Wireless Sensor Networks: New Capabilities and Potentials

Aman Kansal

Networked Embedded Computing

Microsoft Research

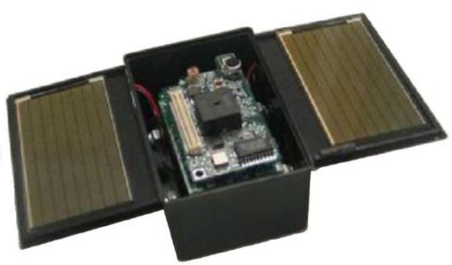
Based on research carried out at: UCLA Microsoft Research

Three New Capabilities

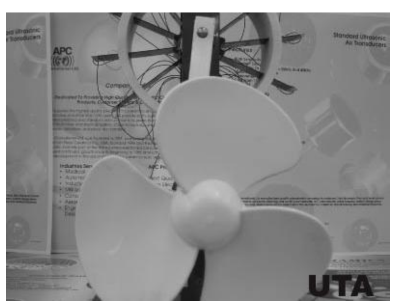
Performance Issue	New Capability
ENERGY SCARCITY	Harvest from Environment
SENSING RESOLUTION	Low Complexity Controlled Motion
DEPLOYMENT	Peer Production

Energy Harvesting

- Energy aware algorithms extend battery life: but battery size is **finite**
- Harvesting from environment: **infinite** energy
  - Power is finite

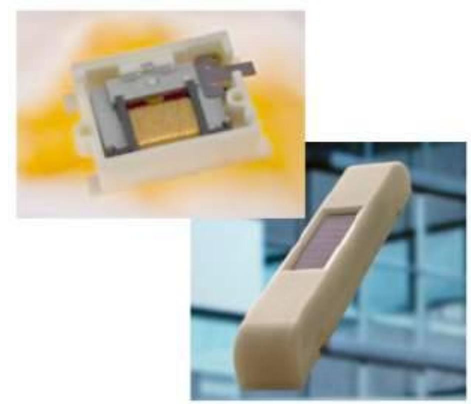


Heliomote (Solar, UCLA) 2003



Piezoelectric Windmill (Wind, UT Arlington) 2004

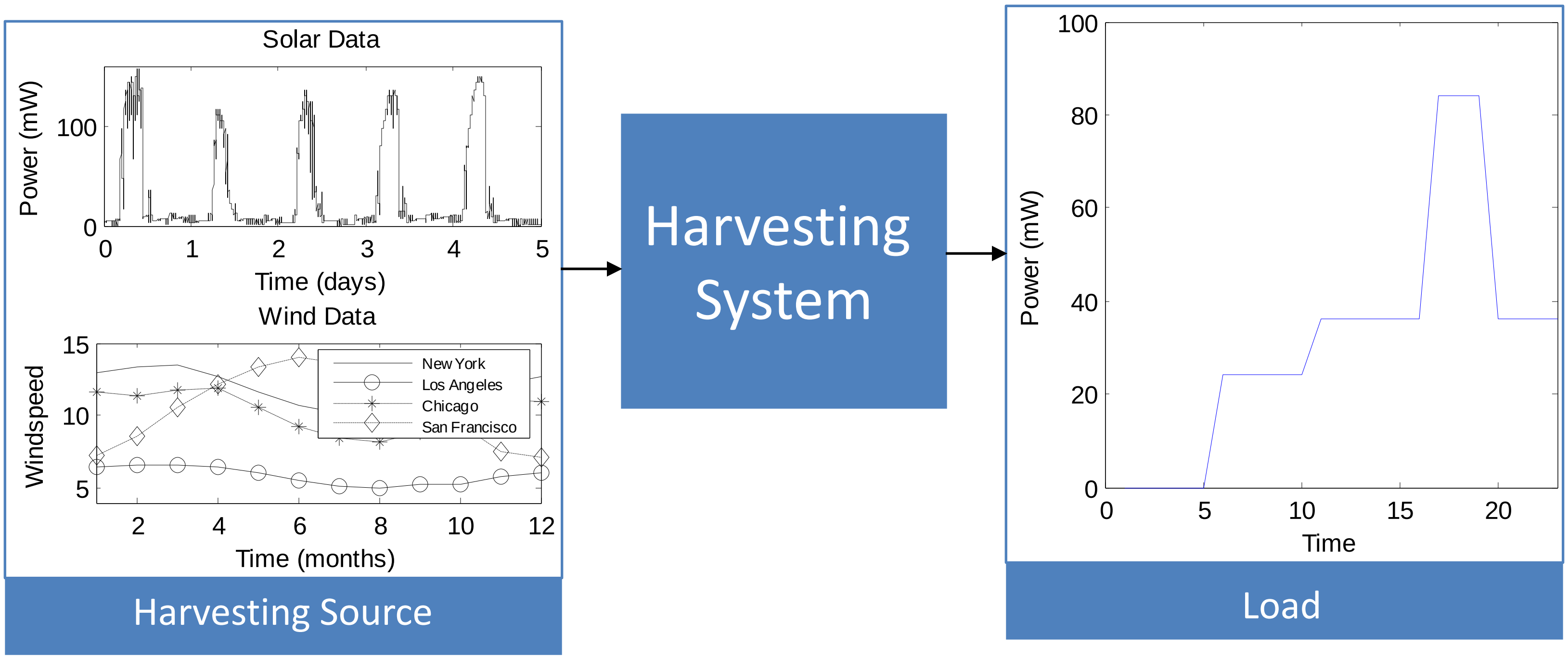
Trio/Prometheus (Solar, UC Berkeley) 2005



EnOcean (Solar/Mechanical) Commercial

Harvesting Issues

- Harvested power level varies in time
  - independent of energy usage profile
- Solution: match consumption to production



Harvesting Theory

- Energy buffer: size B, efficiency η, leakage P_{leak}
- Harvested source power $P_s(t)$, Load power $P_c(t)$

Energy conservation

$$B_0 + \eta \int_0^T [P_s(t) - P_c(t)]^+ dt - \int_0^T [P_c(t) - P_s(t)]^+ dt - \int_0^T P_{leak}(t)dt \geq 0 \quad \forall T \in [0, \infty)$$

Battery size limitation

$$B_0 + \eta \int_0^T [P_s(t) - P_c(t)]^+ dt - \int_0^T [P_c(t) - P_s(t)]^+ dt - \int_0^T P_{leak}(t)dt \leq B \quad \forall T \in [0, \infty)$$

where: $[x]^+ = x$ if $x>0$, 0 otherwise

Performance Guarantee

- Definition:

$P(t)$ is called (ρ,σ_1,σ_2) if

$$\int_{\tau}^{\tau+T} P(t)dt \leq \rho T + \sigma_1$$
$$\int_{\tau}^{\tau+T} P(t)dt \geq \rho T - \sigma_2$$

For all positive T and τ

(Modeling "burstiness")

- **Theorem:** If $P_s(t)$ is $(\rho_1,\sigma_1,\sigma_2)$ and $P_c(t)$ is (ρ_2,σ_3), then energy neutrality can be achieved for

$$\rho_2 \leq \eta\rho_1 - \rho_{leak}$$
$$B_0 \geq \eta\sigma_2 + \sigma_3$$
$$B \geq B_0$$

Proof: "Power Management in Energy Harvesting Sensor Nodes", *ACM Trans. Embedded Computing Systems.*

Designing Practical Systems

What battery size to use?

- Directly calculate from theorem

Adapt performance at each node

- maximize performance at energy neutrality

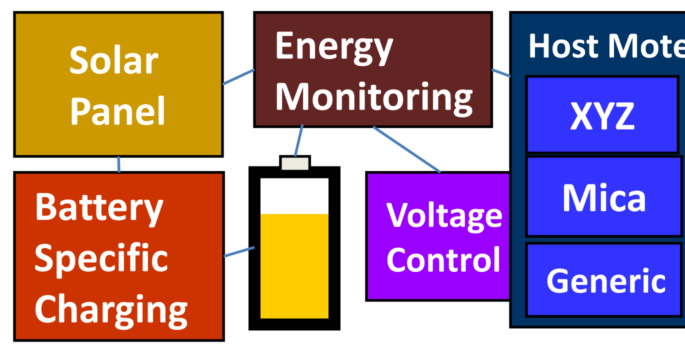
Adaptation across network

- Adapt workload allocation to spatio-temporal harvesting profile

Experimental Harvesting Node

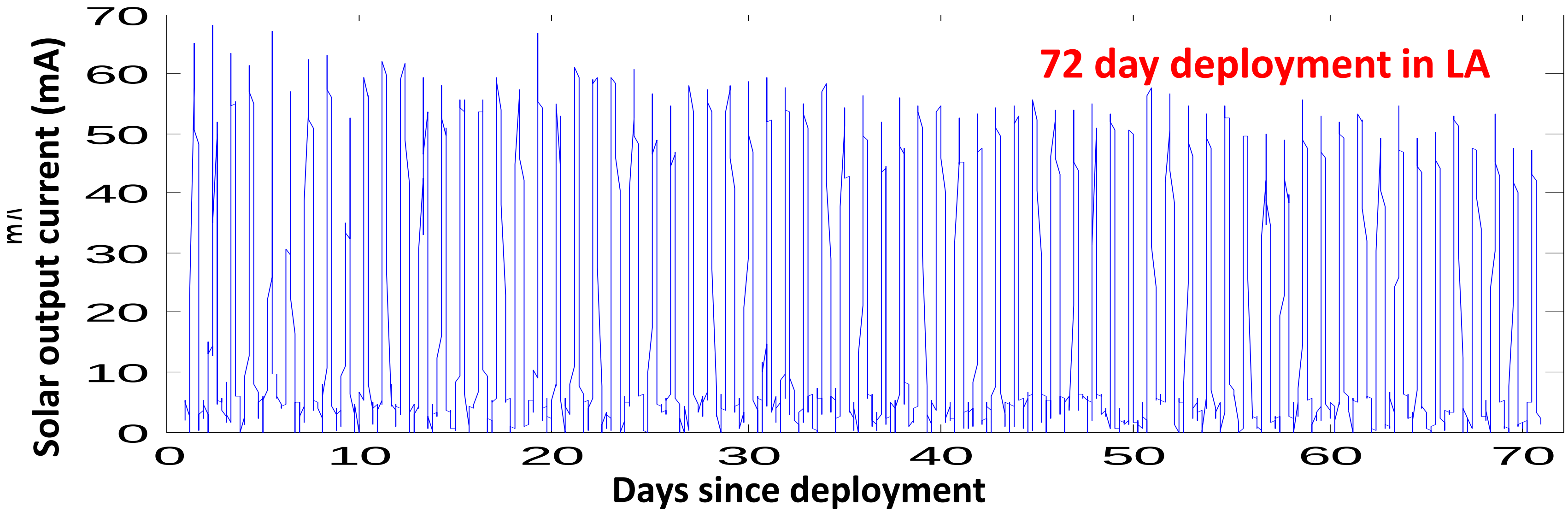
- Hardware Design
  - Maximal power point matching for solar cell
  - Storage management
    - Technology: NiMH vs Li+ vs. ultracapacitor
    - Switching between battery and solar cell
  - Energy **measurement interface** for adaptive algorithms
- Heliomote v2
  - Fully autonomous (no dependence on host mote), >80% efficiency
  - Deployed by many research groups
  - Commercially available [ATLA Labs]

Heliomote v2. UCLA 2005

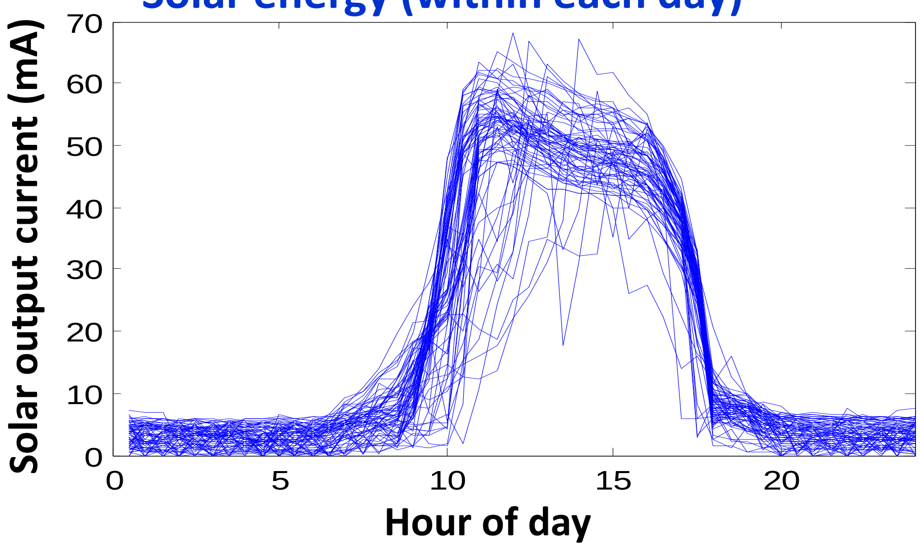


Details: "Design considerations for energy harvesting wireless embedded systems", ACM/IEEE IPSN, 2005.

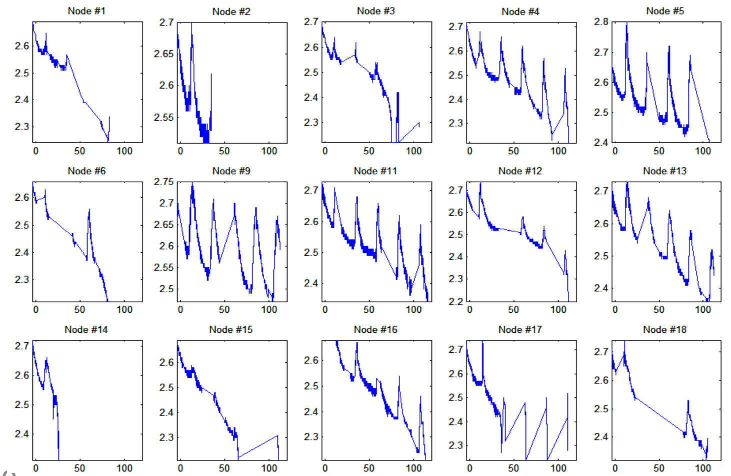
Observed Harvesting Opportunity



Solar energy (within each day)



Multiple nodes, 1 week



Real Time Adaptation

- Change duty cycle, D, to control consumption
  - Higher duty cycle has higher utility

Maximize utility over time window N_W

$$\max \sum_{i=1}^{N_w} D(i)$$

Subject to

(1) Energy conservation

$$B(i)-B(i+1)=\Delta T D(i)\left[P_c-P(i)\right]^{+}-\eta\Delta T P(i)\left\{1-D(i)\right\}-\eta T D(i)\left[P(i)-P_c\right]^{+}$$

(2) Energy neutrality over N_w

$$B(1)=B_0$$

$$B(N_w+1)=B_0$$

B(i) > 0

Embedded Algorithm

Predict availability over N_w

$E^k(i+1)=a*E^k(i)+(1-a)*E^k_{av}$

↓

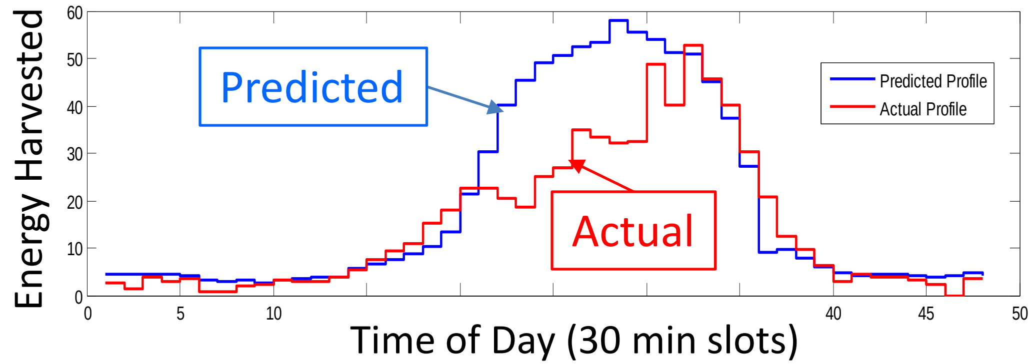
Allocate duty cycles over N_w

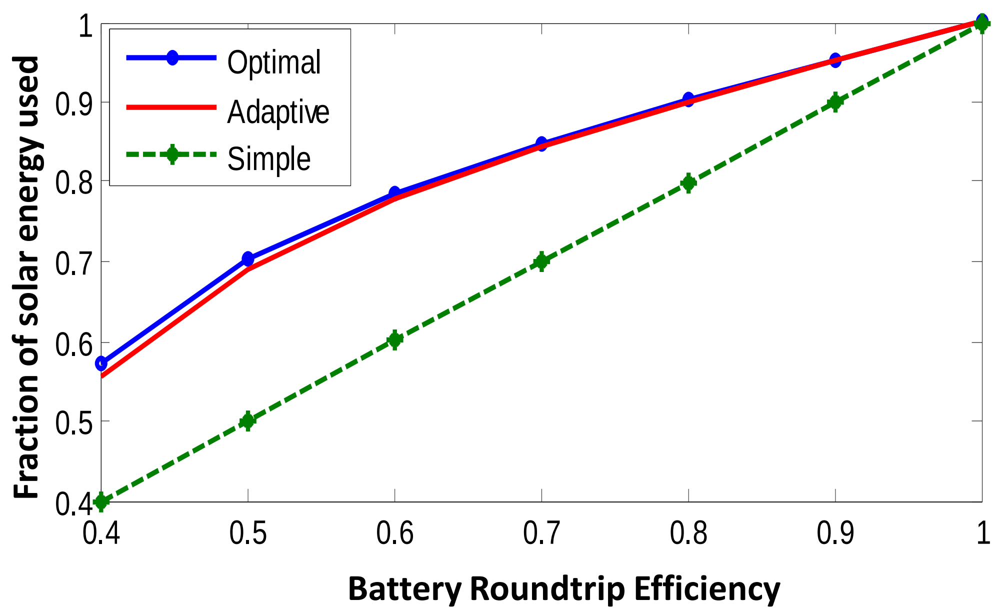
Exploit structure to solve LP

↓

Adapt to energy harvested

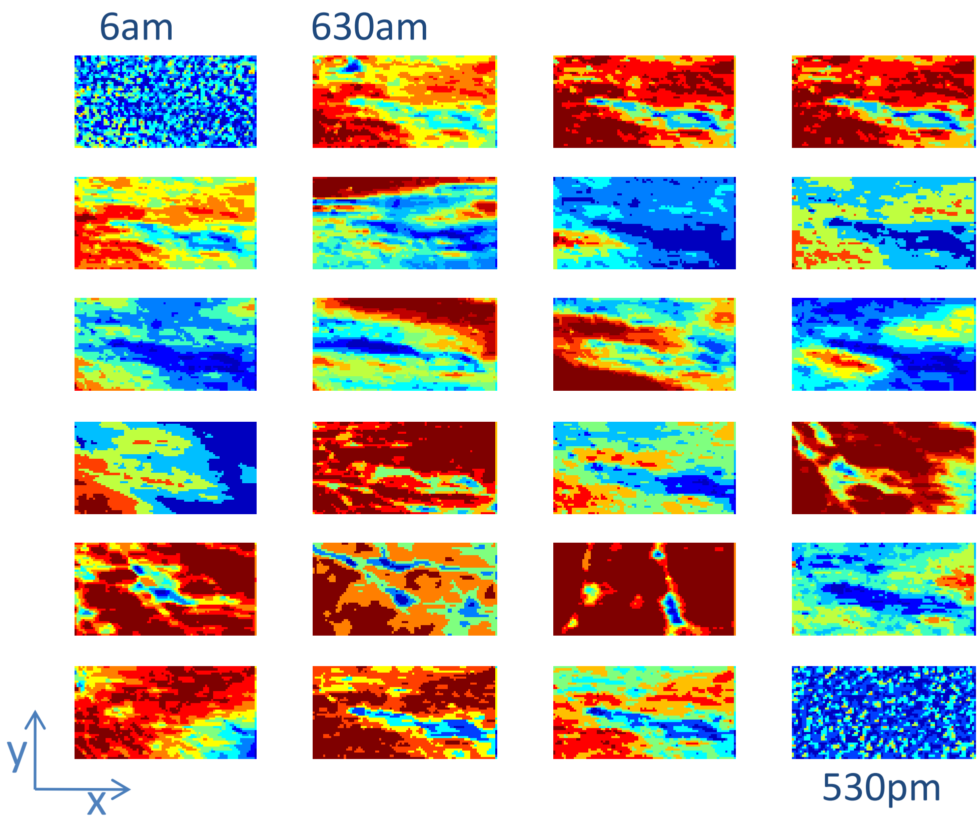
Actual varies from predicted





Algorithm: "Adaptive Duty Cycling for Energy Harvesting Systems", ISLPED 2006.

Harvesting in a Network

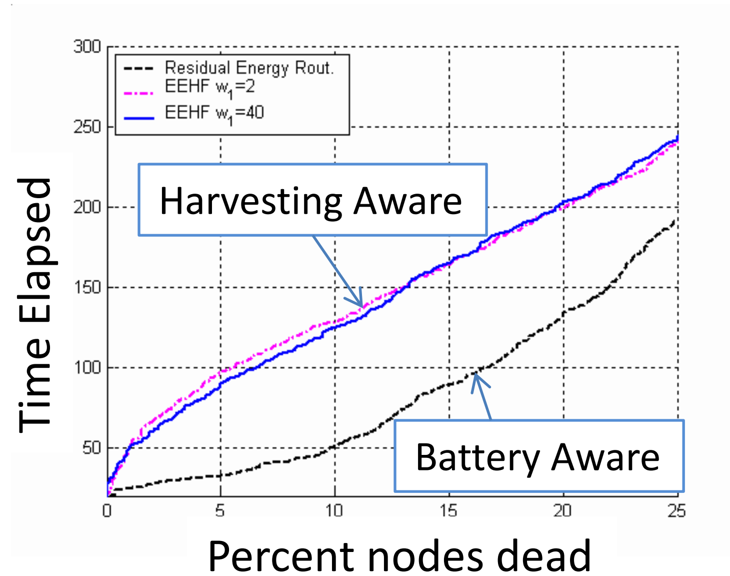


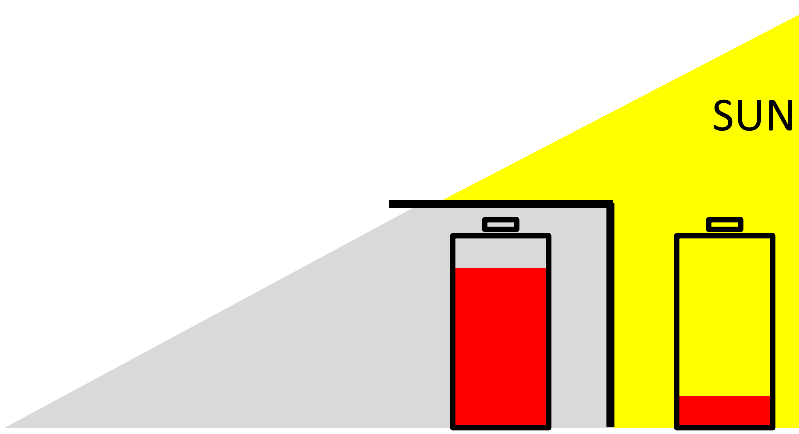
Spatial harvesting opportunity,
at different times of day
(a region in James Reserve, CA)

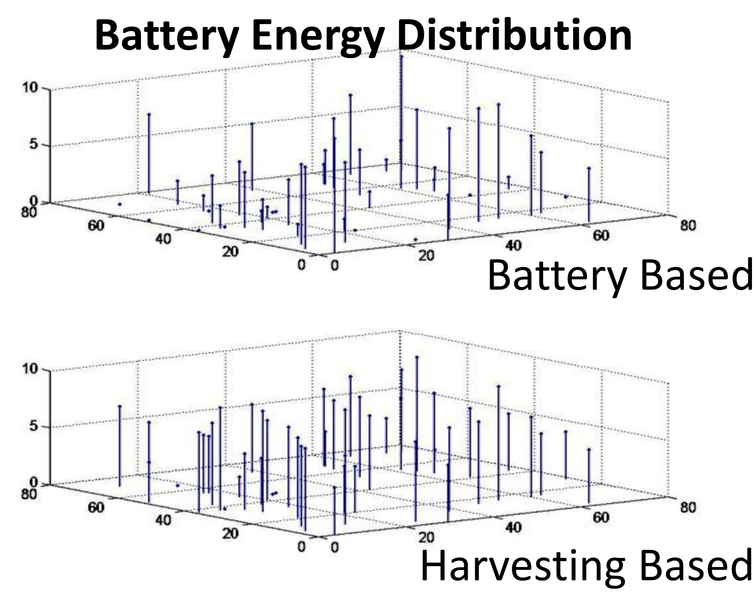
- Example: **Routing**
  1. Field monitoring
     - Data transmission load dominates
  2. Event detection
     - Listen mode energy dominates

Field Monitoring

- Routing based on harvesting potential
  - Include current battery level to avoid short term failure



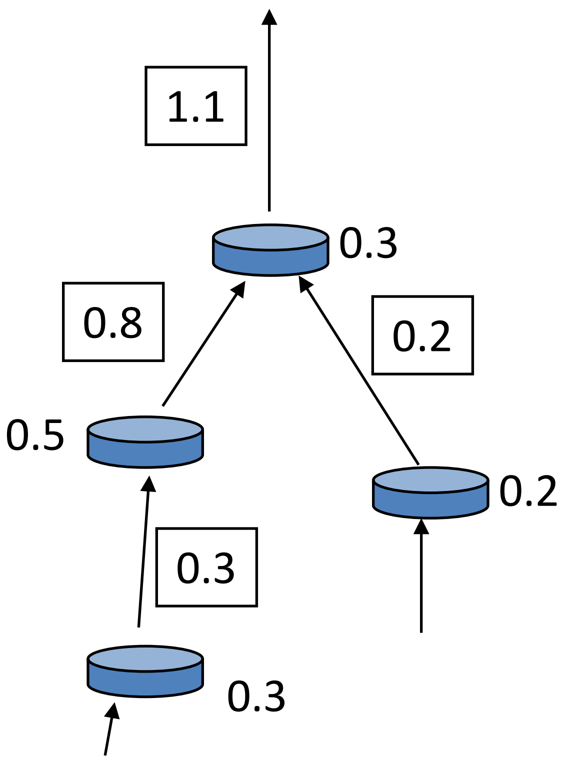


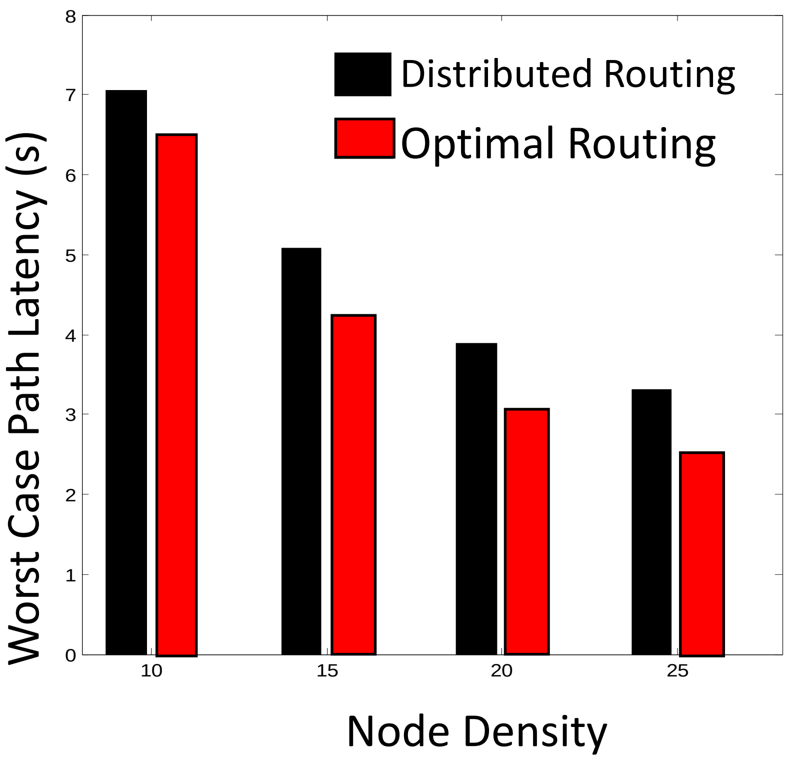


Details: "An environmental energy harvesting framework for sensor networks", ACM/IEEE ISLPED, 2003.

Event Detecting Network

– Energy neutral duty cycle: latency sensitive routing
  - Distributed algorithm to discover best route to base
    – Without exchanging all link costs





Details: "Performance Aware Tasking for Environmentally Powered Sensor Networks", ACM SIGMETRICS, 2004.

Three New Capabilities

Performance Issue	New Capability
ENERGY SCARCITY	Harvest from Environment
SENSING RESOLUTION	Low Complexity Controlled Motion
DEPLOYMENT	Peer Production

Coverage Resolution

Coverage: Total extent of sensing (area, volume)

Resolution: sensor capability (pixels)

15 pixels/m

1MP aggregate sensing resource

200 pixels/m

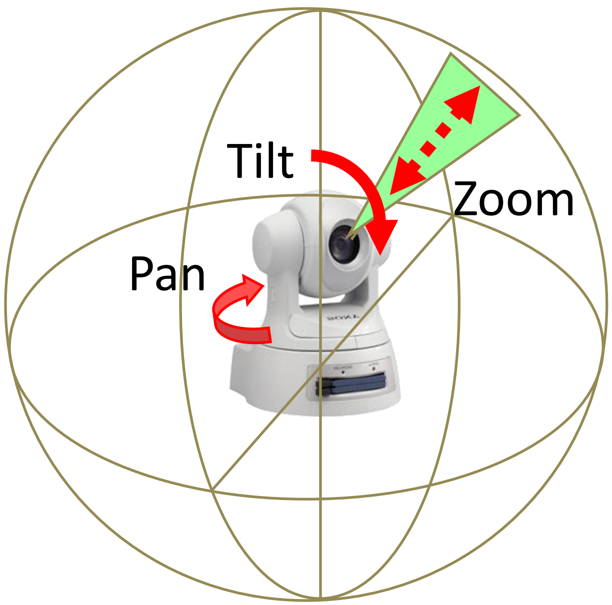
784MP aggregate sensing resource

Mobility

- Mobility useful when static network cannot achieve required coverage resolution
  - Eg. If the space, bandwidth, management resource, or cost for 784MP is not justified for given application
- Mobility helps by:
  - Adaptively allocating sensors where needed
  - Minimizing overlap and occlusions
- Key Trade-off: Motion delay
  - Depends on number of sensors

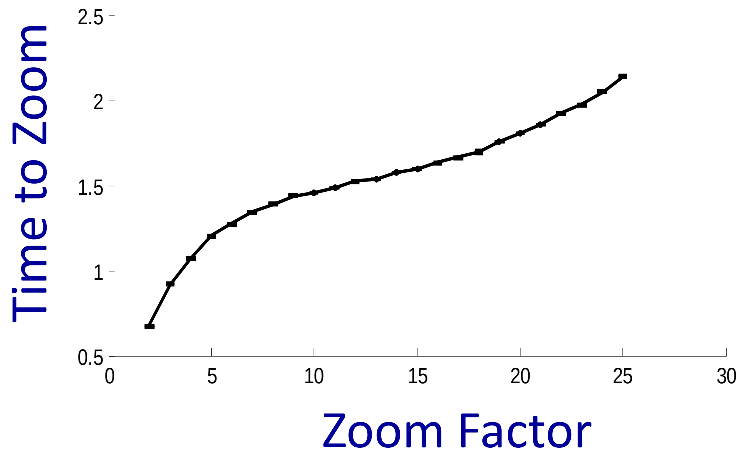
Low Complexity Motion: *Motility*

- Navigational ease
- Low energy
- Feasible in tethered nodes
- Low delay

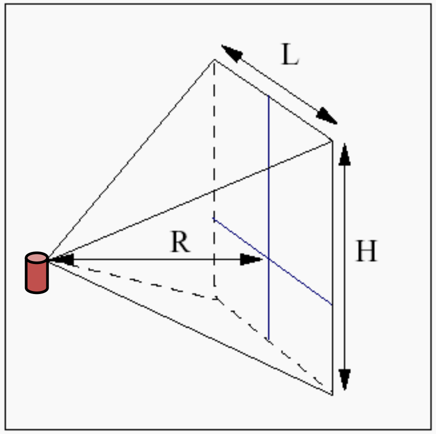


Motion delay for Sony SNC RZ 30N

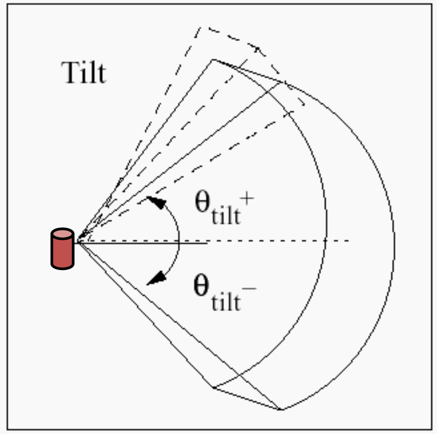
Actuation	Range	Speed
Pan	340º	170º/s
Tilt	115º	77º/s
Zoom	25X	See plot



Motility Advantage

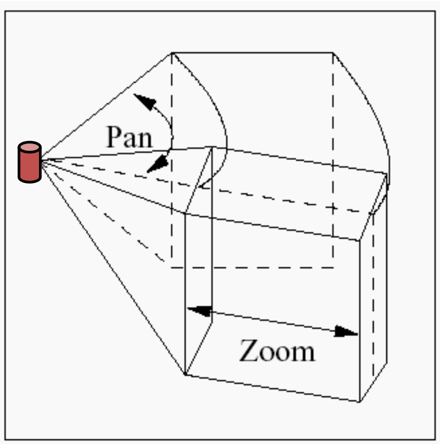


1. Coverage Model

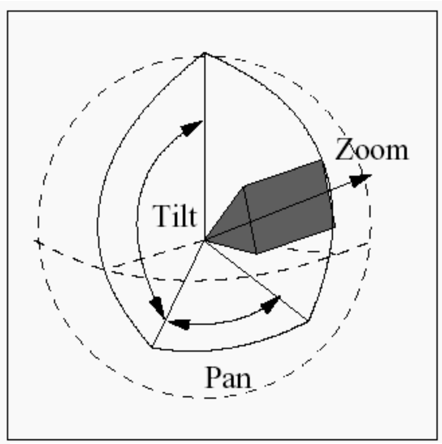


2. With Tilt

$$g_{pzt} = (3z-2) + \frac{\theta_{tilt}R}{2Hl}(3z^2+1) + \frac{\theta_{pan} z^3 R^2 (1-\cos\theta_{tilt})}{LHl^2} + \left[\frac{cos\theta^-_{tilt} + cos\theta^+_{tilt}}{2L}\right]\frac{\theta_{pan}R}{2l}(3z^2+1)$$



3. Pan and Zoom

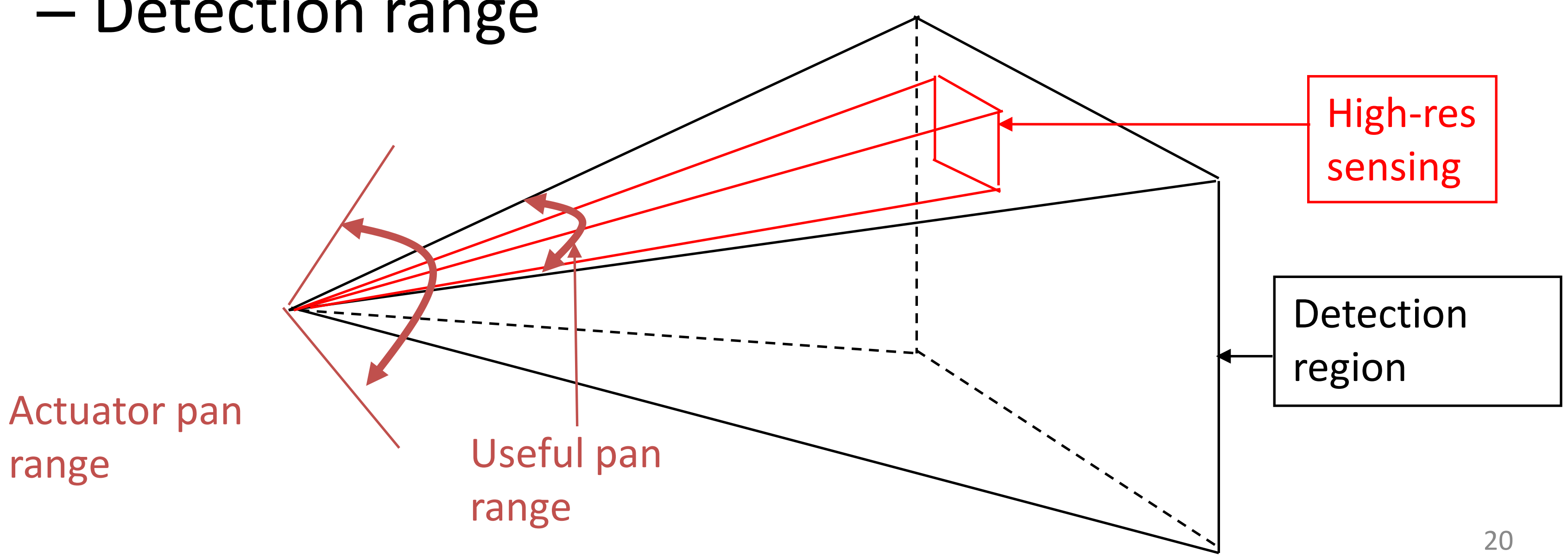


4. Pan, Tilt, Zoom

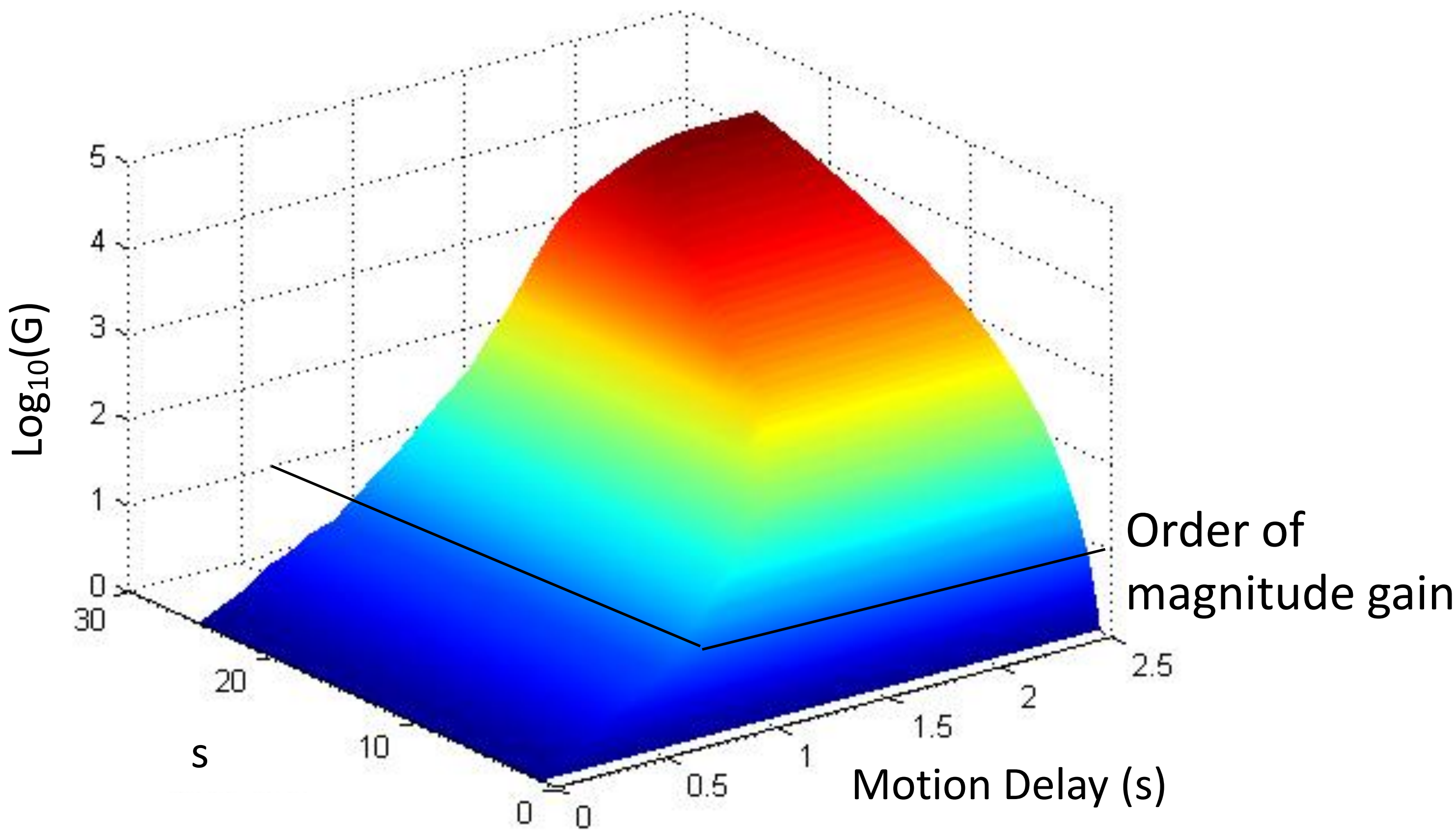
Pan	7.74
Tilt	4.04
Zoom	73
Pan and Zoom	6361
Pan, Tilt and Zoom	226940

Practically Achievable Advantage

- Detect at low res., then sense at high res.
  - Ratio of coverage resolutions: s
- Range of allowed motion limited by
  - Tolerable motion delay
  - Detection range



Practically Achievable Advantage



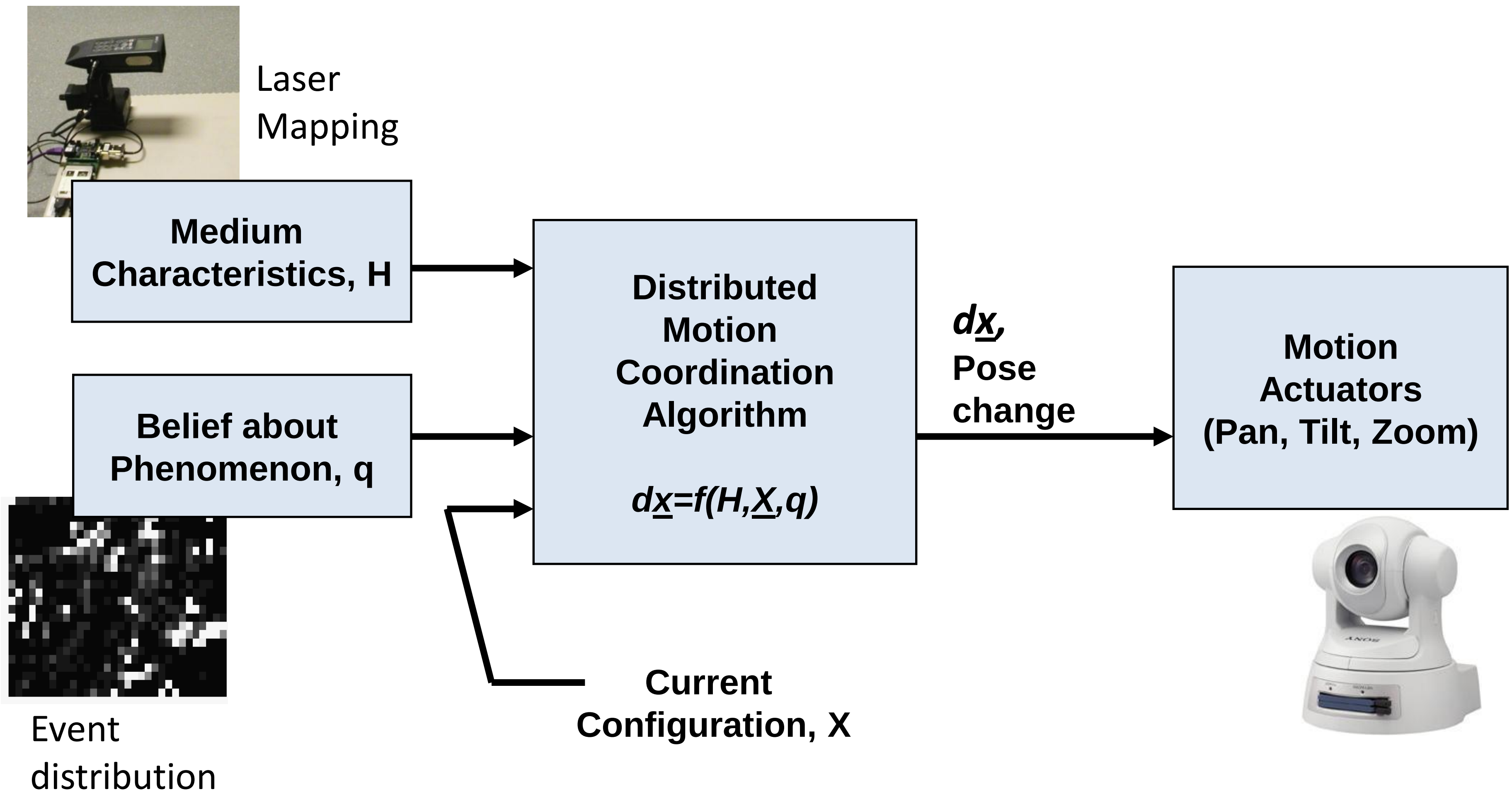
Advantage in a network: "Reconfiguration Methods for Mobile Sensor Networks," *ACM Transactions on Sensor Networks*

Coordination Algorithm

1. **Proactive Phase:** Minimize delay to create a semblance of hi-density deployment
   - Place network in a pose from where responding to sensing demands is quick

2. **Reactive Phase:** Control motility based on real time dynamics
   - Event based
     - Eg: Surveillance - high resolution coverage on intruders
   - Query based
     - Eg: Live view of planet - metro regions may be more queried

Proactive Phase Coordination



Proactive Phase Performance Metric

1. Motion delay
   - Let $\tau(p)$ be the delay to actuate (pan,tilt,zoom) in to point p in covered region:

$$\tau(p) = max\{\delta\theta_p * \omega_p, \delta\theta_t * \omega_t, t_p(\delta_z)\}$$

2. Detection quality, $c(p)$
   - Obstacles (available line of sight), resolution

3. For each point in covered region
   - Linear combination, weight depends on application

$$f(p) = w * c(p) + (1 - w)\frac{1}{\tau(p)}$$

Multiple Sensors, Multiple Points

- For each point consider best camera covering it
  - One that yields highest *f(p)* at *p*
- Weighting by event probability, ***q(p)***

$$u(p) = f(p) * q(p)$$

- Sensing utility of a given network configuration
  - Given area A and pose of each node $\boldsymbol{S}_i = \{pan_i, tilt_i, zoom_i\}$

Optimization Objective

$$U(\mathbf{S}) = \sum_{p \in \mathcal{A}} u(p)$$

Optimization search space:

$$\mathcal{S} = \mathcal{S}_1 \times \mathcal{S}_2 \times \ldots \times \mathcal{S}_N$$

Distributed Algorithm

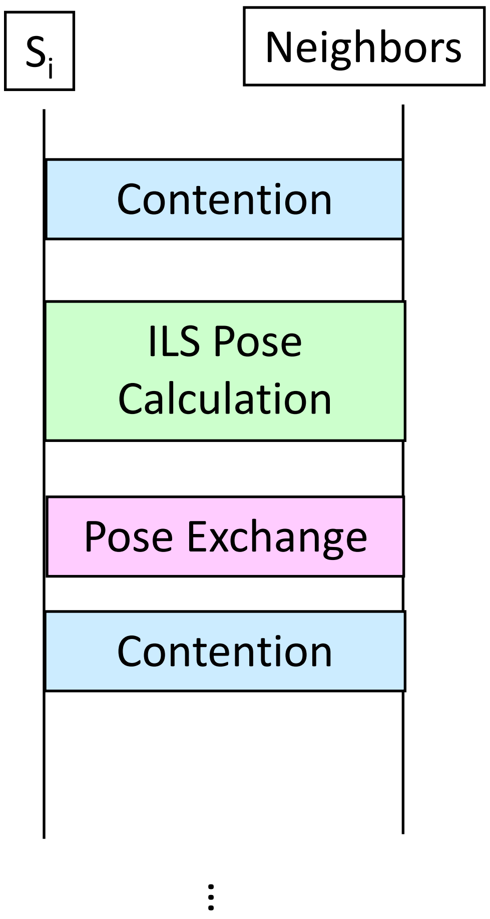
- $U(\mathbf{S})$ can be partitioned such that each sensor i can evaluate the component of utility metric, U_i, affected by sensor i
  - Sensors which affect U_i from any pose are in set ω_i
- Incremental Line search (ILS):
  - When $\boldsymbol{i}$ moves, sensors in ω_i stay static, other sensors may move
    - Sensor $\boldsymbol{i}$ knows configuration of all sensors in ω_i
  - Sensor $\boldsymbol{i}$ chooses configuration $\boldsymbol{S}_i$={p,z,t} to maximize utility locally

$$\theta_p(i) = \max_{\theta_p \in [\theta_p^{min}, \theta_p^{max}]} U_i$$

(for pan angle)

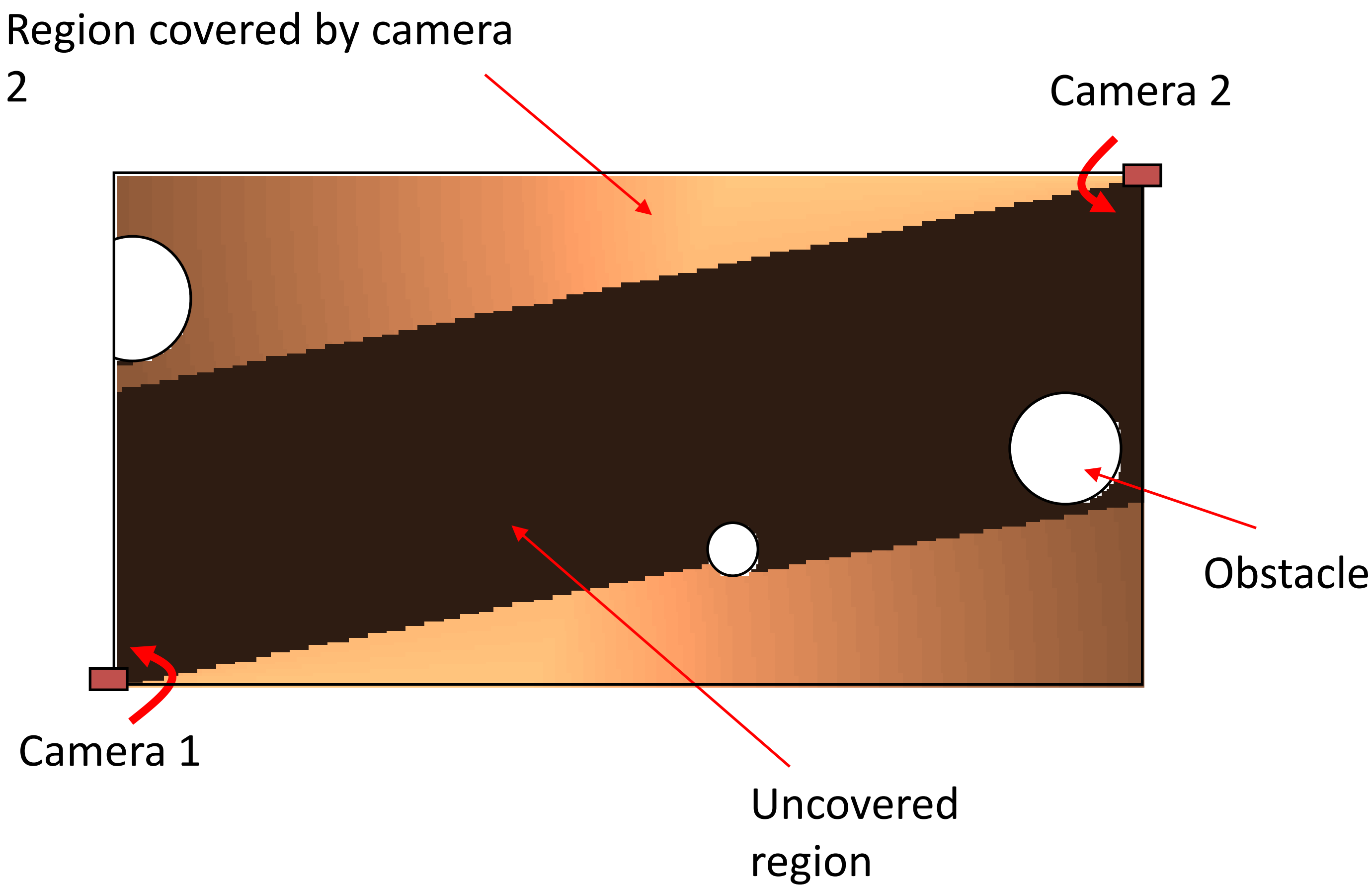
Protocol to Implement ILS

- Local Communication
  - Discover set ω_i based on location
  - Assume reliable message delivery to neighbors
- Coordination of motion within ω_i
  - Randomized contention: select moving sensor
    - Random wait and transmit Request To Update (RTU)
    - If no CONFLICT, RTU successful
  - Execute ILS to choose pose update
    - Transmit Update Finish (UF): informs neighbors of selected sensor pose



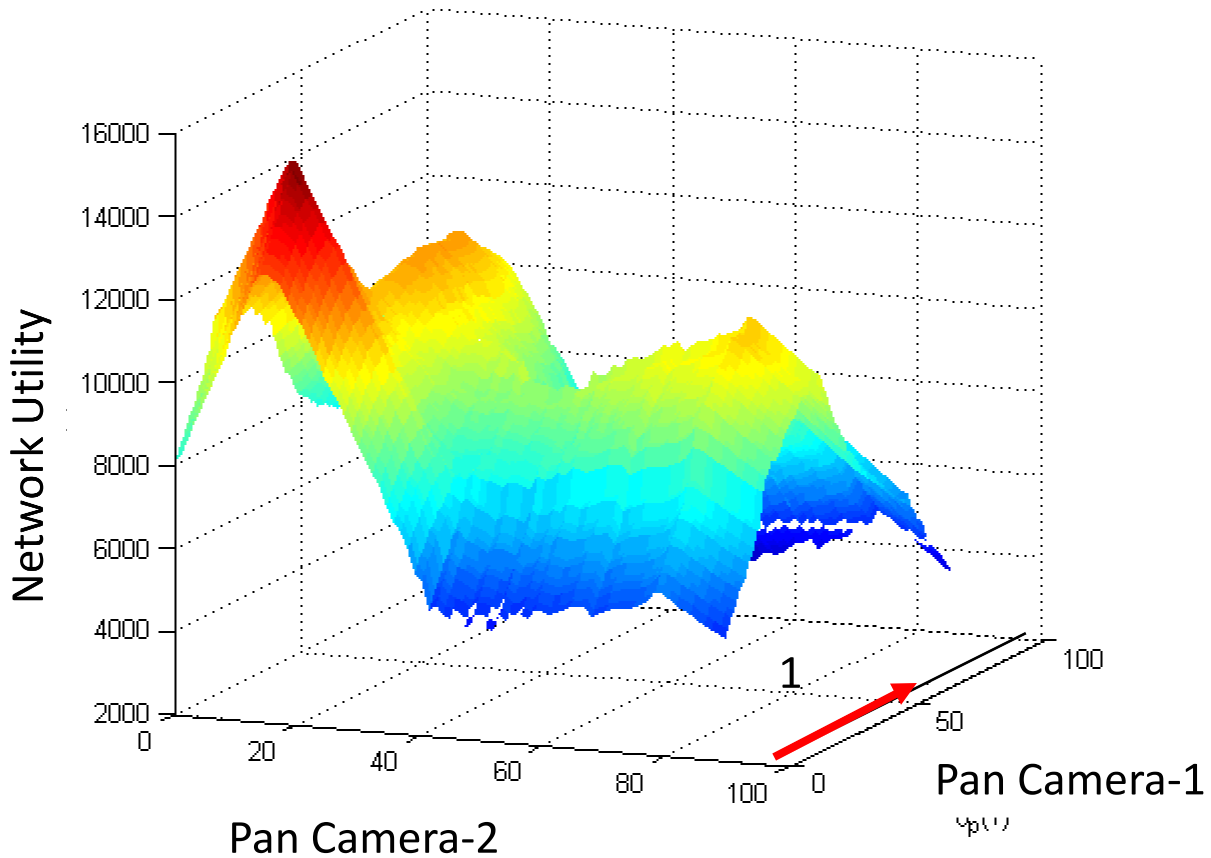
Algorithm details: "Virtual High Resolution for Sensor Networks," *ACM SenSys 2006.*

Visualizing ILS

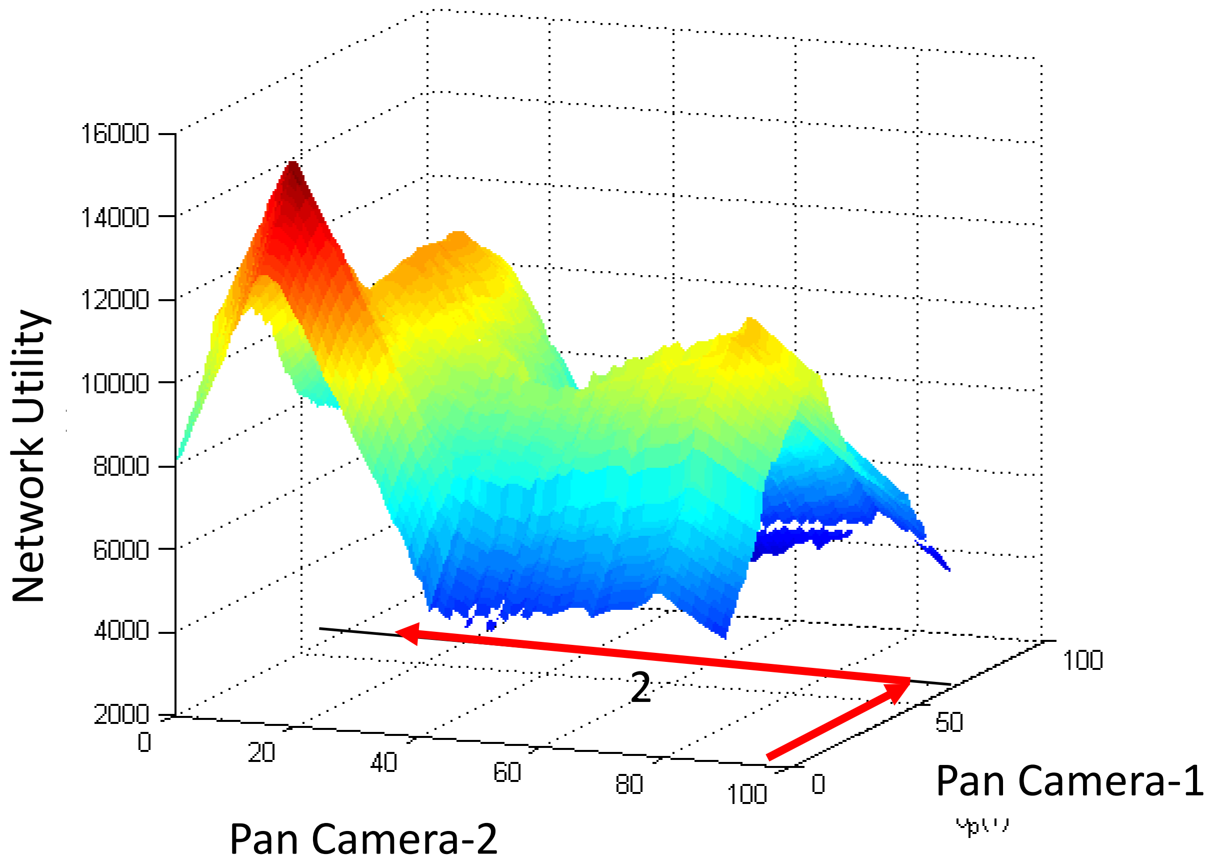


Visualizing ILS

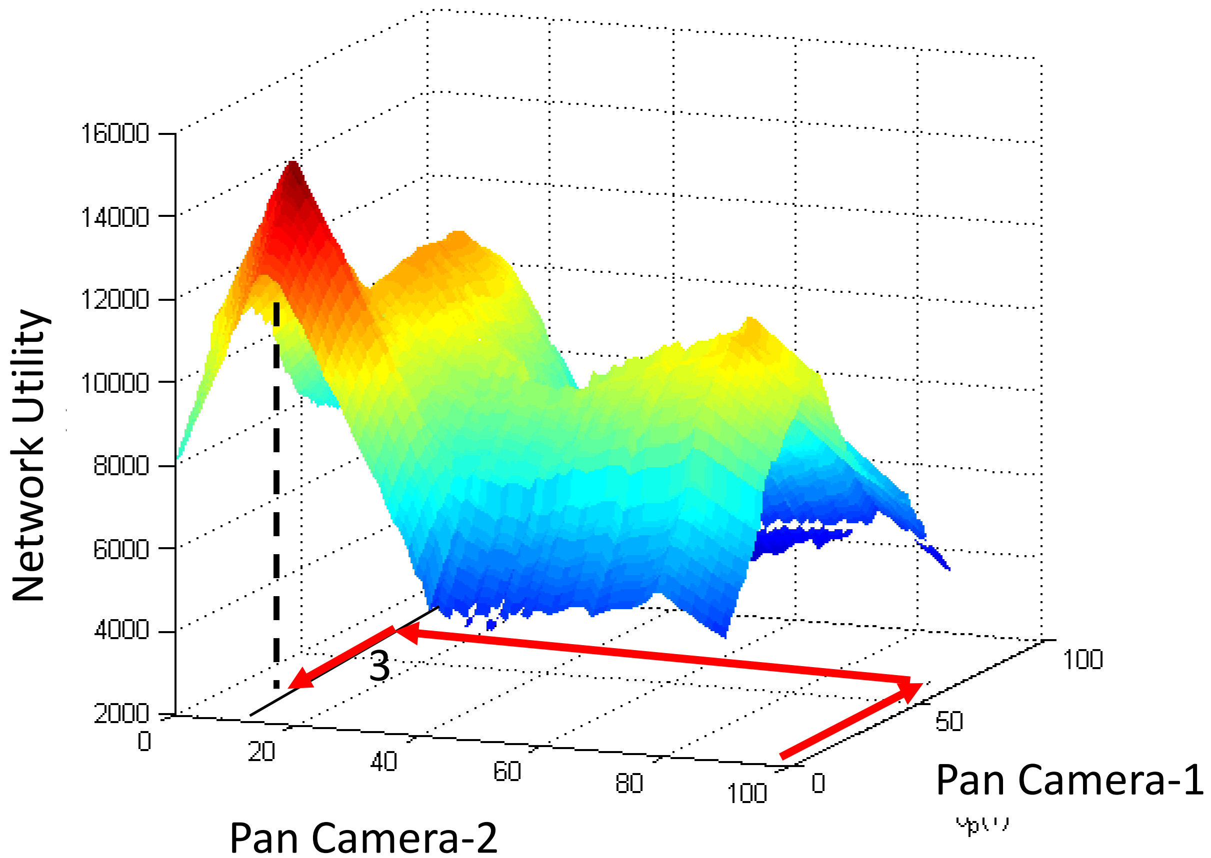
Utility function for all possible network configurations



Visualizing ILS



Visualizing ILS



Convergence of Distributed Algorithm

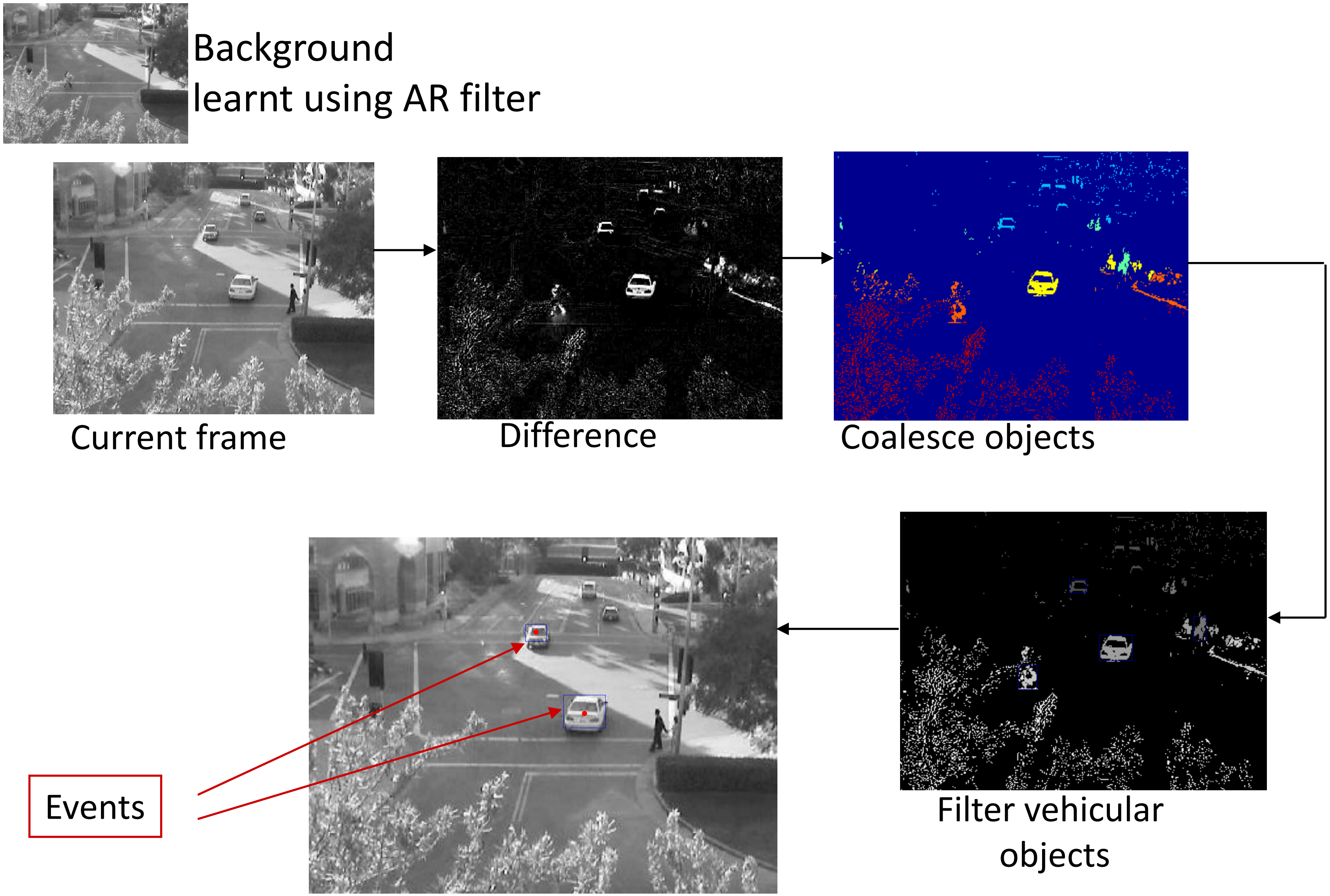
- **Theorem 1:** Under ILS, the network converges to a stable configuration in a finite number of steps
- Convergence speed
- Proximity to true optimum

Proof, Simulations: "Reconfiguration Methods for Mobile Sensor Networks," *ACM Transactions on Sensor Networks*

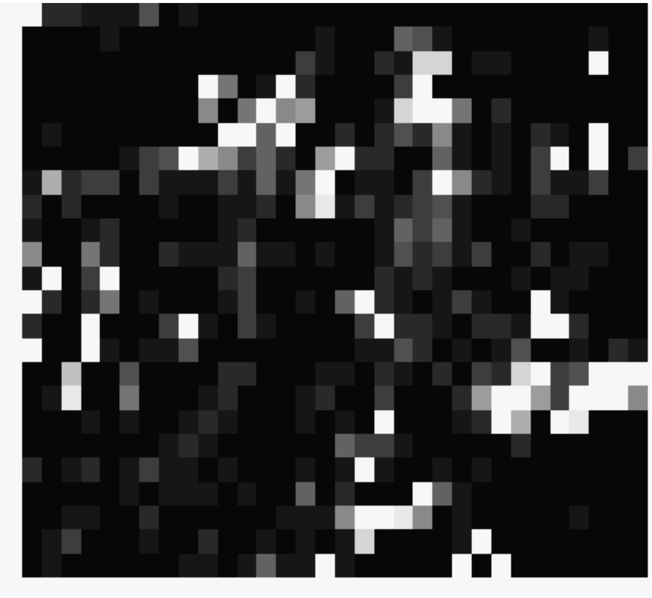
Experiments with Real World Data

- Application: Log images of vehicles approaching E4 building at UCLA
  - Detect vehicles at low resolution using motion detection
  - Minimize delay in high resolution coverage

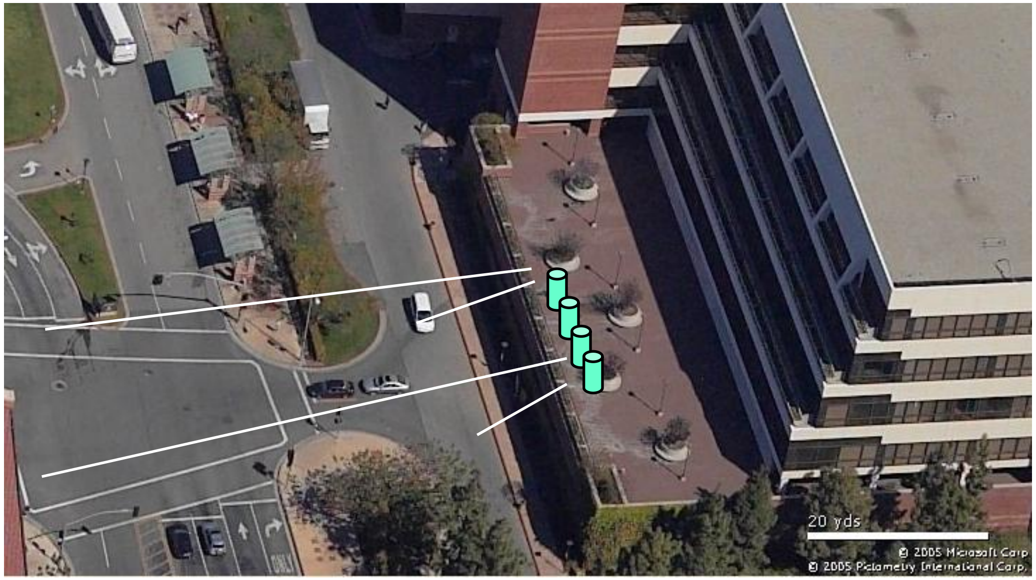
Event Detection



Experiment

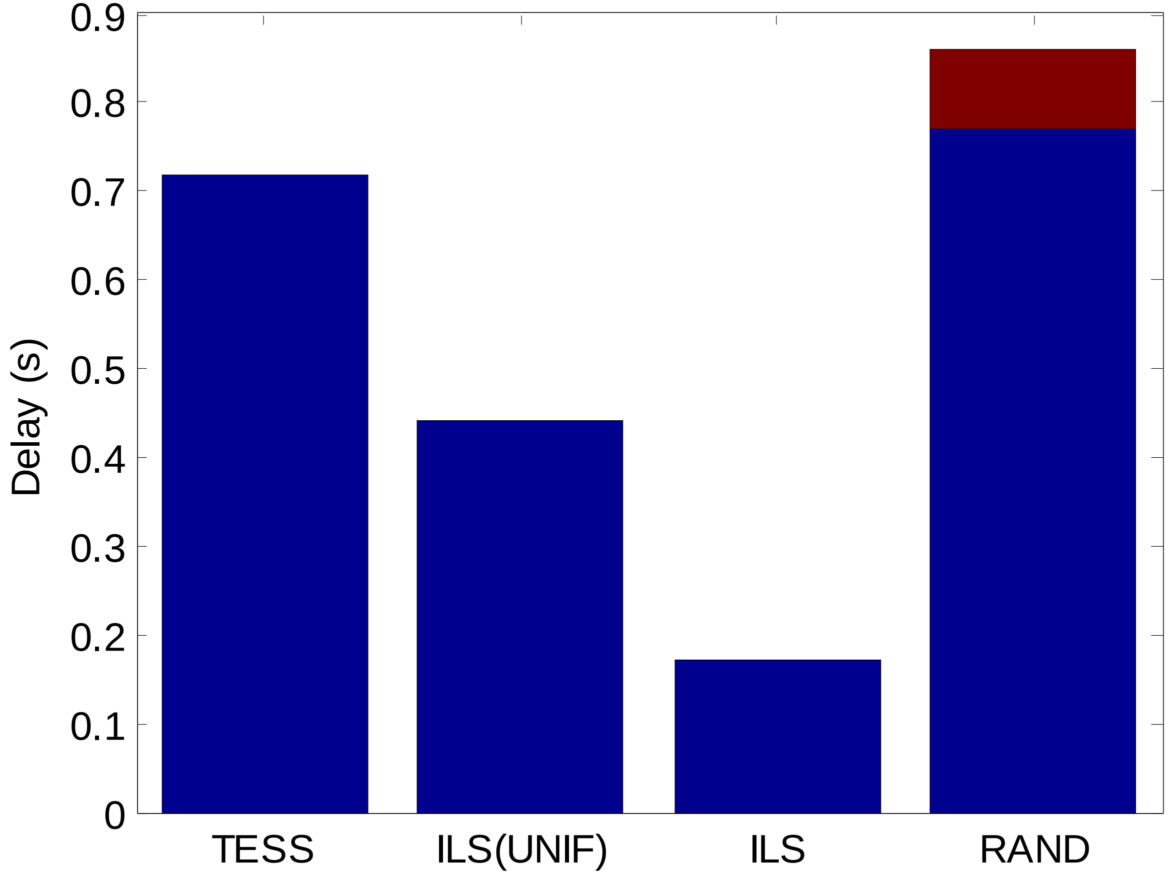


Event Trace



- A trace of 5000 events collected
  - Also used to learn event density
- ILS evaluation scenario
  - 4 cameras mounted on one side (camera calibration separated)
  - Actuation delay for event: measured for best camera covering it

ILS Delay Performance

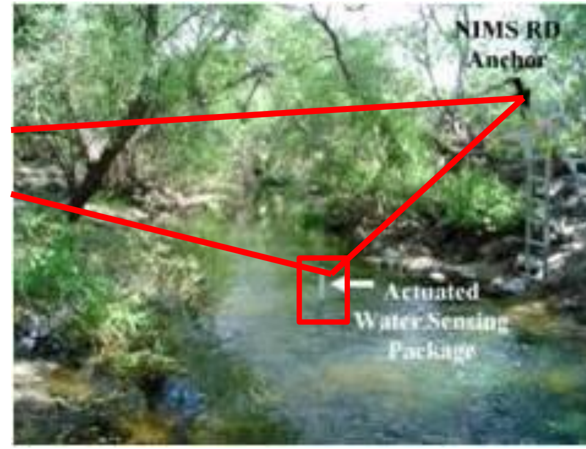


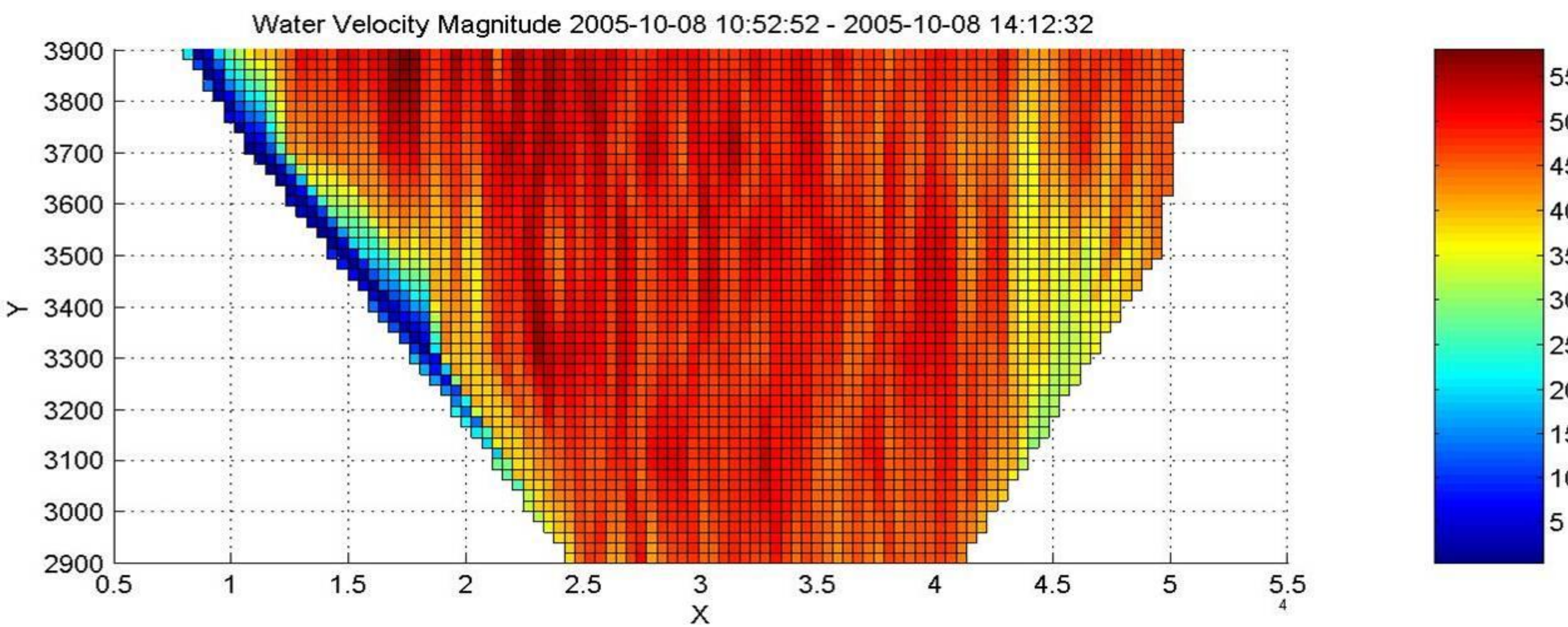
10 runs for RAND

More: "Reconfiguration Methods for Mobile Sensor Networks," *ACM Transactions on Sensor Networks*

Motility for Other Sensors

Point sensors: nitrate concentration measurement across river cross-section with constrained motion





Low Complexity Motion

- Can help achieve previously unachievable sensing resolutions
  - Orders of magnitude advantage over static networks
- Motion delay is a key issue
  - Leverage motion maximally within tolerable delay

Three New Capabilities

Performance Issue	New Capability
ENERGY SCARCITY	Harvest from Environment
SENSING RESOLUTION	Low Complexity Controlled Motion
DEPLOYMENT	Peer Production

Peer Production

- Sharing resources for mutual advantage
  - Multiple contributors and multiple users: costs efficiently amortized

WIKIPEDIA

SETI HOME

SecondLife
YouTube
…

Peer Produced Sensor Networks

Standalone deployments

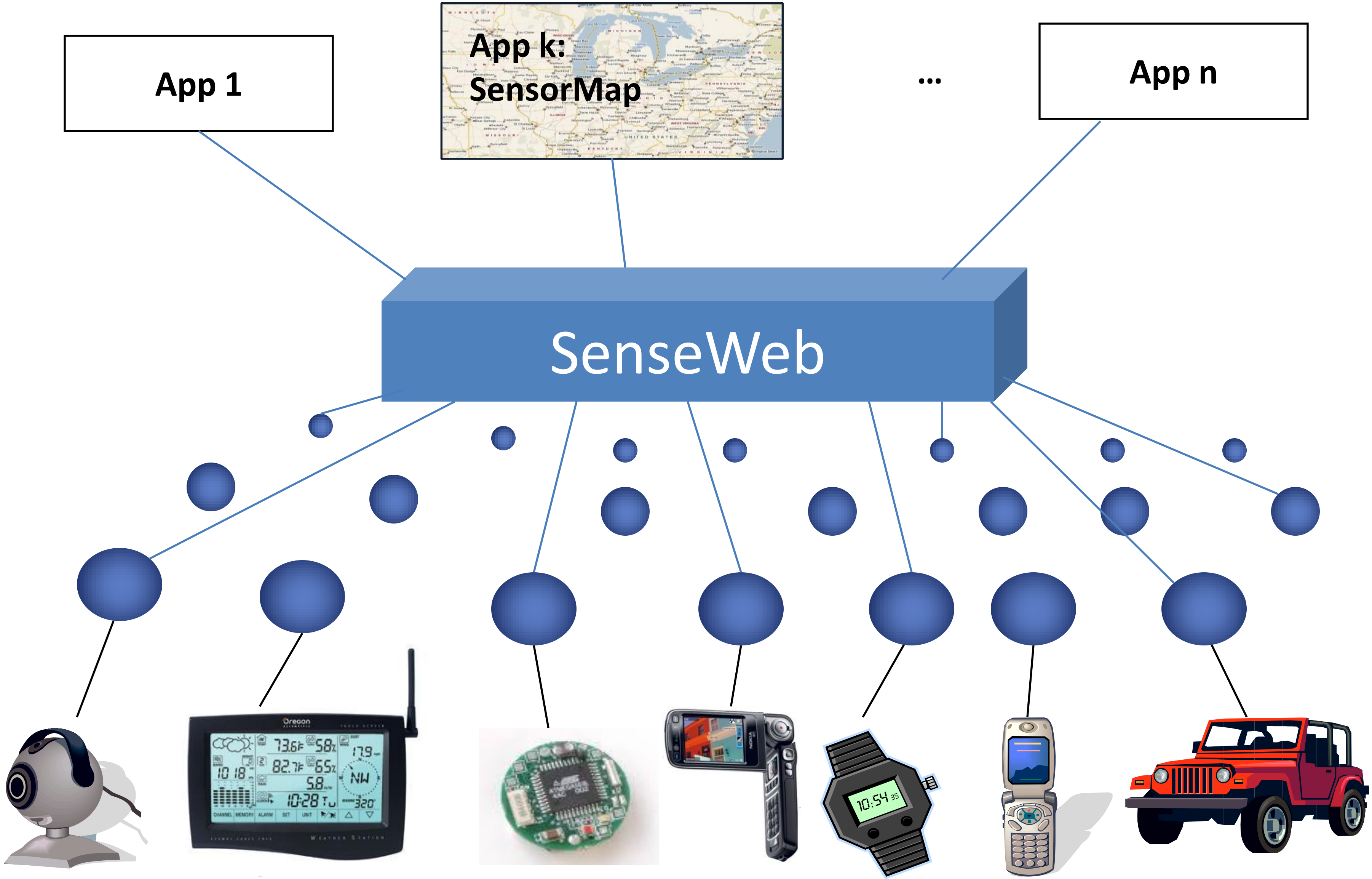
- Large scale deployment is challenging: cost, access rights, network/ power infrastructure, maintenance overhead
- Sensors and data are underused, replicated

SenseWeb: shared sensor networks

- Many sensor owners deploy at small scale
- Shared system yields large coverage
- Common sensors used by multiple apps

SenseWeb

http://atom.research.microsoft.com/sensormap



Challenges

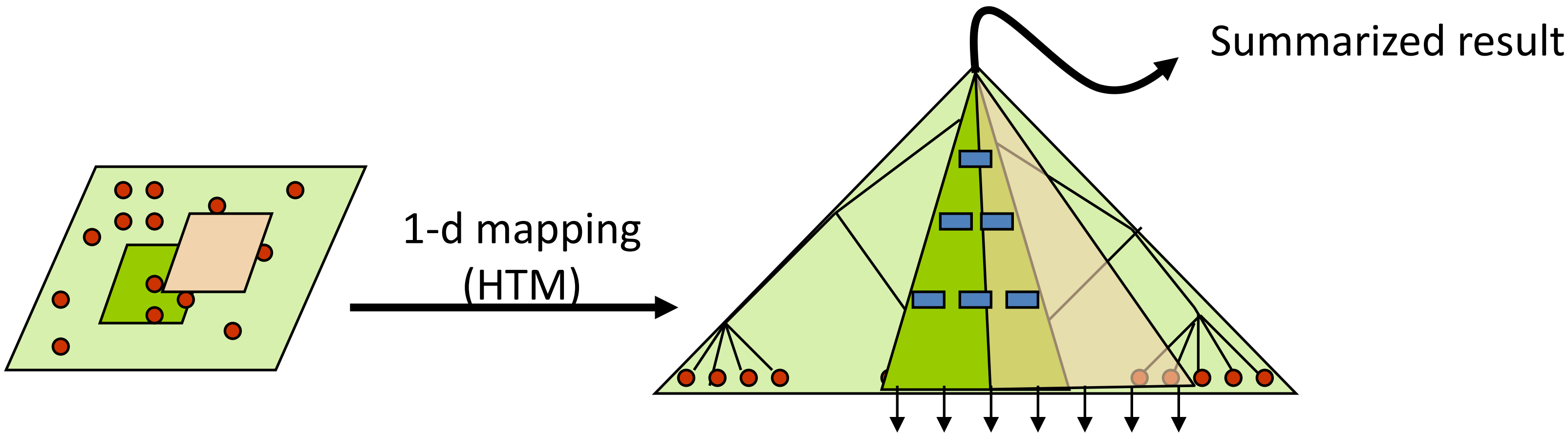
- Heterogeneity
  - Resource capability: bandwidth, power, computation
  - Willingness to share
  - Measurement accuracy
- Scalability
  - Streaming all raw data from all sensors to all applications not feasible
- Security and Privacy
- Data Verifiability, Trust

Data Re-use

- Sensor access provided through SQL-like queries
- Many applications may need similar data
  - within a tolerable latency of each other
  - From overlapping region
- Can cache data at server to reduce load on sensors and network
  - Overlap may be partial: computed aggregates may need partial new data

COLR-Tree (COLlection R-Tree)

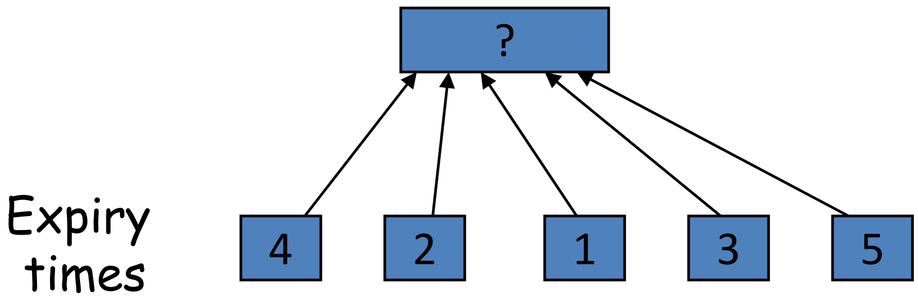
Index 2-D data with aggregates



- Minimizing sensor access
  - Cached data may have skewed distribution
  - Sample more from non-cached region
- Implemented on MS-SQL Server: usable with all SQL server capabilities

COLR-Tree: Aggregates

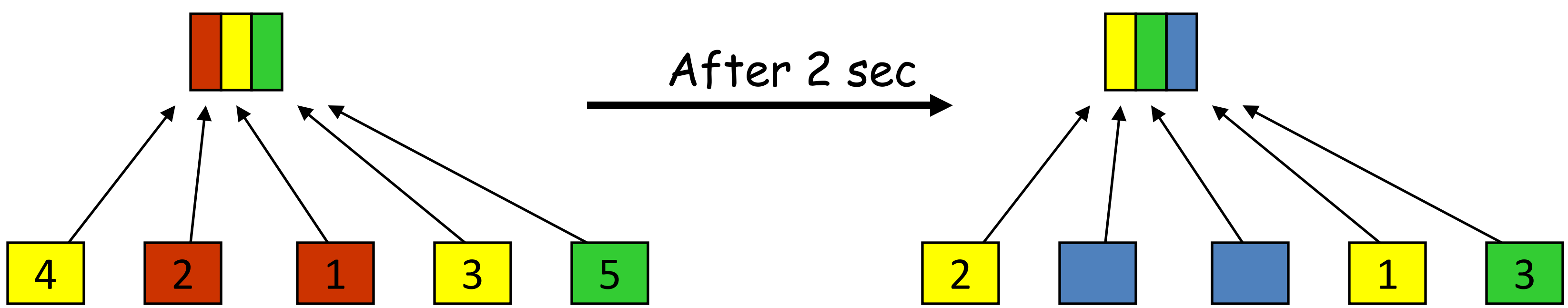
- Challenge: temporal aggregation



1: discard aggregate data after 1 sec (not much sharing)

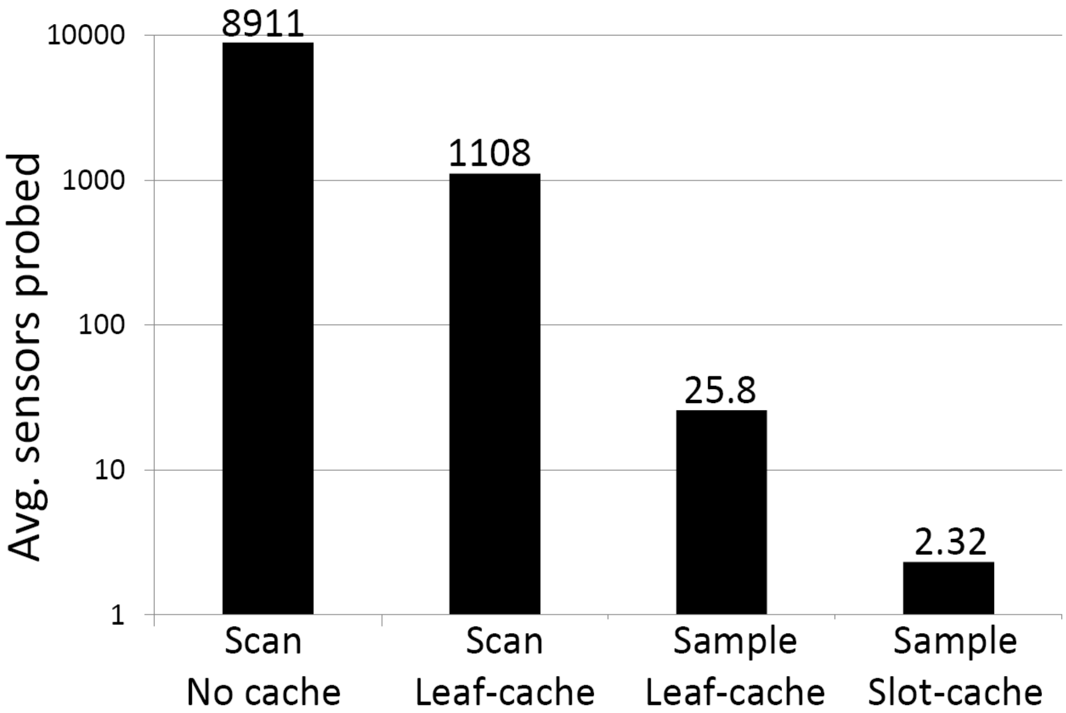
5: invalid after 1 sec

Solution: slotted aggregation



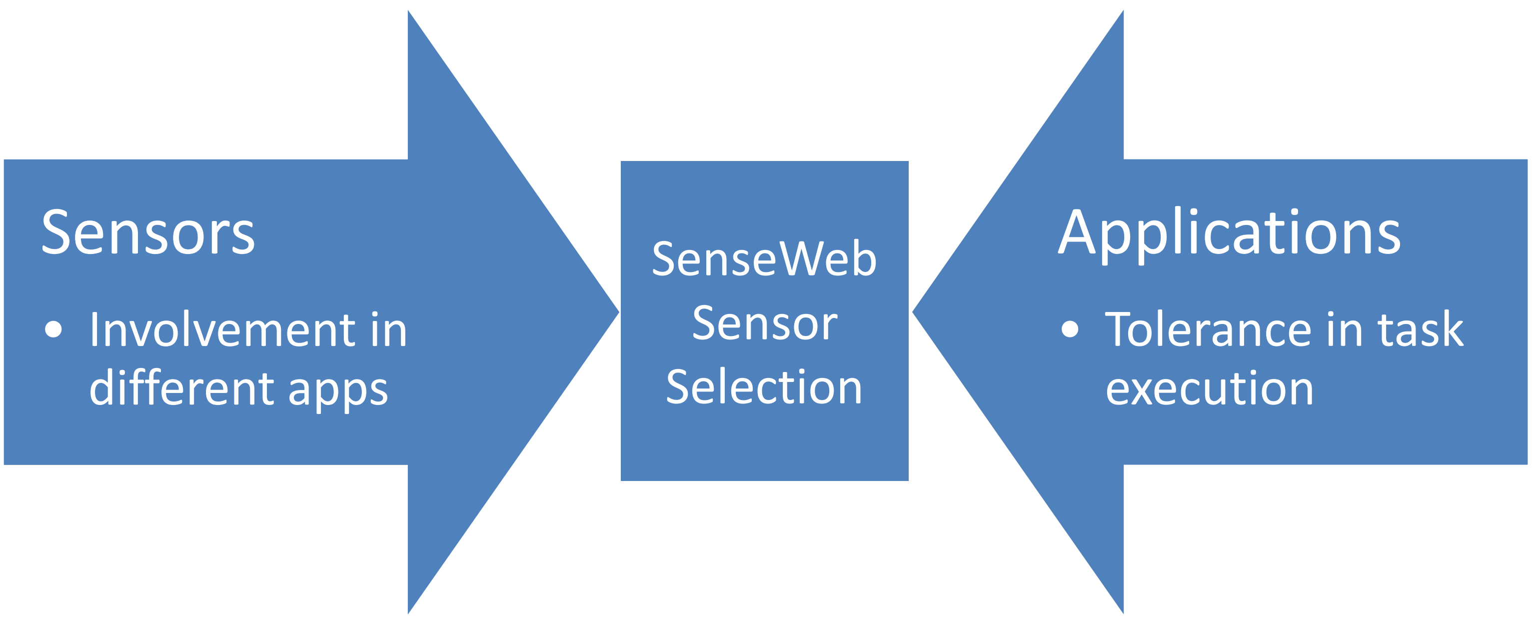
COLR-Tree Evaluation

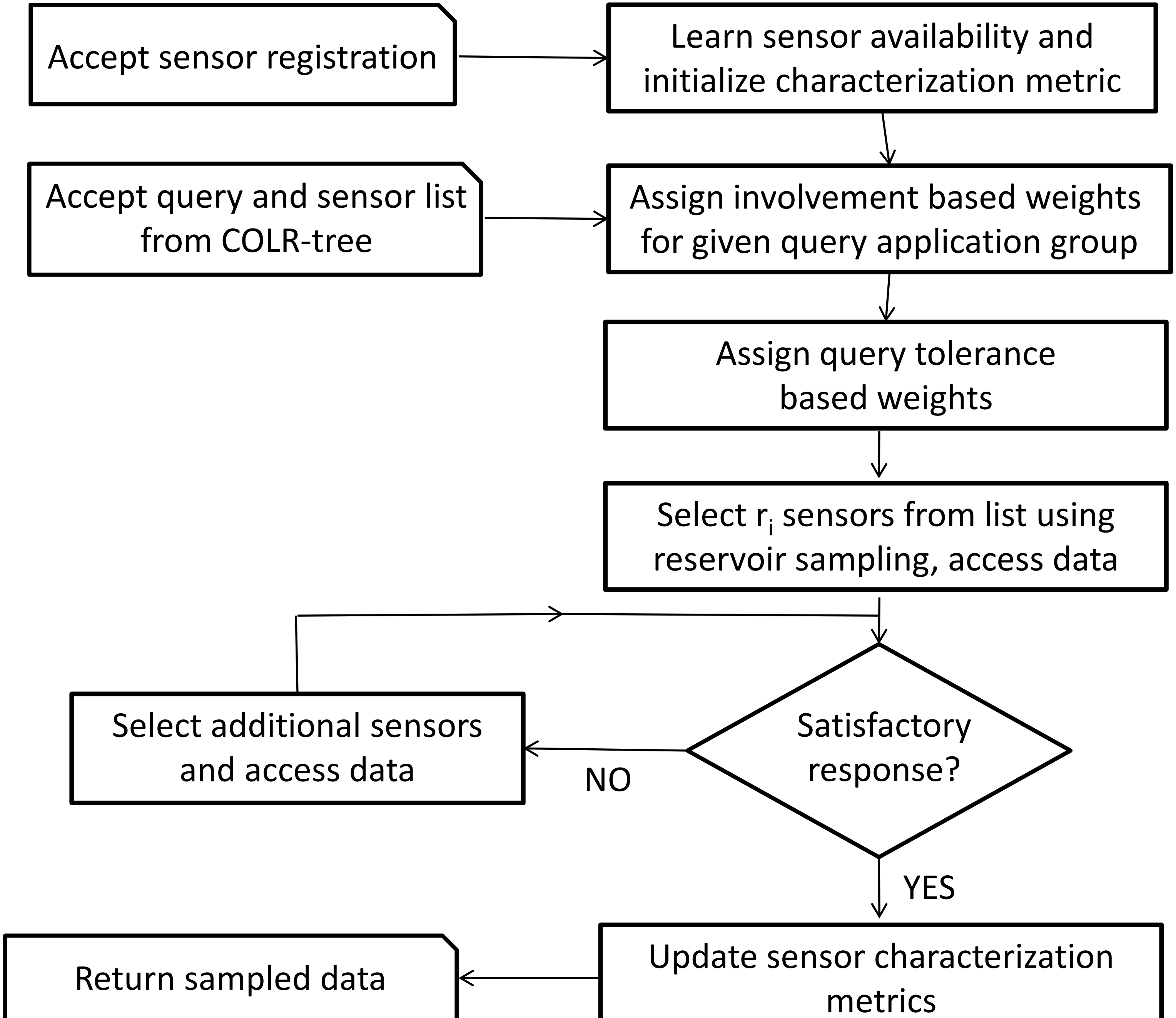
- Test data
  - 400K points from VE Yellow Pages
  - Regions queried: Virtual Earth usage trace



Tasking Heterogeneous Sensors

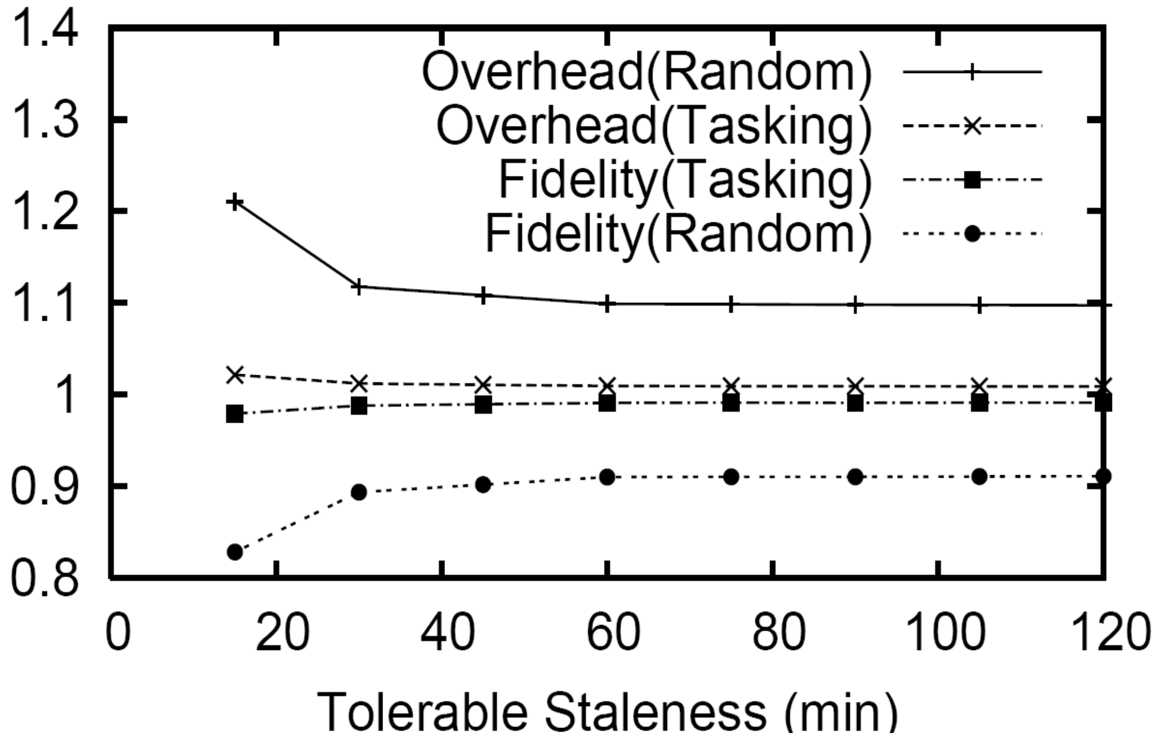
- Select uniformly rather than overloading the best sensors
- Leverage lower capability sensors when usable for a query
- Learn and adapt to sensor characteristics: availability, bandwidth
- Weighted reservoir sampling
  - Weighted random selection, with desired number of sensors




Accept sensor registration
Learn sensor availability and initialize characterization metric
Accept query and sensor list from COLR-tree
Assign involvement based weights for given query application group
Assign query tolerance based weights
Select r_i sensors from list using reservoir sampling, access data
Select additional sensors and access data
NO
Satisfactory response?
YES
Return sampled data
Update sensor characterization metrics

Tasking Algorithm Performance

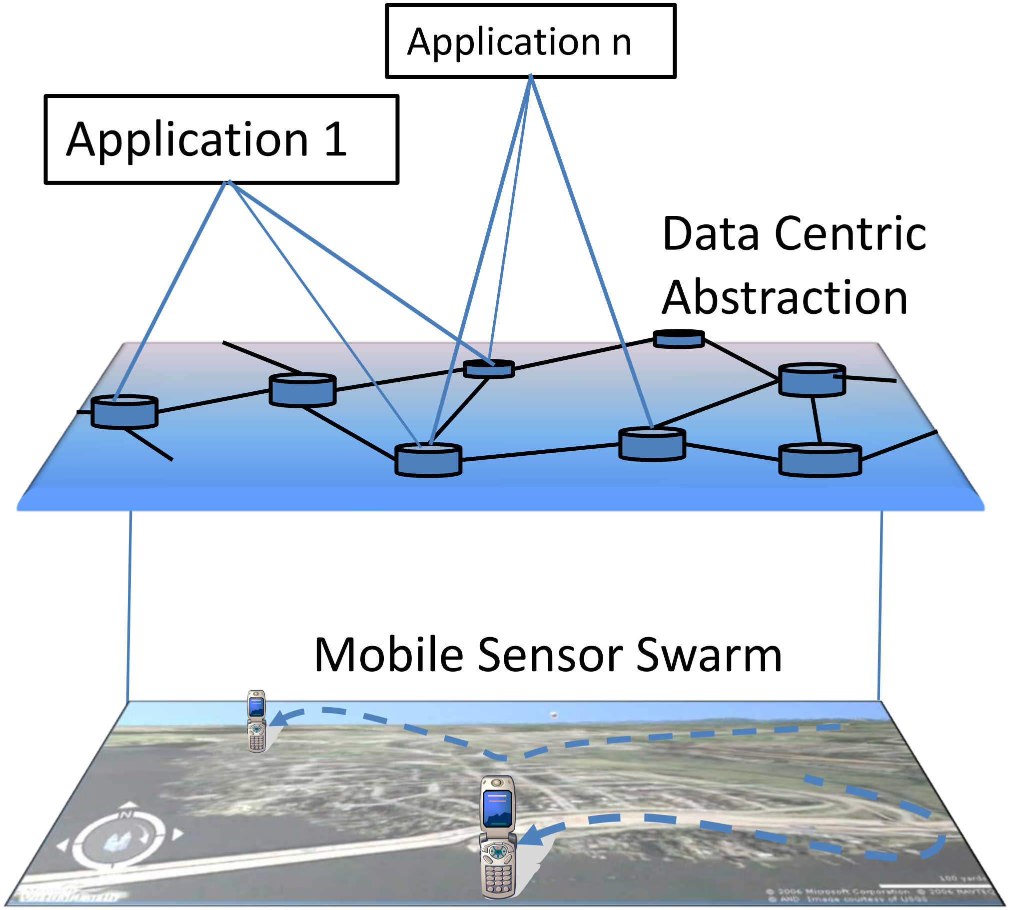
- Test on USGS stream water sensors
  - Random selection vs. Weighted reservoir sampling



Mobile Contributors in SenseWeb

+ More coverage
- Hard for application to track relevant devices

- **Solution:** data centric abstraction
  - Location based indexing
    - using GPS, cell-tower triangulation, content based location



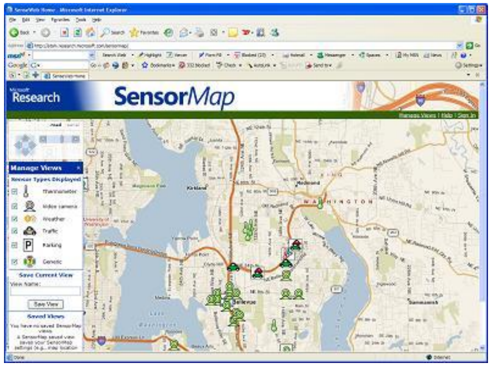
Implementation: “Demo: Building a Sensor Network of Mobile Phones," IPSN 2007.
Details: “Location and Mobility in a Sensor Network of Mobile Phones,” ACM NOSSDAV 2007.

Peer Production Summary

- Peer production can make low cost sensor networks possible for many apps.
- Design system to work with highly heterogeneous resources and dynamically varying availability
  - Adaptively allocate resources to tasks and services

Networked Embedded Computing

http://atom.research.microsoft.com/sensormap/



SenseWeb

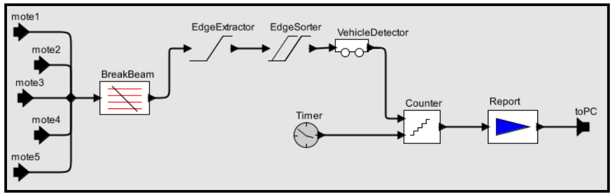
Shared sensor networks

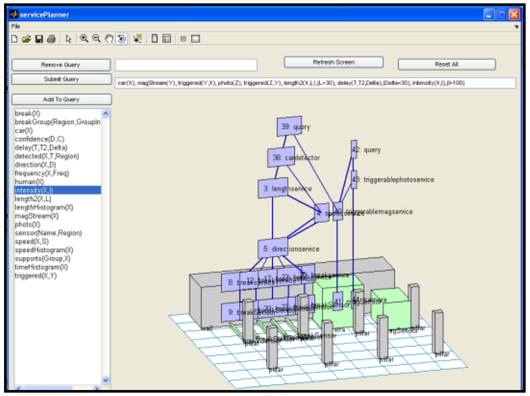
mPlatform

reconfig proc., comm, storage

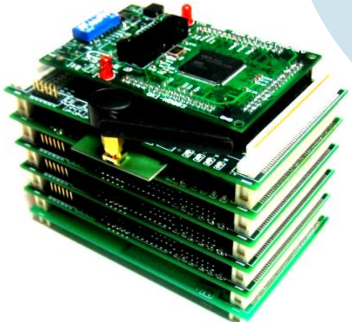
MSRSense

Composing apps on sensor streams





(open source. Download from http://research.microsoft.com/nec/msrsense/)



ftp://ftp.research.microsoft.com/pub/tr/TR-2006-142.pdf