



Energy Efficient Map Interpolation for Sensor Fields Using Kriging

Brian Harrington, Yan Huang, Jue Yang, and Xinrong Li

[brh, huangyan, jy0074, xinrong]@unt.edu

University of North Texas



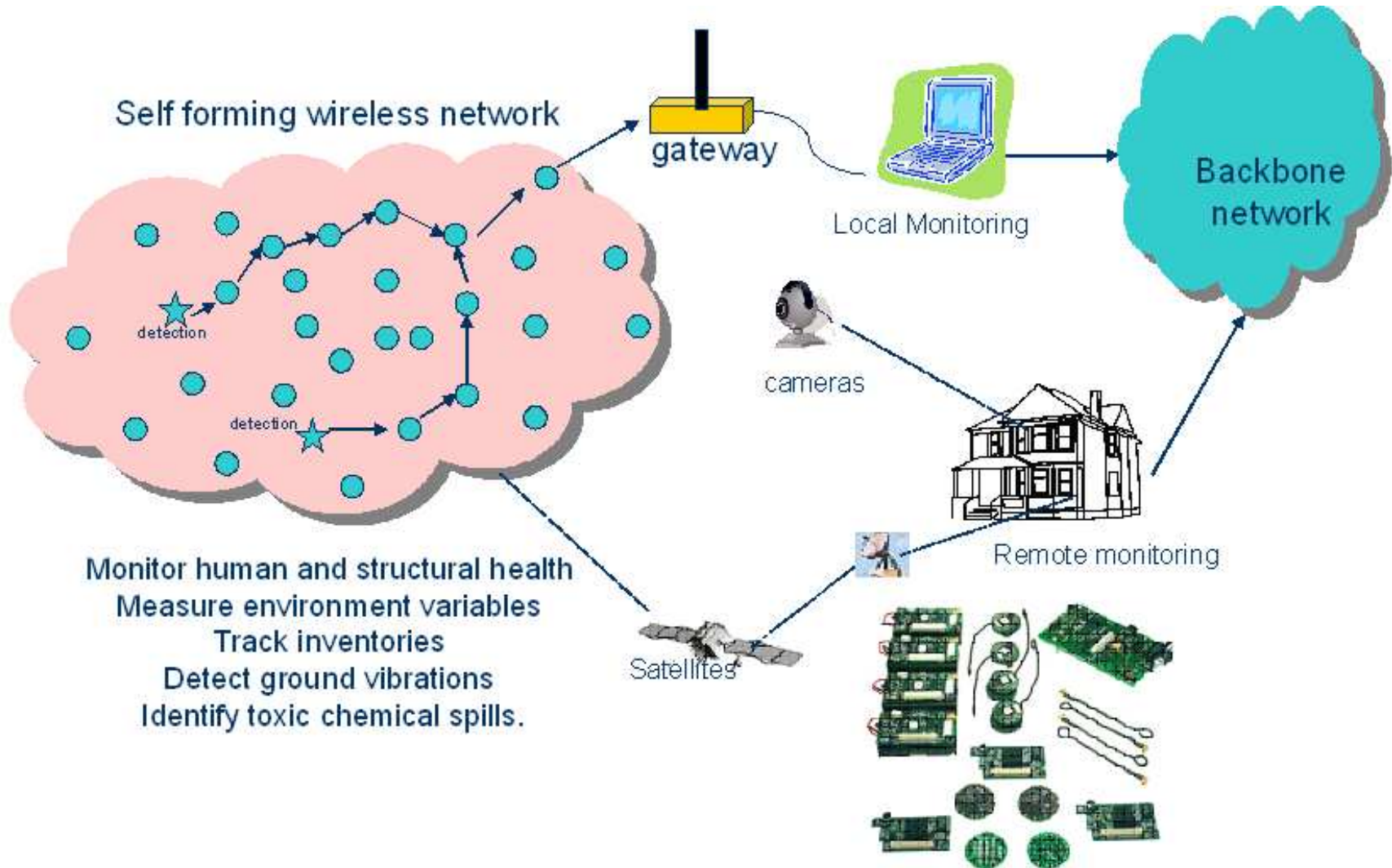
Overview



- Motivation
- Related work
- Problem statement
- E2K framework
- Choosing interpolation methods
- Temporal E2K
- Experimental results
- Conclusion



Motivation



Motivation

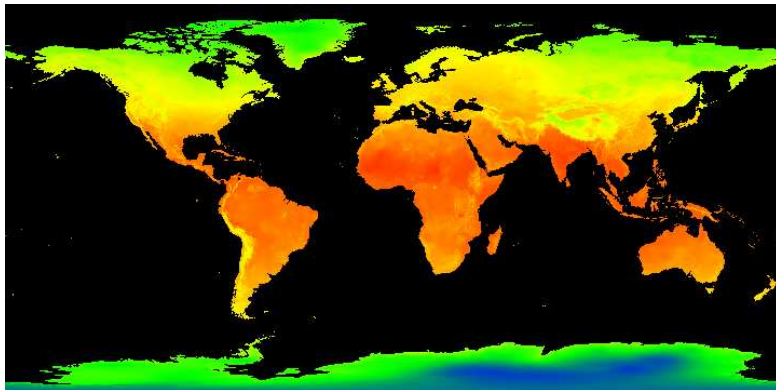


- Many current sensor work focuses on in-network aggregations
- Aggregates, e.g., sum, min, max, are of limited usefulness to domain scientists
- Many natural phenomena are best represented as a continuous surface over the sensor field, e.g.:
 - Temperature
 - Hydraulic head
 - Soil moisture
 - Ocean current velocity

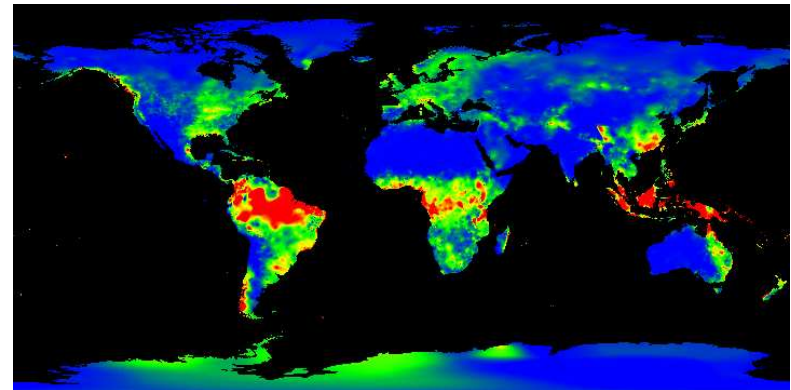


Motivation (cont)

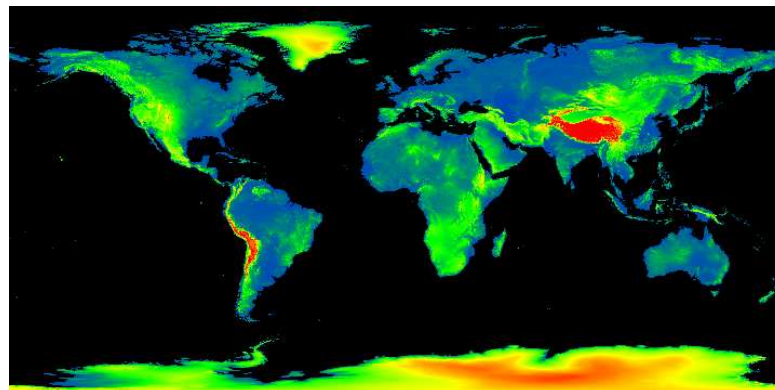
- Example surface maps:



Temperature



Precipitation



Elevation

Overview



- Motivation
- **Related work**
- Problem statement
- E2K framework
- Choosing interpolation methods
- Temporal E2K
- Experimental results
- Conclusion



Related Work



- Four broad categories of sensor work:
 - In-network aggregation
 - Correlation based sensor reporting
 - Data compression
 - Interrogation



In-Network Aggregation



- Use simple in-network computation to reduce messaging cost
- TAG system [Madden et al. 2002a]:
 - Aggregation tree is created during query insertion
 - Values aggregated from child to parent on way to root
 - Beneficial for distributive (e.g. sum, min, max) and algebraic (e.g. average) aggregations
- Others include [Madden et al. 2002b, Considine et al. 2004, Trigoni et al. 2004, Sharifzadeh and Shahabi 2004].



Correlation Based Sensor Reporting

- Utilize spatial and/or temporal correlation
- Temporal:
 - Approximate caching [Olston et al. 2001] relies on cached values
 - In [Goel et al. 2004] a function based on past values is used
- Spatial
 - Cluster [Goel et al. 2004, Ali et al. 2005, Vuran et al. 2004] a group of sensor nodes according to spatial proximity
 - Snapshot [Kotidis 2005] uses heuristics for selecting representatives
- Spatial and temporal was used in [Chu et al. 2006].

Data Compression and Interrogation

- Data compression:
 - Compress individual readings [Deligiannakis et al. 2004]
 - Compress values along route [Krishnamachari et al. 2002, Pradhan and Ramchandran 1999, Scaglione and Servetto 2002]
- Interrogation
 - Specifically ask for given values using model at sink [Deshpande et al. 2004]
 - Event-detection is a problem

Overview



- Motivation
- Related work
- **Problem statement**
- E2K framework
- Choosing interpolation methods
- Temporal E2K
- Experimental results
- Conclusion



Problem Statement



- Once we have sensor readings, map interpolation becomes trivial
- To get a map:
 - Grid the space at desired spatial resolution
 - Use sensor reading to interpolate values for each cell
- General problem is then:
 - **SELECT * FREQUENCY f WITHIN ϵ**



Problem Statement (cont)

- Let S be a set of spatially distributed sensors and C be the sink
- These sensors monitor an attribute A at some time instance $t \in T$
- For $s \in S$, let $Z_t(s)$ be the value of A for sensor s at time t
- Let $Z_t^*(s)$ be the value C estimates for $Z_t(s)$
- Let $\epsilon > 0$ be the error threshold
- Devise an algorithm for C and the sensors in S
 - Such that $\forall s, t, |Z_t(s) - Z_t^*(s)| < \epsilon$
 - With objective of reducing total messaging cost

Overview



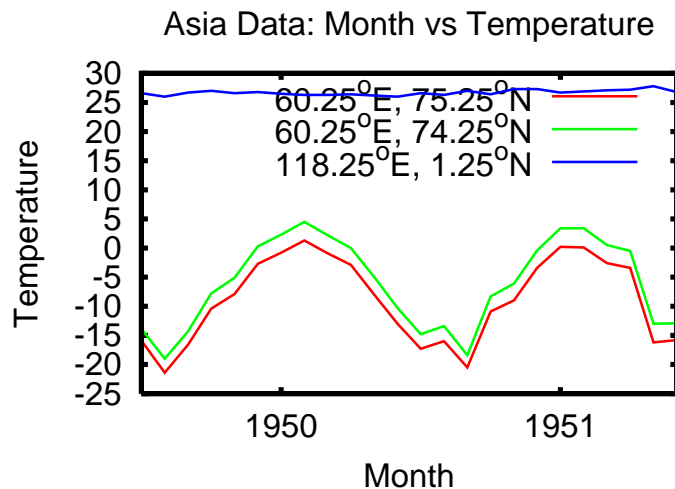
- Motivation
- Related work
- Problem statement
- **E2K framework**
- Choosing interpolation methods
- Temporal E2K
- Experimental results
- Conclusion



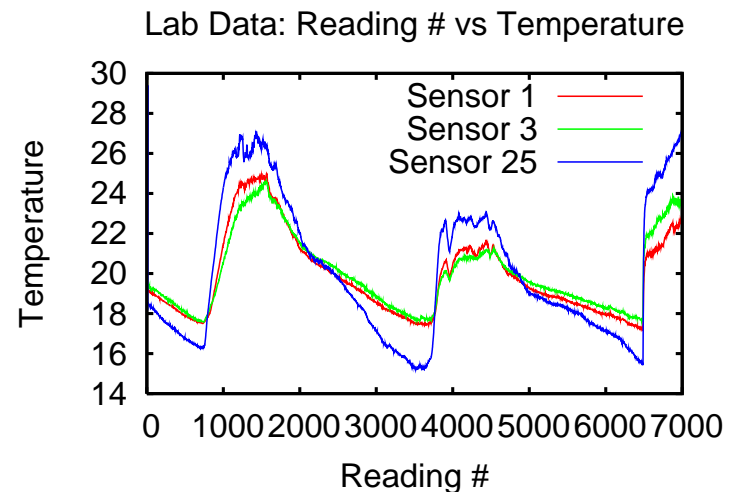
E2K Framework



- How do we capture spatial autocorrelation?
- Tobler's first law of geography [Cressie 1991] states that in space everything is related to everything else, but nearby things are more related than distant things.



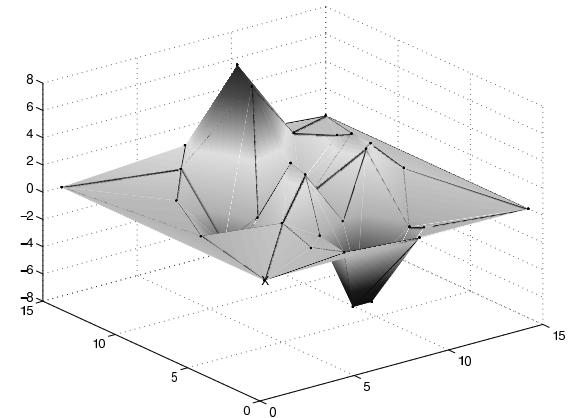
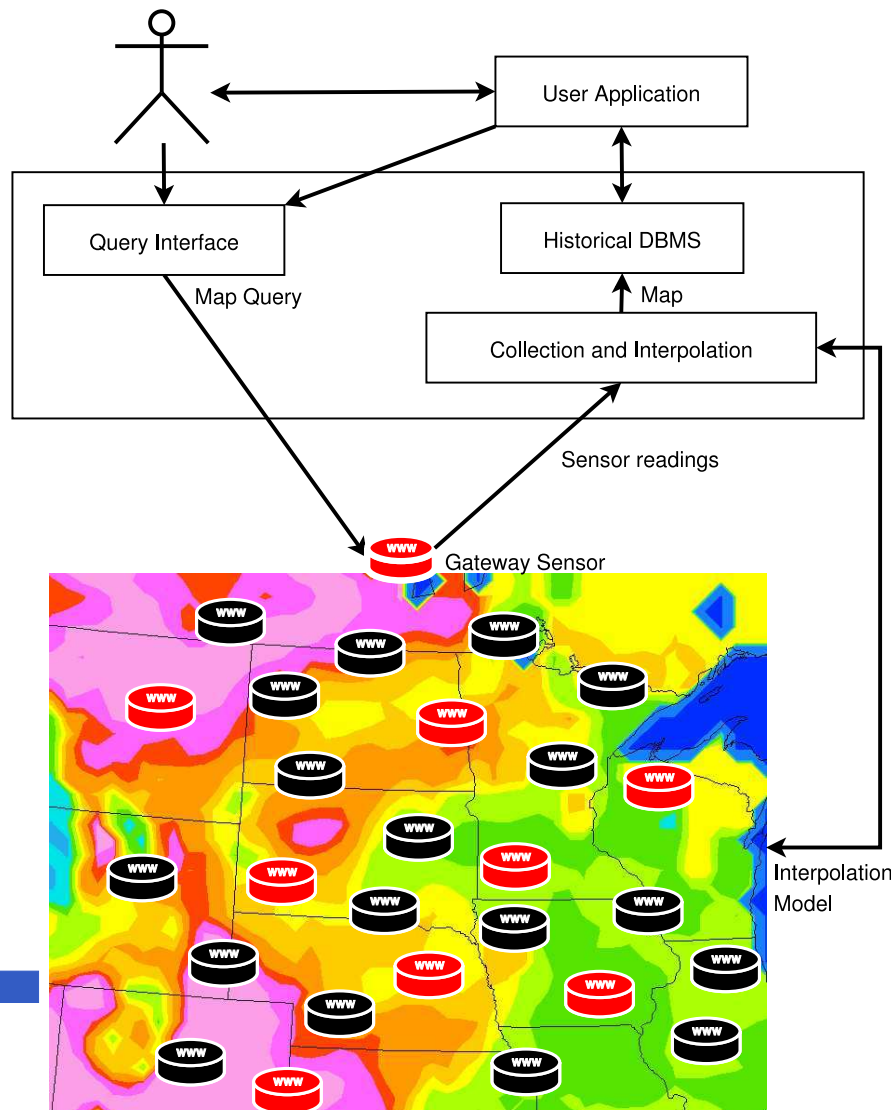
Asia temperature trends



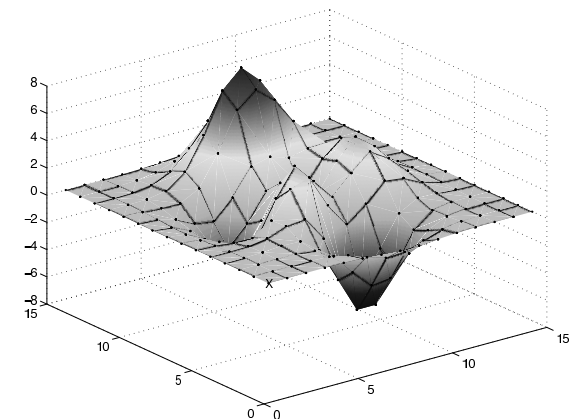
Lab temperature trends



System Overview



Map interpolated by a subset of the sensors



Map interpolated by all sensors



Challenges



- Need to find balance between coordination cost and the savings from having fewer sensors report
- An error guarantee for each sensor is desirable
- Competing goals:
 - Minimize the number of sensors that report
 - Minimize coordination costs among sensors
 - Allow the sink to interpolate readings for non-reporting sensors within an error threshold ϵ



Simple Methods



- A naive probabilistic approach would let each sensor report with a probability p .
 - (+) No coordination cost
 - (-) No error bound
- An alternative would be to have all sensors send their value to neighbors. Each sensor interpolates its own reading using the readings from its neighbors. If the interpolated value deviates from the real reading by more than ϵ , then the sensor reports.
 - (-) High coordination cost
 - (-) No error bound



E2K: Sensor (s_0) Algorithm

- 1: $Z_t(s_0) \leftarrow$ value of A for this sensor at time t .
- 2: $p \leftarrow \frac{n_{desired}}{n_{current}}$
- 3: $rand \leftarrow$ a random number $\in [0, 1]$
- 4: **if** ($rand < p$) **then**
- 5: {Round 1}
- 6: Report ($Z_t(s_0), round_1$) to the central site and neighbors within distance r .
- 7: **else**
- 8: {Round 2}
- 9: $R \leftarrow$ the set of readings from sensors within distance r that reported in first round.
- 10: $Z_t^*(s_0) \leftarrow interp(R)$
- 11: **if** ($|Z_t^*(s_0) - Z_t(s_0)| \geq \epsilon$) **then**
- 12: Report ($Z_t(s_0), round_2$) to the central site.
- 13: **end if**
- 14: **end if**

E2K: Central Site Algorithm

- 1: Let S be the set of all sensors.
- 2: $R_1 \leftarrow$ the set of values received from sensors for attribute A at time t in round 1.
- 3: $R_2 \leftarrow$ the set of values received from sensors for attribute A at time t in round 2.
- 4: **for all** $s \in S$ **do**
- 5: **if** (s reported a value) **then**
- 6: $Z_t^*(s) \leftarrow$ value of s in $R_1 \cup R_2$
- 7: **else**
- 8: $R_n \leftarrow$ the subset of R_1 within distance r of s .
- 9: $Z_t^*(s) \leftarrow \text{interp}(R_n)$
- 10: **end if**
- 11: **end for**

E2K: Analysis



- Probabilistic first round and use neighbors in second round
 - (+) No coordination cost
 - (+) Error bound
- First round has no coordination cost
- In second round, coordination is avoided by only looking at neighbors that reported in the first round
- Second round provides error bound



E2K: Analysis (cont)

Lemma 1 (Conditional Zero Coordination Cost). *In E2K, the number of messages sent in order to coordinate sensors in deciding which ones need to report and maintain an error bound is 0 when the spatial interpolation neighborhood is less than or equal to the radio range.*

E2K: Analysis (cont)

Lemma 2 (Error Bounding). *For any sensor s , let $Z_t(s)$ be the actual value and $Z_t^*(s)$ be the estimated value of s for attribute A at time t . The $|Z_t^*(s) - Z_t(s)| < \epsilon$ using the proposed algorithms for each sensor and the central site.*

Proof. There are two cases to consider:

1. If s reported its value, then $Z_t^*(s) = Z_t(s)$ which implies $|Z_t^*(s) - Z_t(s)| = 0 < \epsilon$ since from the problem statement $\epsilon > 0$.
2. If s did not report its value, then $Z_t^*(s) = \text{interp}(R)$ where R is the set of values that reported in the first round and that are within distance r from s . This is the same as the estimated value used in the second round for sensor s . If $|Z_t^*(s) - Z_t(s)| \geq \epsilon$, then s would have reported. Therefore $|Z_t^*(s) - Z_t(s)| < \epsilon$.

Overview



- Motivation
- Related work
- Problem statement
- E2K framework
- **Choosing interpolation methods**
- Temporal E2K
- Experimental results
- Conclusion

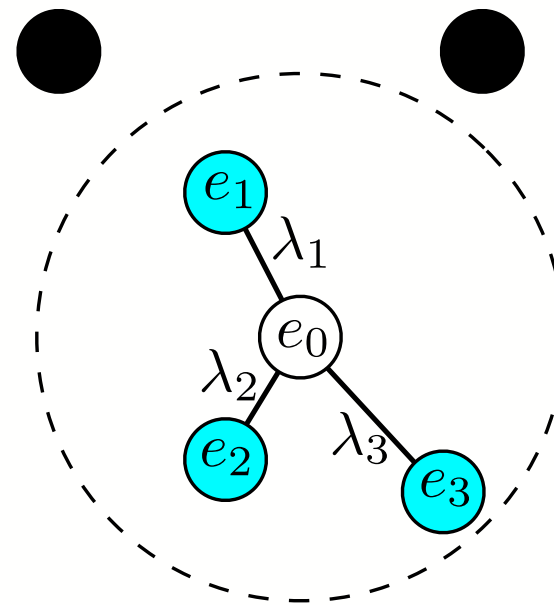


Choosing Interpolation Methods

- Interpolation method can greatly impact performance
- Let $Z(e)$ be a random function at a location e of a sensor field
- The value for location e_0 is estimated using neighbors within distance r as

$$Z^*(e_0) = \sum_{i=1}^n \lambda_i Z(e_i)$$

- Example:



$$Z^*(e_0) = \lambda_1 Z(e_1) + \lambda_2 Z(e_2) + \lambda_3 Z(e_3)$$

Assigning Weights



- How do we assign weights?
- It is desirable for sum of weights to be 1.
- *Simple Average*: all neighbors have the same weight
- *Inverse Distance*: inverse of the distance
($\lambda_j = \frac{1/d_j}{\sum_{i=1}^n 1/d_i}$)
- These methods do not consider the spatial structure of the data.



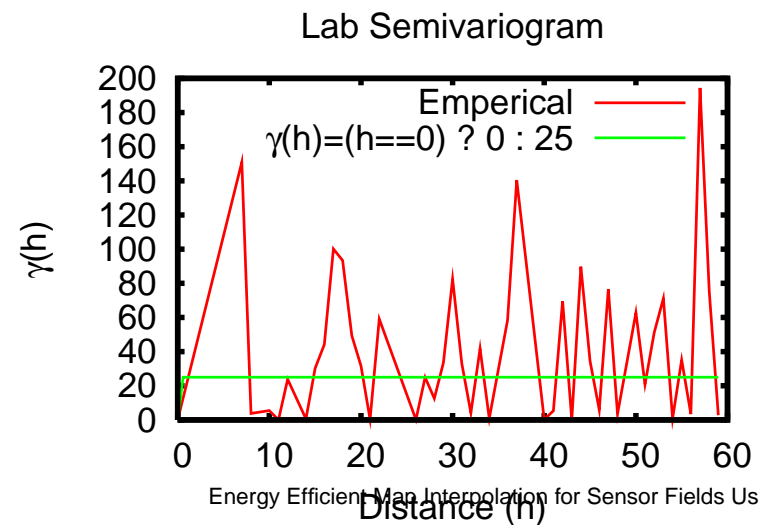
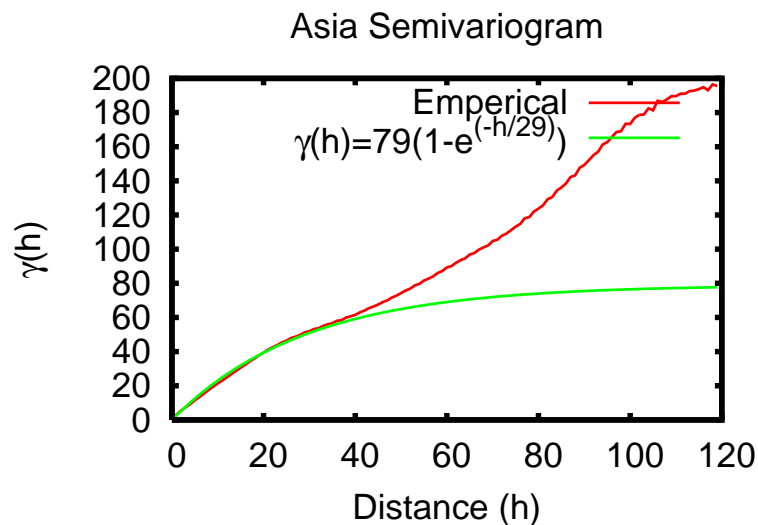
Kriging

- Kriging [Cressie 1991, Wackernagel 1995] is widely used in many natural science domains
- It is a best fit linear unbiased estimator of a spatial variable
- It estimates a value at a location of a region for which a covariance/variogram is known
- We assume $Z(e)$ is *second-order stationary*
 - Expected value $E[Z(e)] = m$, where m is the mean, for any point of the domain
 - Covariance between any pair of locations depends only on the vector h that separates them

Kriging (cont)

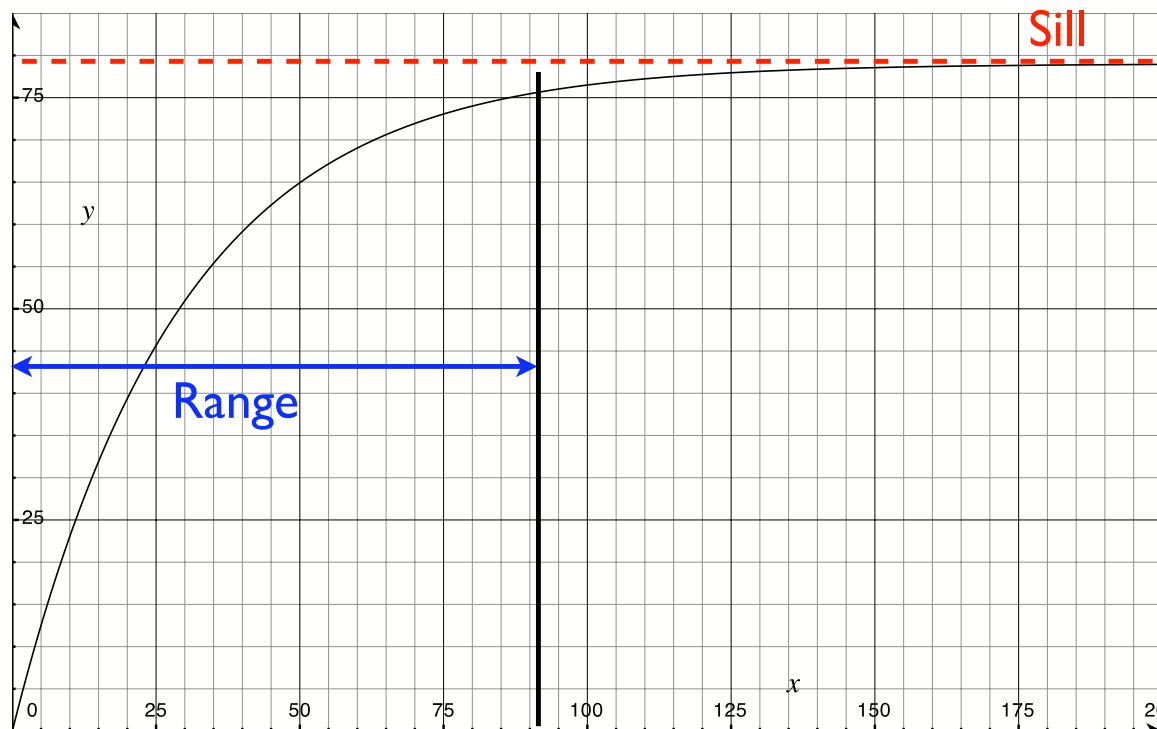
- Kriging assigns weights according to a known or estimated covariance/variogram function which captures the spatial autocorrelation
- Empirical estimate:

$$\hat{\gamma}_z(h) = \frac{1}{2N_h} \sum_{i=1}^{N_h} (z(e_i) - z(e_i + h))^2$$



Variogram

Important parameters: **range** and **sill**



Optimal Weights

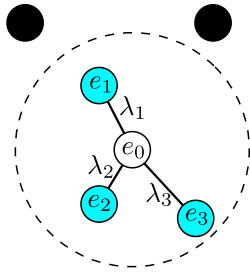
The error variance is:

$$\begin{aligned}\sigma_E^2 &= E[(Z^*(e) - Z(e))^2] \\ &= E[(Z^*(e))^2] - 2E[Z^*(e) \times Z(e)] + E[(Z(e))^2] \\ &= \sum_{i=1}^n \sum_{j=1}^n \lambda_i \lambda_j E[Z(e_i) \times Z(e_j)] - 2 \sum_{i=1}^n \lambda_i E[Z(e_i) \times Z(e)] + E[(Z(e))^2] \quad (1) \\ &= \sum_{i=1}^n \sum_{j=1}^n \lambda_i \lambda_j (C(e_i, e_j) + m^2) - 2 \sum_{i=1}^n \lambda_i (C(e_i, e) + m^2) + C(0)\end{aligned}$$

Since the sum of the weights should be one, a Lagrange parameter μ is added to get $L = \sigma_E^2 + 2\mu\{1 - \sum_{i=1}^n \lambda_i\}$ along with the constraint $\sum_{i=1}^n \lambda_i = 1$. The optimal value for weight λ_i is then:

$$\begin{aligned}\frac{\partial(L)}{\partial(\lambda_i)} &= 2 \sum_{j=1}^n \lambda_j (C(e_i, e_j) + m^2) - 2(C(e_i, e) + m^2) - 2\mu \\ &= -2 \sum_{j=1}^n \lambda_j \gamma(e_i, e_j) + 2\gamma(e_i, e) - 2\mu = 0\end{aligned} \quad (2)$$

Optimal Weights (cont)



For our example, , we get the following system:

$$\begin{pmatrix} \gamma(e_1, e_1) & \gamma(e_1, e_2) & \gamma(e_1, e_3) & 1 \\ \gamma(e_2, e_1) & \gamma(e_2, e_2) & \gamma(e_2, e_3) & 1 \\ \gamma(e_3, e_1) & \gamma(e_3, e_2) & \gamma(e_3, e_3) & 1 \\ 1 & 1 & 1 & 0 \end{pmatrix} \begin{pmatrix} \lambda_1 \\ \lambda_2 \\ \lambda_3 \\ \mu \end{pmatrix} = \begin{pmatrix} \gamma(e_1, e_0) \\ \gamma(e_2, e_0) \\ \gamma(e_3, e_0) \\ 1 \end{pmatrix}$$

Method Comparison



- Simple Average and Inverse Distance
 - (+) No training required
 - (-) Does not consider spatial structure
 - (+) Little computation needed
- Kriging
 - (-) At least one surface for training
 - (+) Adapts to spatial structure
 - (*) Potentially large linear system

* Number of neighbors is limited to $\frac{n_{desired}}{n_{current}}$ in first round to

get a desired number of neighbors.

Robustness to Failure



- Sensor or communication failures are common
- Sink does not know if a value was not received because it could be interpolated or due to error
- It is assumed the value could be interpolated
- Since there is no static groups, a sensor failure does not have a large affect on nearby sensors



Overview



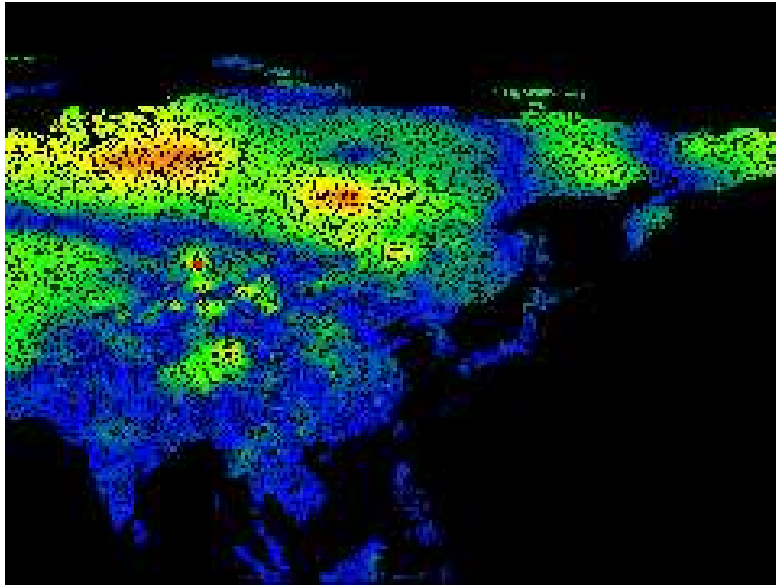
- Motivation
- Related work
- Problem statement
- E2K framework
- Choosing interpolation methods
- **Temporal E2K**
- Experimental results
- Conclusion



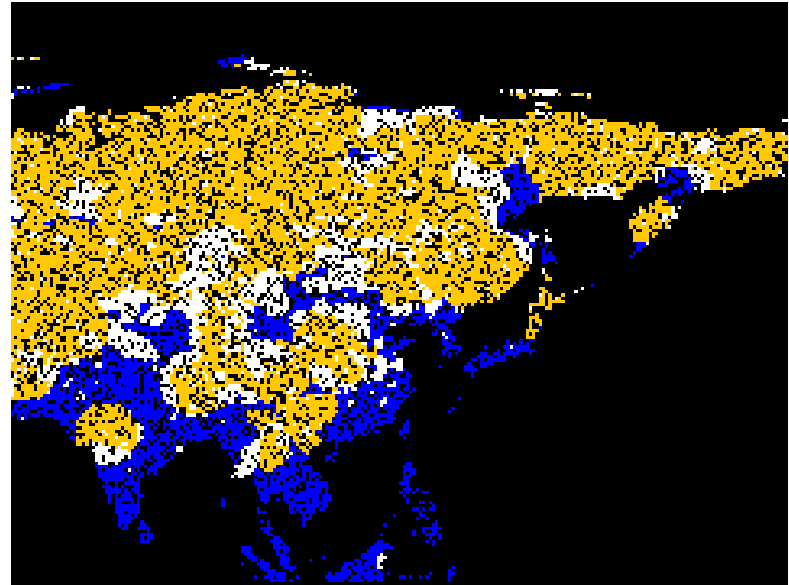
Temporal E2K



- Many times there is both spatial and temporal autocorrelation.



Residual map



Prediction method map

Does not report due to temporal, Does not report due to spatial, Reports (white)



Temporal Sensor (s_0) Algorithm

$Z_d(s_0) \leftarrow$ a temporal predicated value of A for this sensor.

$Z_t(s_0) \leftarrow$ value of A for this sensor at time t .

$p \leftarrow \frac{n_{desired}}{n_{current}}$

$rand \leftarrow$ a random number $\in [0, 1]$

if ($|Z_t(s_0) - Z_d(s_0)| \geq \epsilon$ **and** $rand < p$) **then**

{Round 1}

Report ($Z_t(s_0), round_1$) to central site and neighbors within distance r .

$Z_d(s_0) \leftarrow Z_t(s_0)$

else

{Round 2}

$R \leftarrow$ the set of readings from sensors within distance r that reported in first round.

if ($size(R) > n_{min}$) **then**

$Z_t^*(s_0) \leftarrow interp(R)$

if ($|Z_t^*(s_0) - Z_t(s_0)| \geq \epsilon$) **then**

Report ($Z_t(s_0), round_2$) to central site. $Z_d(s_0) \leftarrow Z_t(s_0)$.

end if

else

if ($|Z_t(s_0) - Z_d(s_0)| \geq \epsilon$) **then**

Report ($Z_t(s_0), round_2$) to central site. $Z_d(s_0) \leftarrow Z_t(s_0)$.

Temporal Central Site Algorithm

We interpolate the readings of the sensors which do not report using the values of sensors that reported. This is achieved by the following scheme:

- If a non-reporting sensor has less than n_{min} neighbors that report in the first round, then use the temporal prediction as the estimation of the sensor reading.
- Otherwise, use the neighbors that report in the first round to interpolate the estimated value for the sensor.

The n_{min} parameter is an important quality control to balance temporal and spatial prediction mechanisms.

Overview



- Motivation
- Related work
- Problem statement
- E2K framework
- Choosing interpolation methods
- Temporal E2K
- **Experimental results**
- Conclusion



Experiment Design



- Experiments were done using simulations and a real sensor network.



Simulation Experiment Design



- All models were implemented using Java on a standard PC.
- Datasets:
 - *Lab*: an Intel lab dataset [Madden] consisting of 54 sensors
 - *Asia Temperature*: from the University of Delaware global surface air temperature monthly grids [asi].
- For the lab data the first 94 hours was used for training and the next 452 hours was used for testing.
- For the Asian temperature data the first 10 years was used for training and the next 40 years was used for testing.



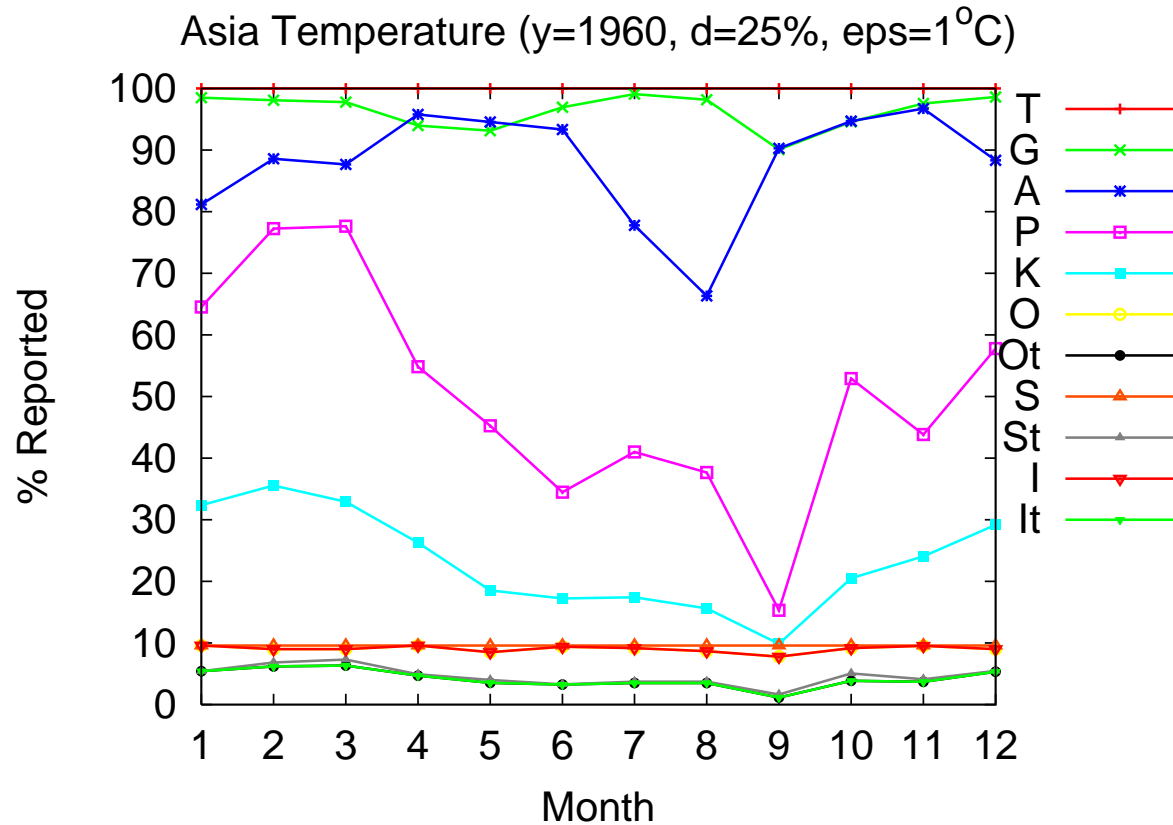
Schemes in Our Experiments



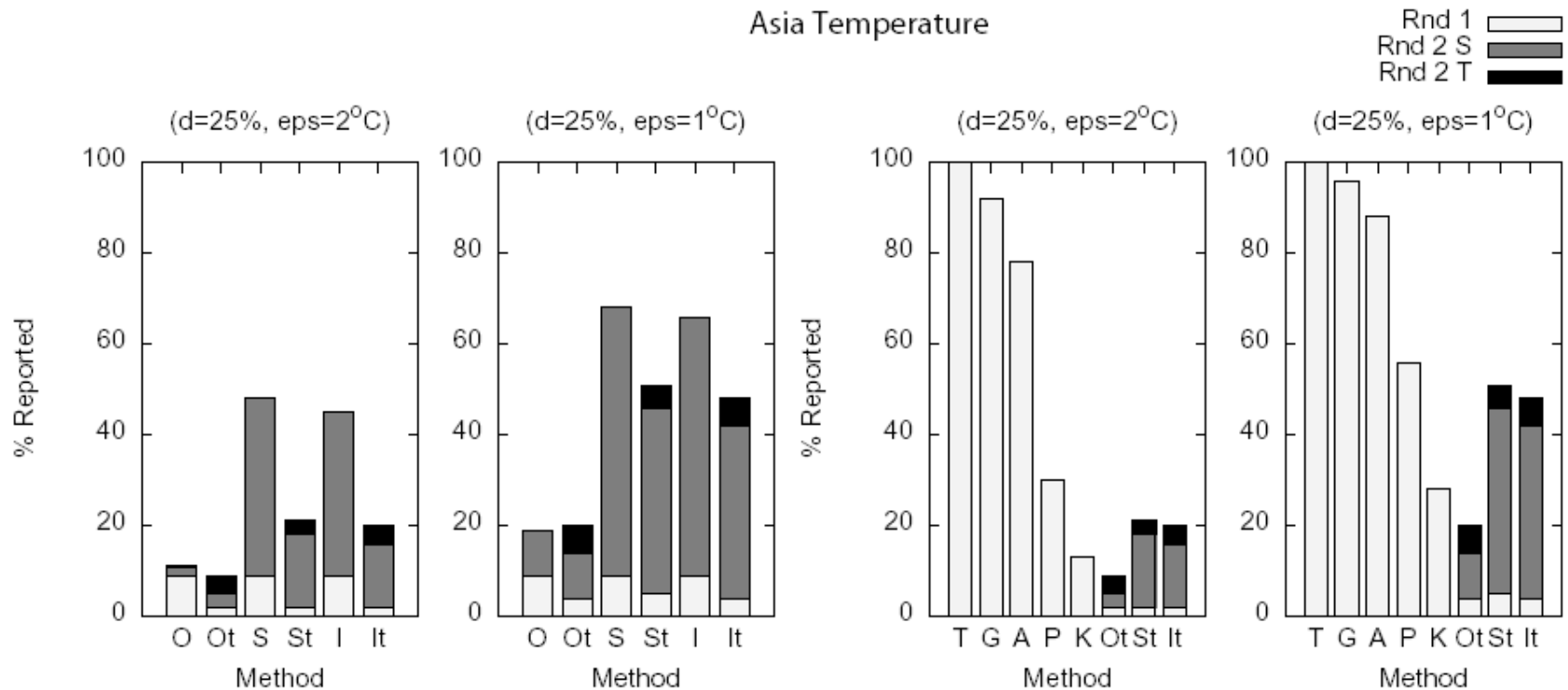
- TinyDB (T)
- Global Average (G)
- Approximate Caching (A)
- Periodical Approximate Caching (P)
- Ken Model (K)
- Ordinary Kriging (O)
- Simple Average (S)
- Inverse Distance (I)



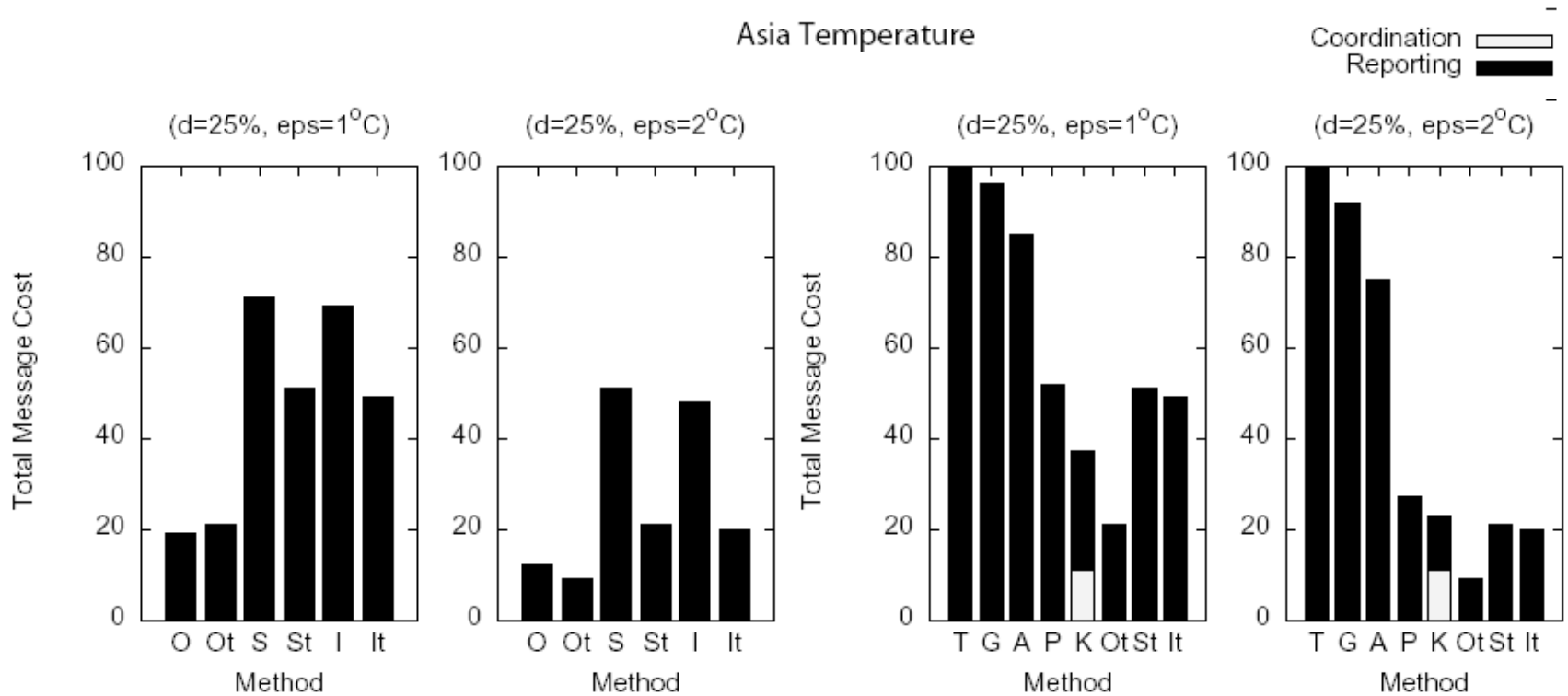
Asia: Monthly Variations



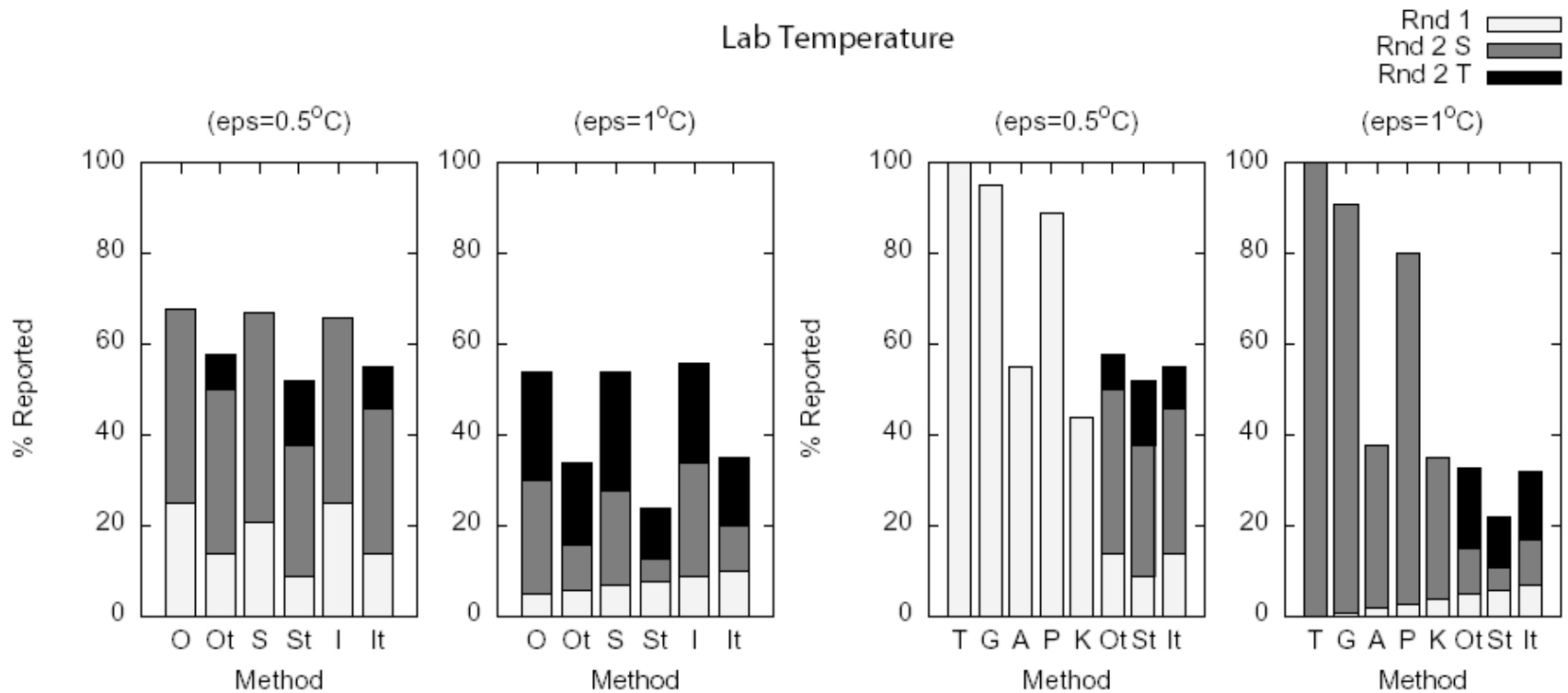
Asia: Message Savings w/out Routing



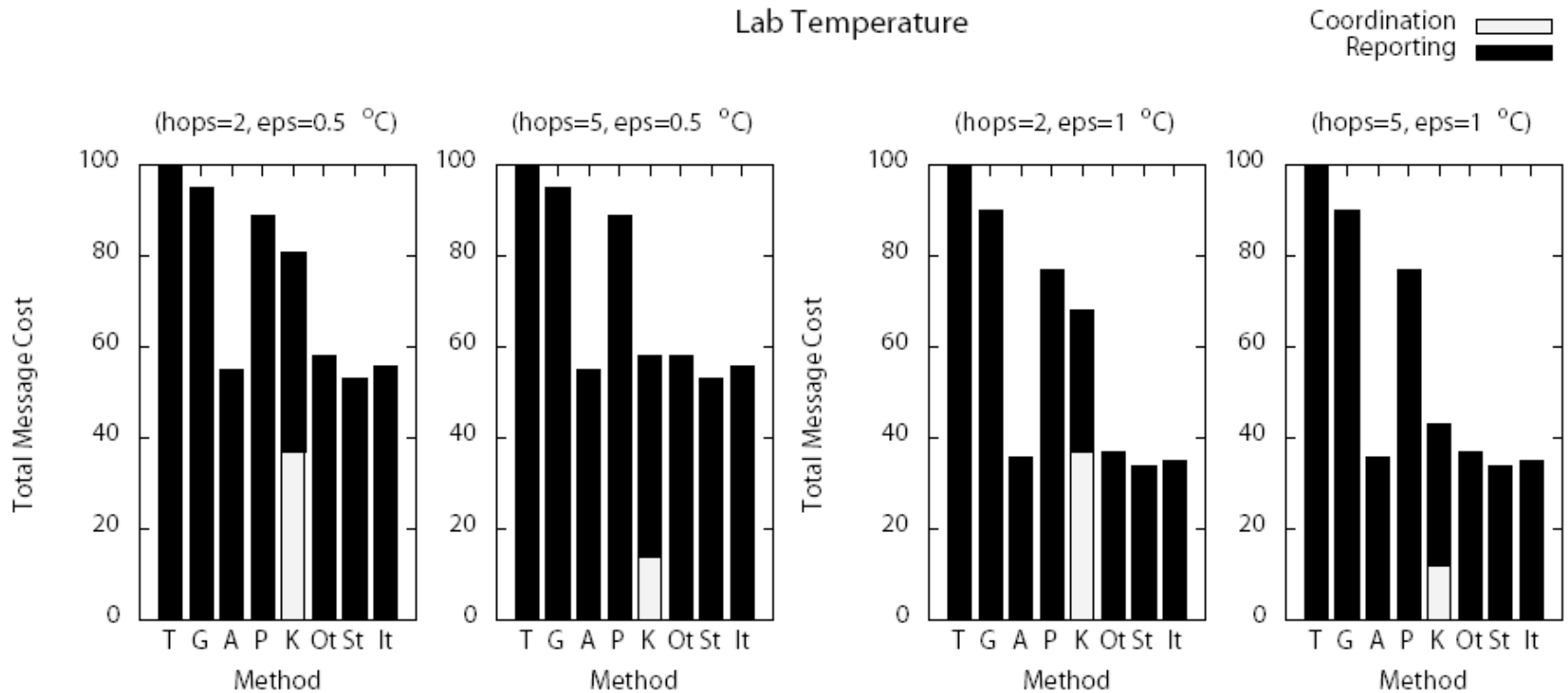
Asia: Message Savings with Routing



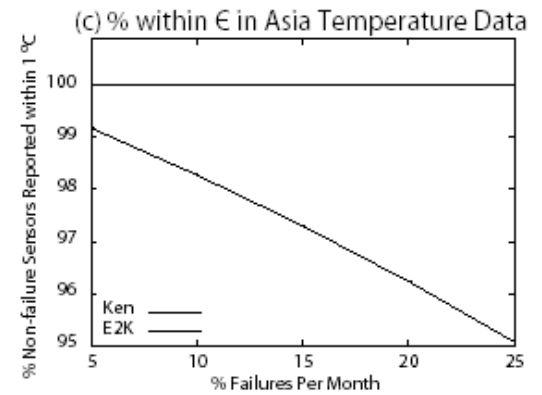
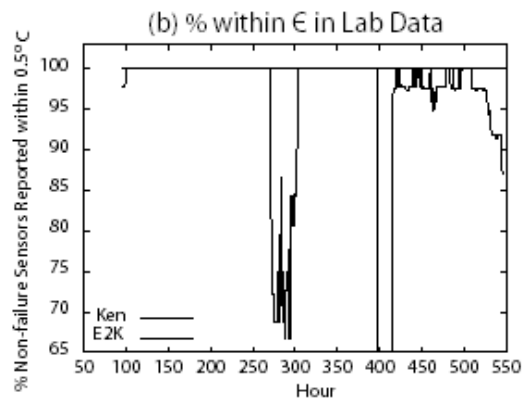
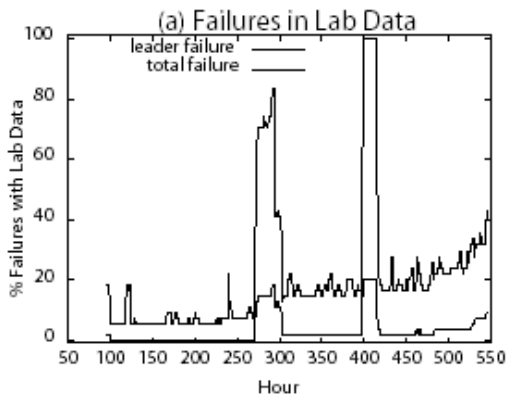
Lab: Message Savings w/out Routing



Lab: Message Savings with Routing



Failures

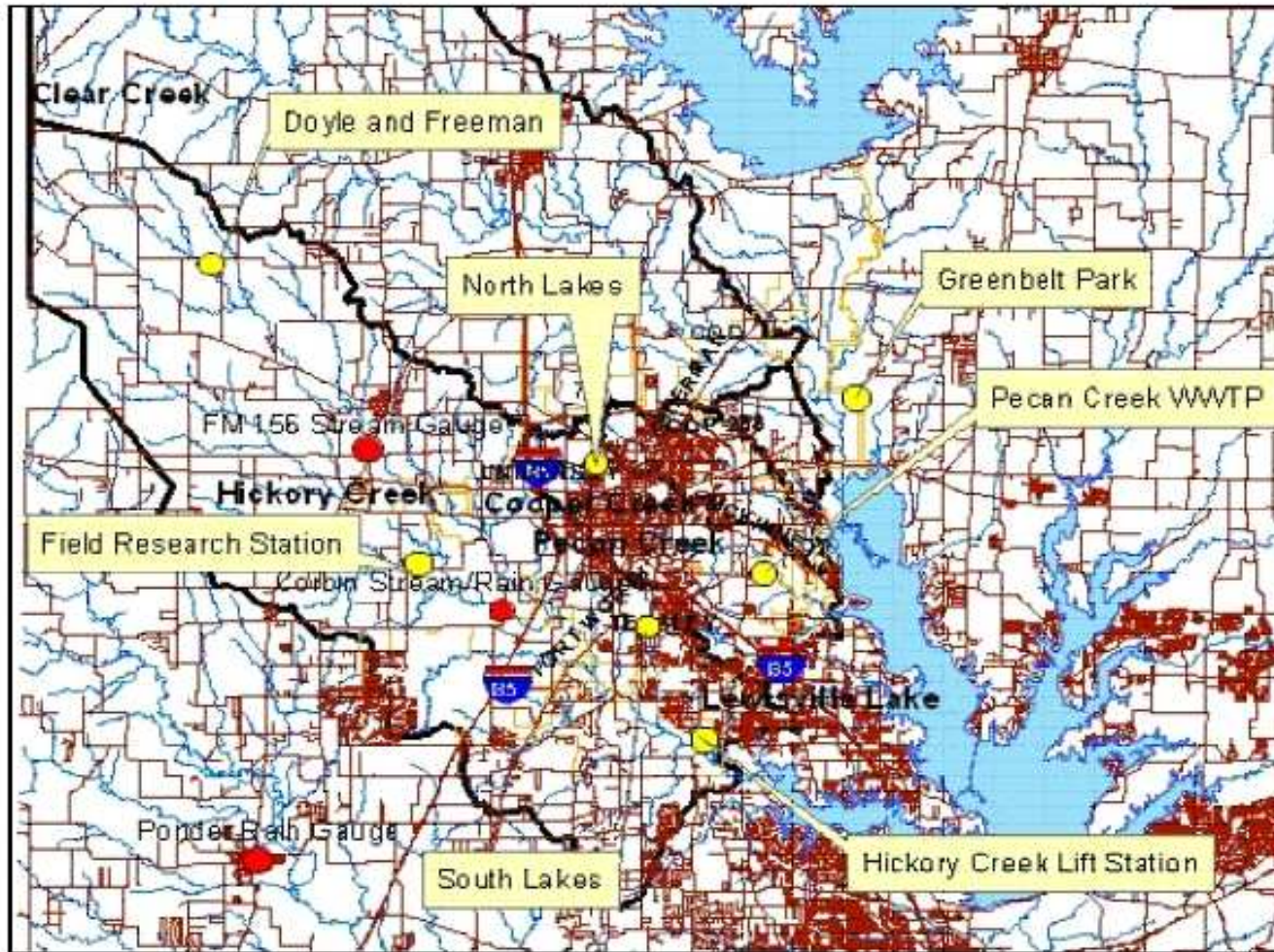


Real Network Experiment Design

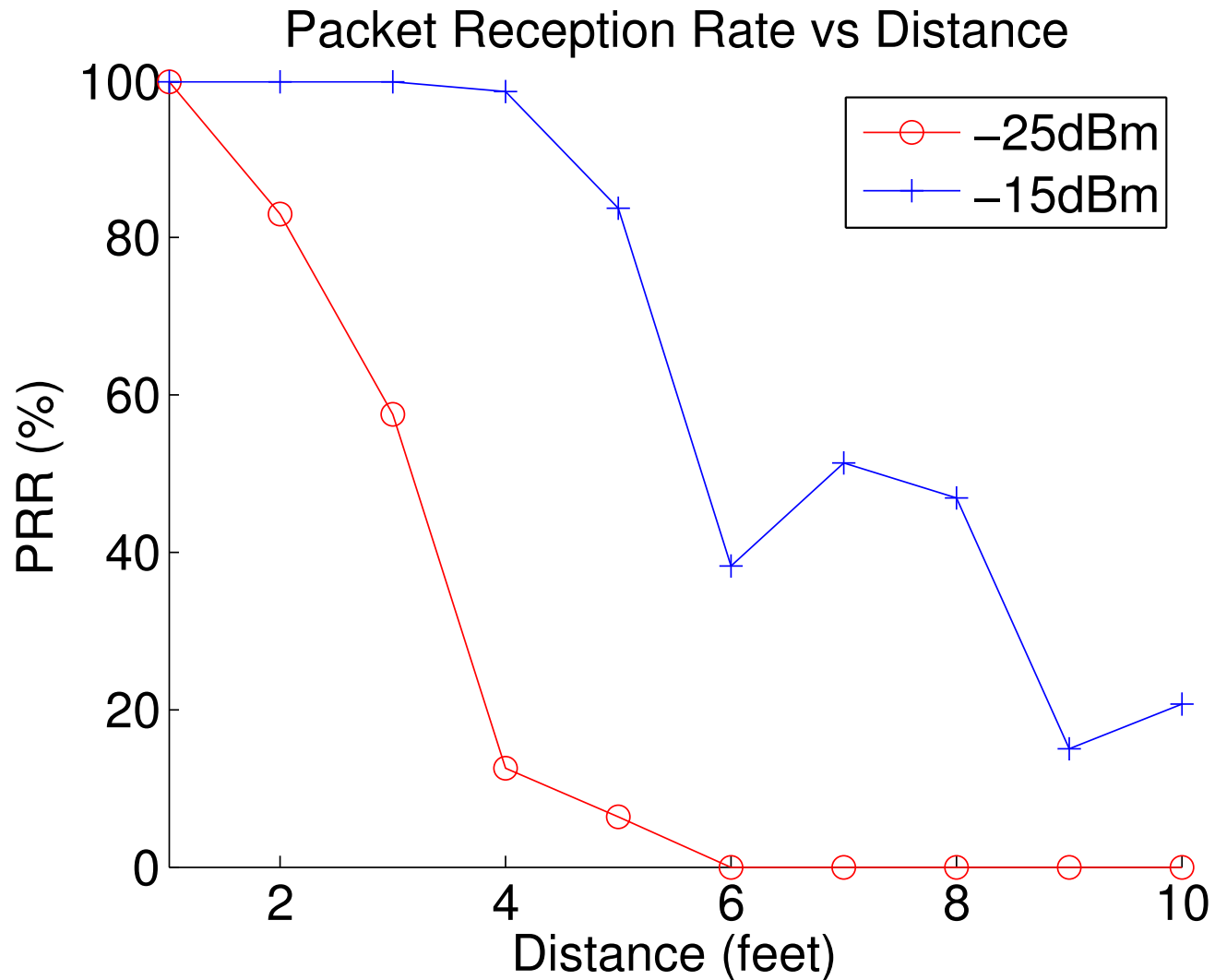
- Naive, E2K, Approximate Caching, and Temporal E2K were implemented on MicaZ motes.
- Tests were done with a network of 25 sensors using a projector to vary light.



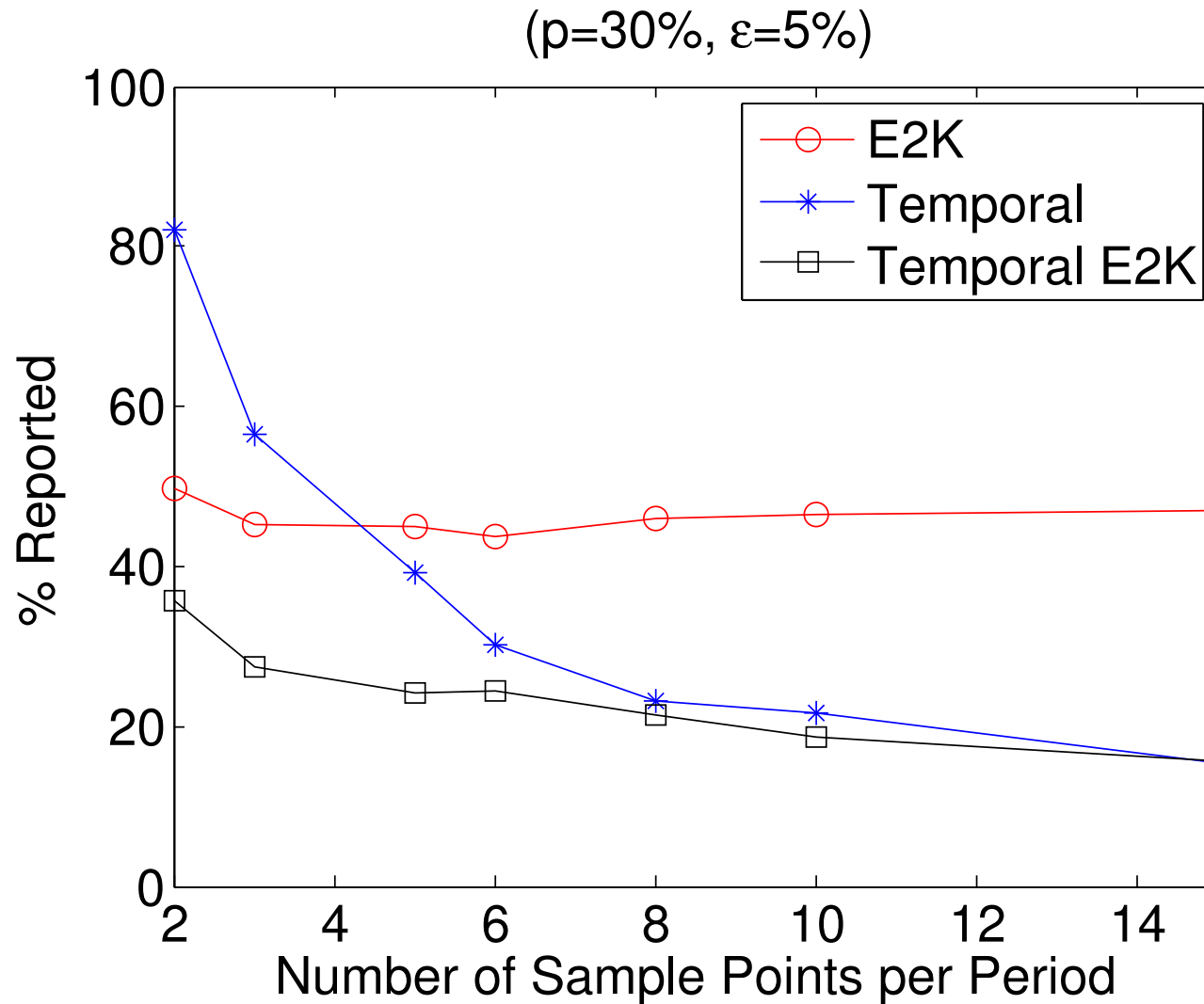
Planned Deployment



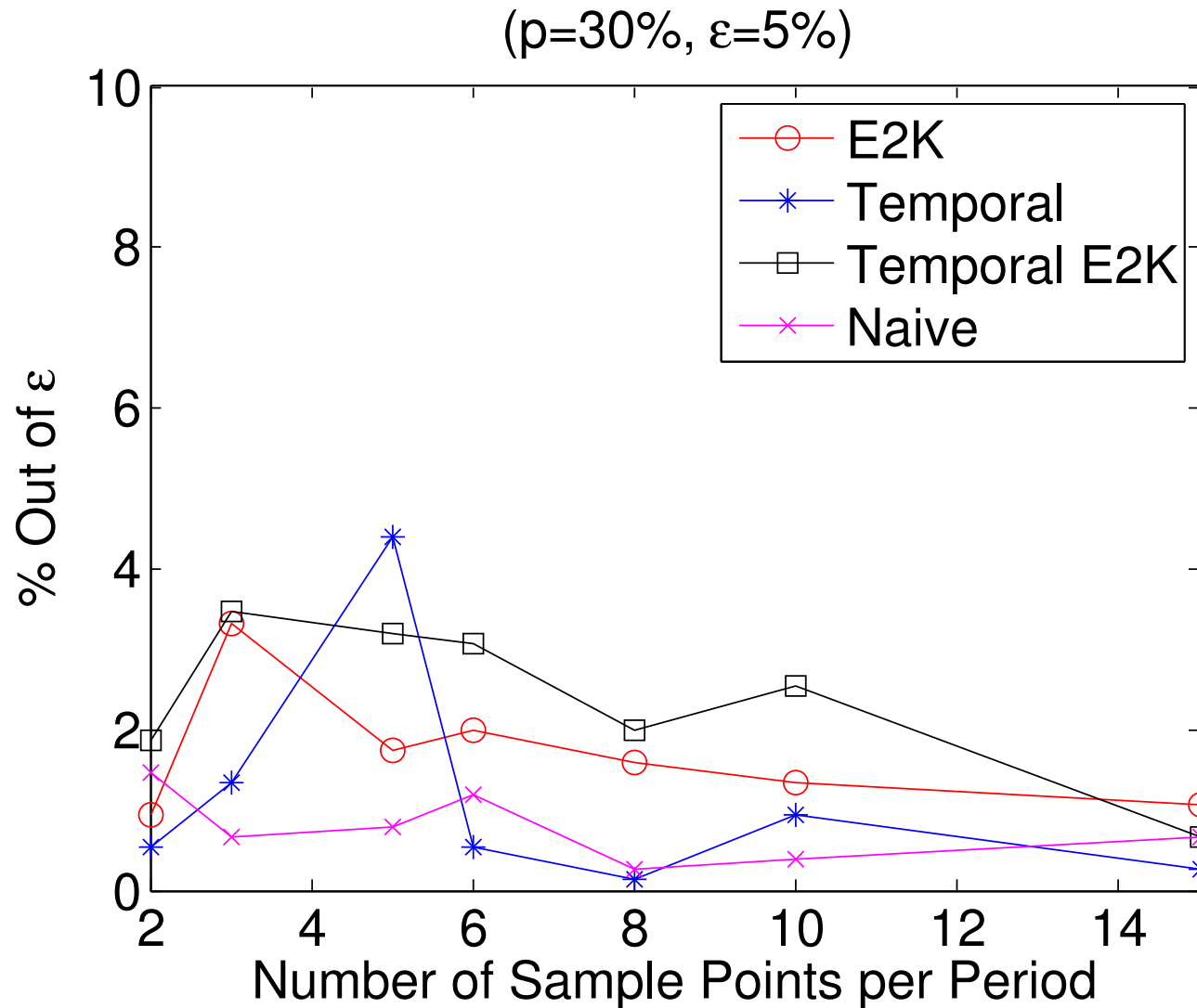
Packet Reception Rate



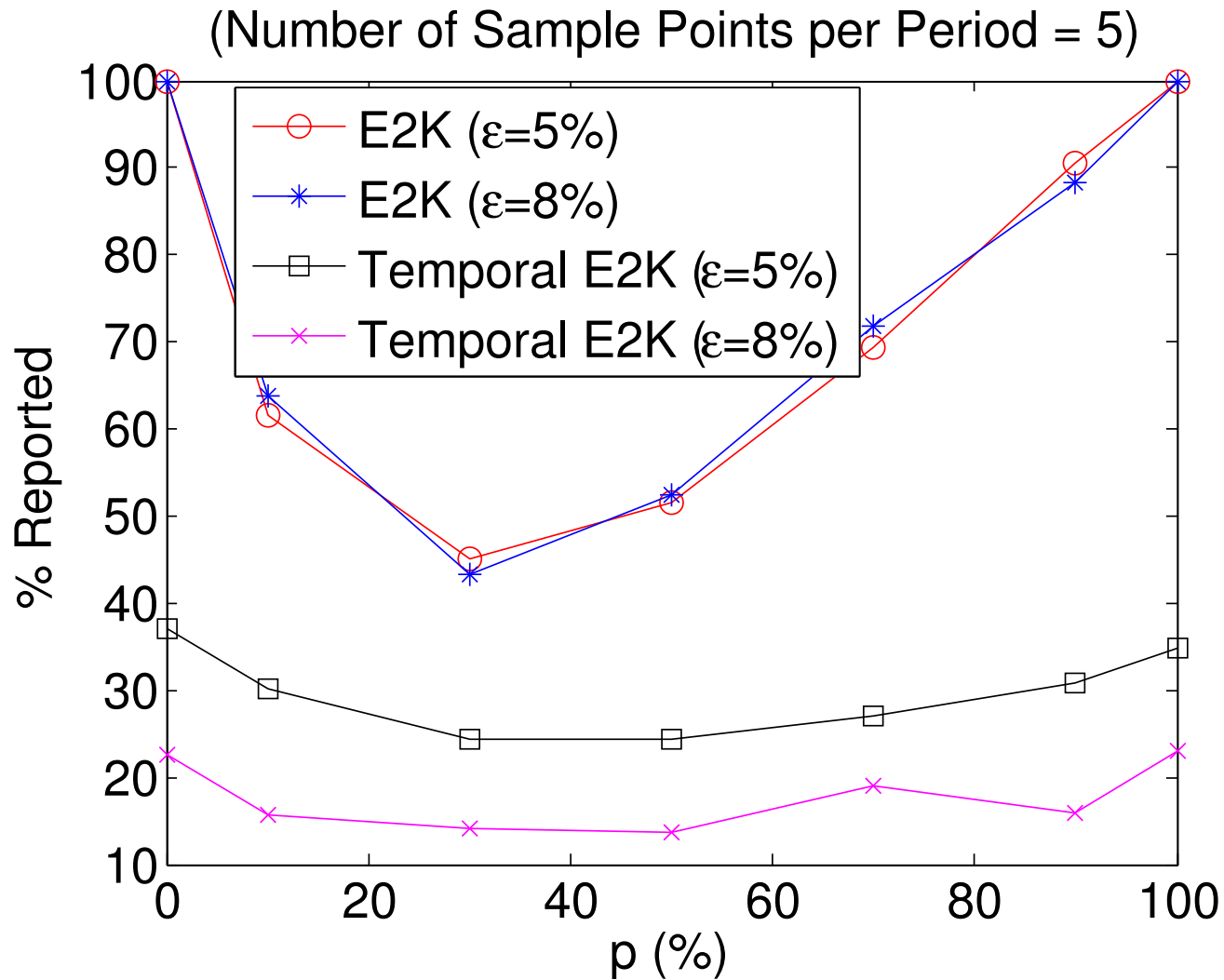
Percent Reporting vs Sample Rate



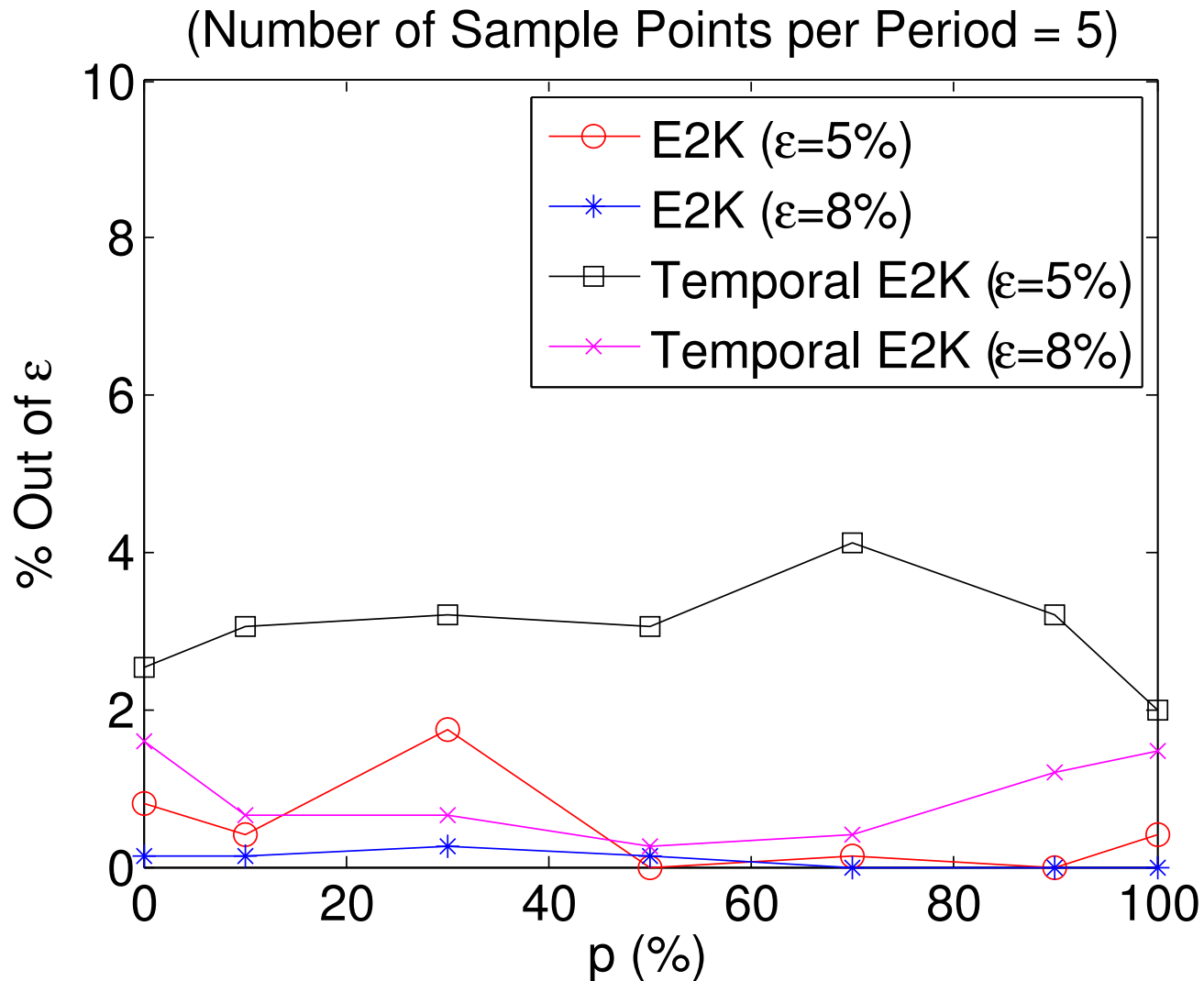
Percent Out of Bound vs Sample Rate



Percent Reporting vs Probability to R



Out of Bound Error vs Probability to



Overview



- Motivation
- Related work
- Problem statement
- E2K framework
- Choosing interpolation methods
- Temporal E2K
- Experimental results
- **Conclusion**



Conclusion



- The proposed E2K framework works for any spatial interpolation method
- It requires only localized information for spatial interpolation
- It can incorporate temporal methods
- Future extensions:
 - Incorporating cokriging
 - Probabilistic backoff reporting
 - Query optimizer at the sink



References

University of delaware surface air temperature data.

<http://climate.geog.udel.edu/~climate>.

M. H. Ali, W. G. Aref, and C. Nita-Rotaru. Spass: Scalable and energy-efficient data acquisition in sensor databases. In *MobiDE*, 2005.

D. Chu, A. Deshpande, J. Hellerstein, and W. Hong. Approximate data collection in sensor networks using probabilistic models. In *ICDE*, 2006.

J. Considine, F. Li, G. Kollios, and J. Byers. Approximate aggregation techniques for sensor databases. In *Proceedings of the 20th International Conference on Data Engineering*, 2004.

N. Cressie. *Statistics for Spatial Data*. Wiley and Sons, ISBN:0471843369, 1991.

A. Deligiannakis, Y. Kotidis, and N. Roussopoulos. Compressing historical information in sensor networks. In *ACM SIGMOD*, pages 527–538, 2004.

A. Deshpande, C. Guestrin, S. R. Madden, J. M. Hellerstein, and W. Hong. Model-driven data acquisition in sensor networks. In *Proceedings of VLDB*, pages 588–599, 2004.

S. Goel, A. Passarella, and T. Imielinski. Using buddies to live longer in a boring world, 2004. Rutgers Department of Computer Science Technical Report DCS-TR-558.

Y. Kotidis. Snapshot queries: Towards data-centric sensor networks. In *ICDE*, pages 131–142, 2005.

B. Krishnamachari, D. Estrin, and S. B. Wicker. The impact of data aggregation in