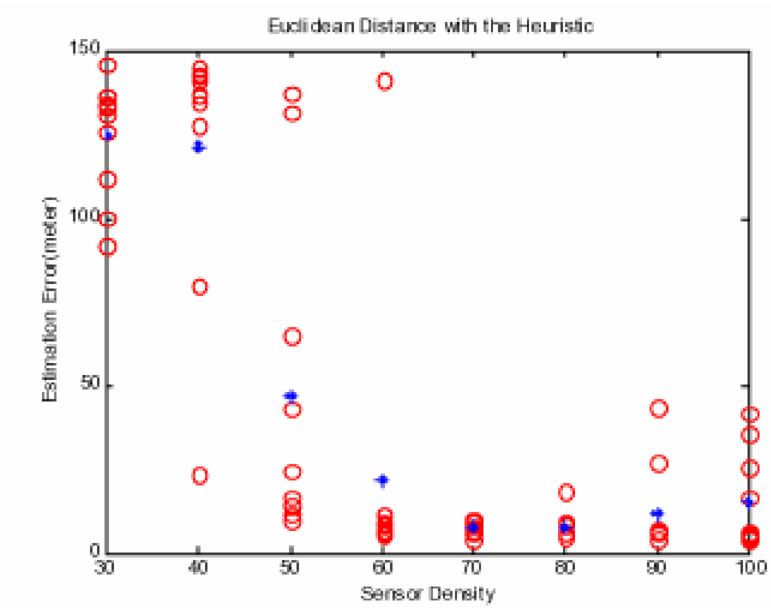


Using Adjacent Sensor

Using Mahalanobis Information Utility

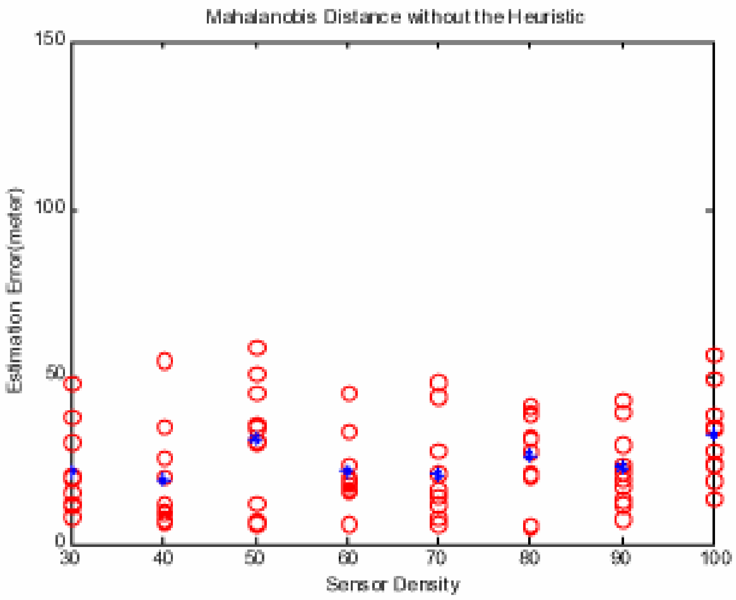
Experimental Results: Euclidean Distance to Mean

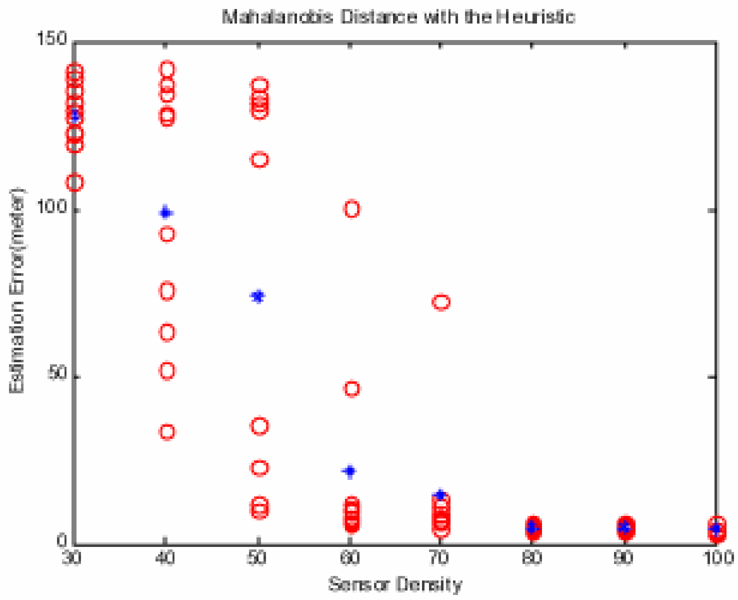




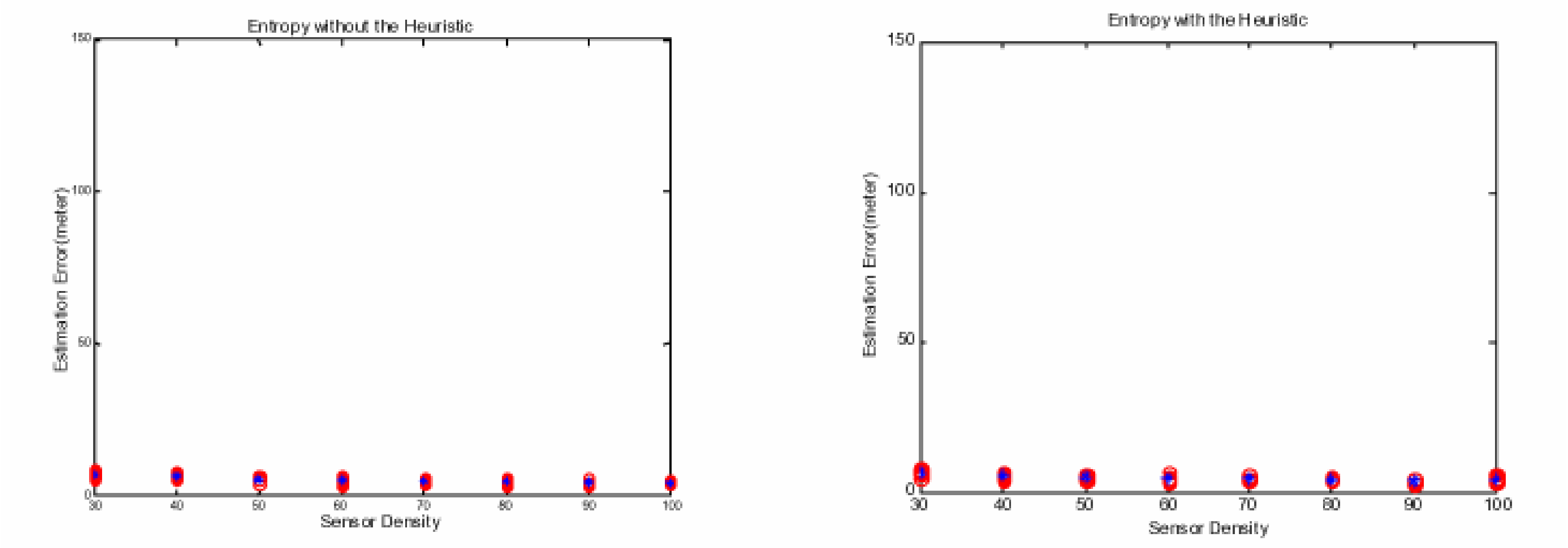
(heuristic prevents node from being selected N times)

Experimental Results: Mahalanobis Distance to Mean

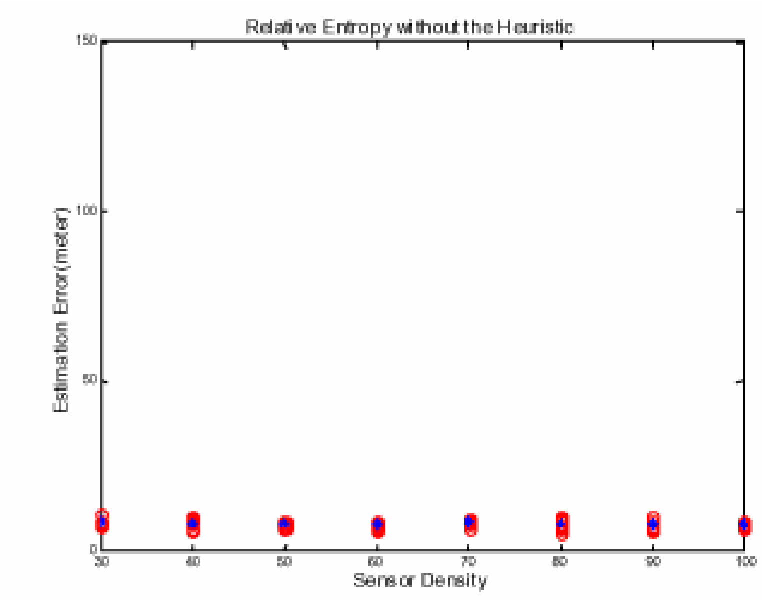


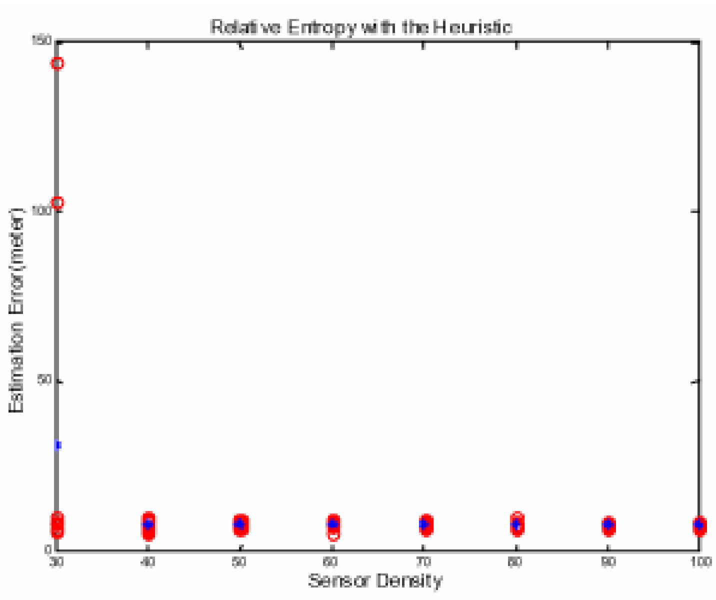


Experimental Results: Entropy (unattainable)



Experimental Results: Relative Entropy





Experimental Results: Tracker Performance

	# Lost tracks/ total runs	Mean error
(a) Nearest neighbor	75/80	34.39
(b) Nearest neighbor with heuristic	37/80	44.79
(c) Mahalanobis distance	70/80	24.86
(d) Mahalanobis distance with heuristic	32/80	44.20
(e) Entropy	0/80	5.13
(f) Entropy with heuristic	0/80	5.05
(g) Relative entropy	0/80	8.09
(h) Relative entropy with heuristic	2/80	10.79

Conclusion

- Power-aware sensor management:
  - Increases sensor lifespan
  - Decreases number of sensors needed
- Information utility:
  - Directs sensing to find more valuable information
  - Balances power consumption and information acquisition