approximate data collection in sensor networks

the appeal of probabilistic models

ICDE 2006
Atlanta, GA
3-7 April 2006

David Chu
Amol Deshpande
Joe Hellerstein
Wei Hong

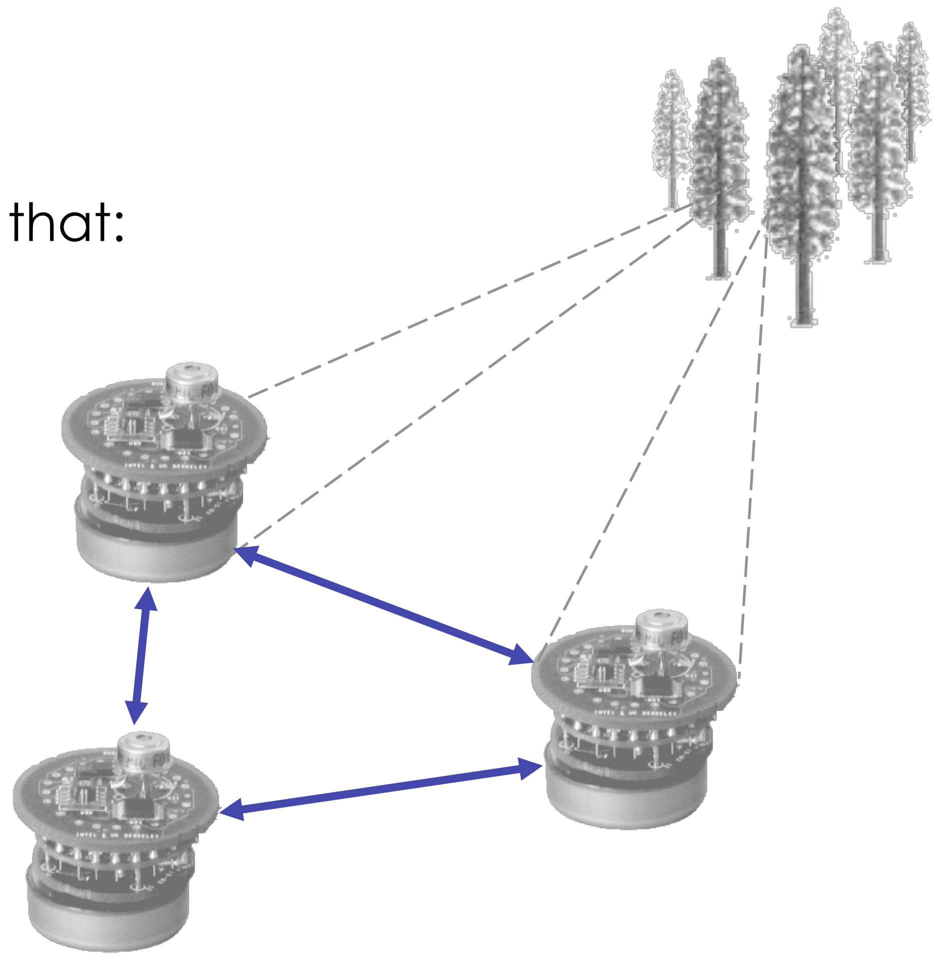
context

Sensor network

Collection of miniature devices that:

- can sense
- can actuate
- can communicate
  - over wireless radios

e.g. *berkeley motes*



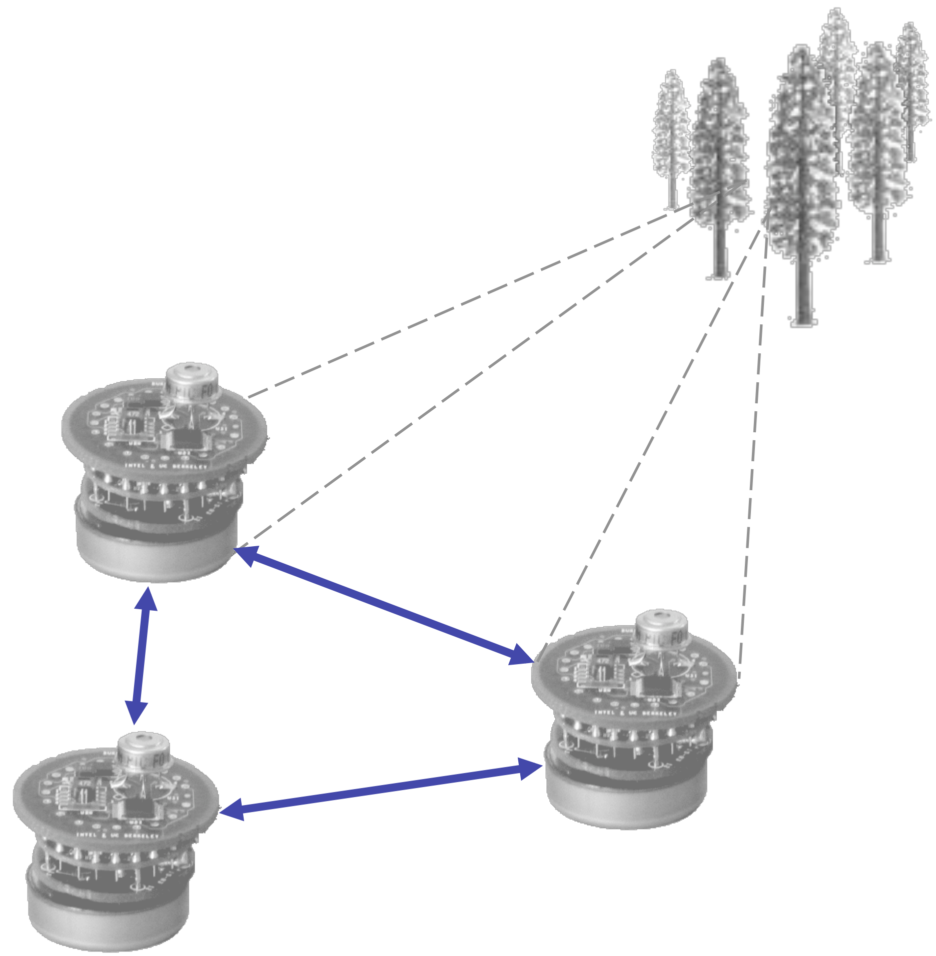
context

Sensor network

Battery lifetime very limited

Communication expensive

Processing relatively cheap



context

Many real deployments

10's – 100's – 1000's – 10,000's

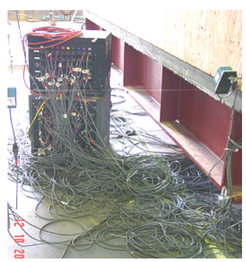
context

Many real deployments

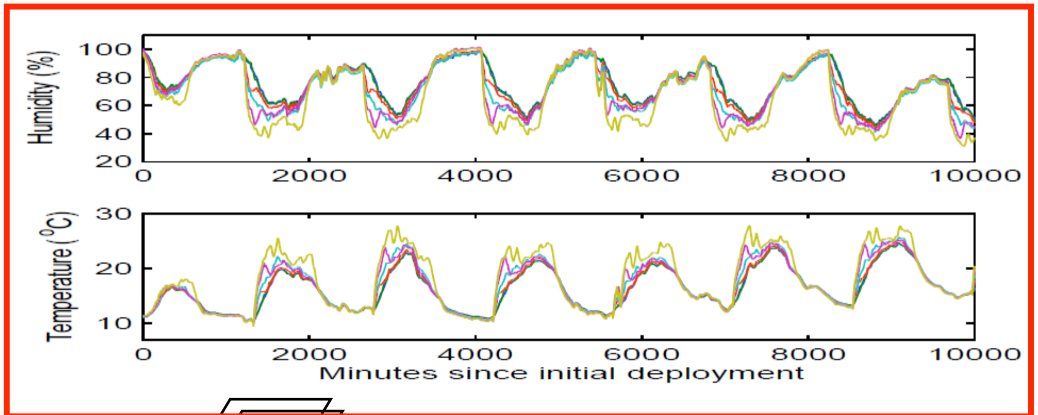
10’s – 100’s – 1000’s – 10,000’s

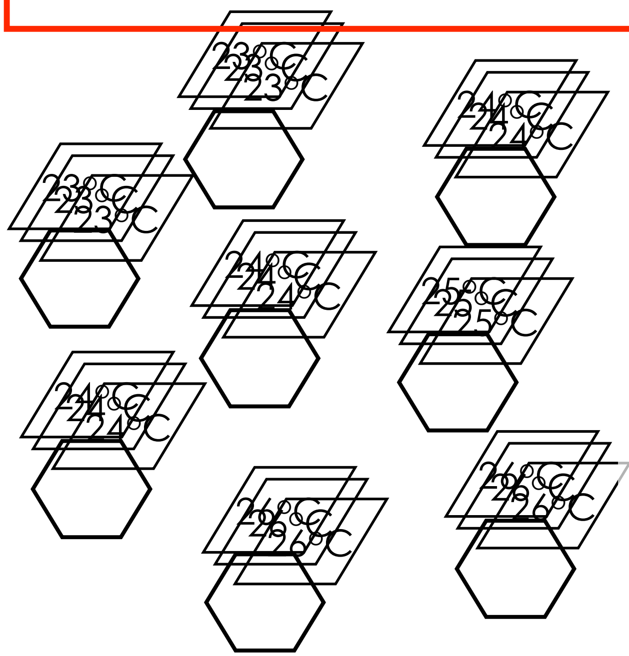
One of the most common uses:

Collect all sensed data

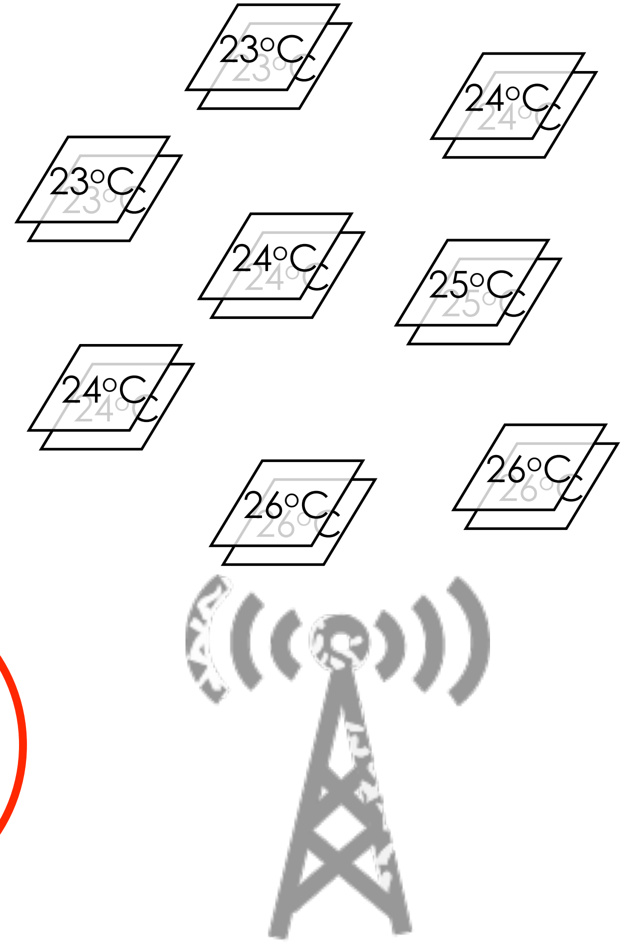


bulk collection





network of nodes



base station

problem

- Communication is costly.

- Users prefer all the data

```
SELECT *
FROM sensors
EPOCH 5 mins
```

- Low res. at high frequency rather than high res. at low frequency

(collection)

- Anomaly detection requires periodic sampling

- Anomaly triggers notification of event

- Why not let user know about all sampled data?

(event detection)

observations

- Physical environments → predictable correlated states
- Bounded error is acceptable
  - Sensed data is noisy
- Processor inexpensive and often idle
- *Report data only if it differs significantly from what is expected.*

introducing (prediction) models

time

I know past samples from this node:

$\ldots, X_i^{t-2}, X_i^{t-1}, X_i^{t}$

space

I know samples from other nodes:

$X_k^{t}, X_m^{t}, X_n^{t}, \ldots$

My guess for the new sensor samples is…

X_i^{t+1}

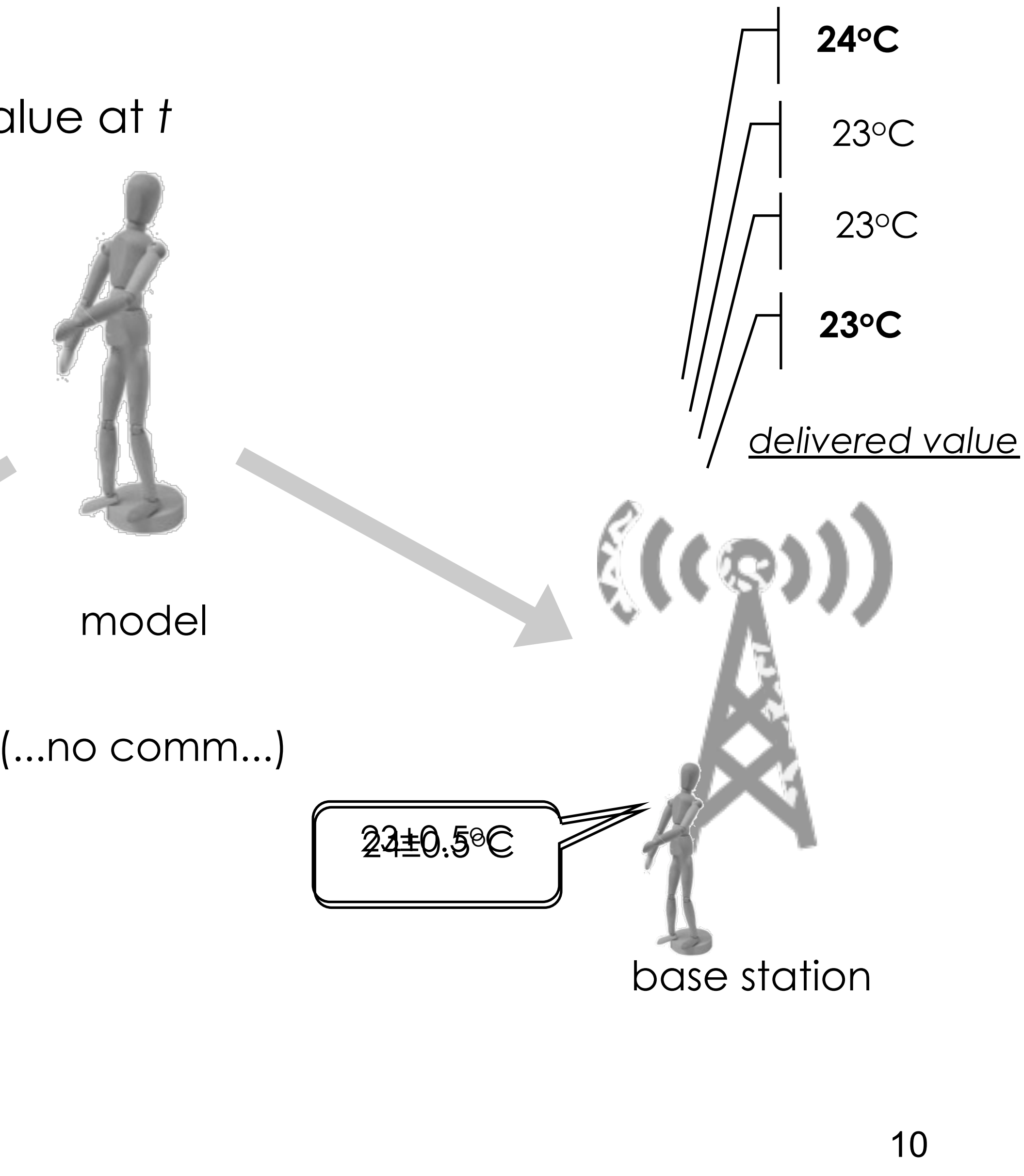
model ([INPUT] ...) → EXPECTED_VAL

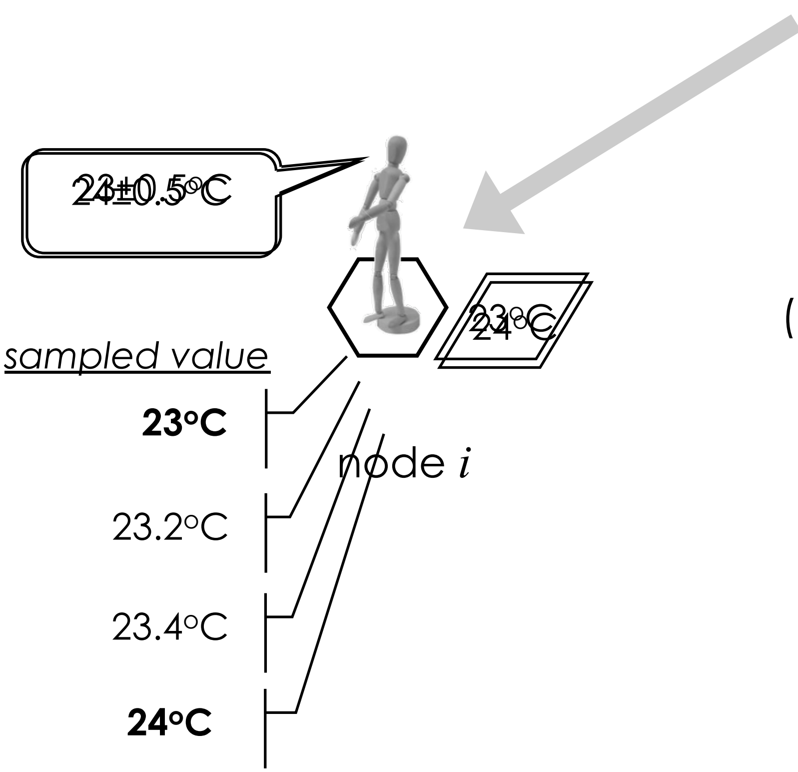
an example

Model: expected value at $t+1$ = value at t

Another possibility: A linear model

$$X^{t+1} = a\,X^{t} + b$$



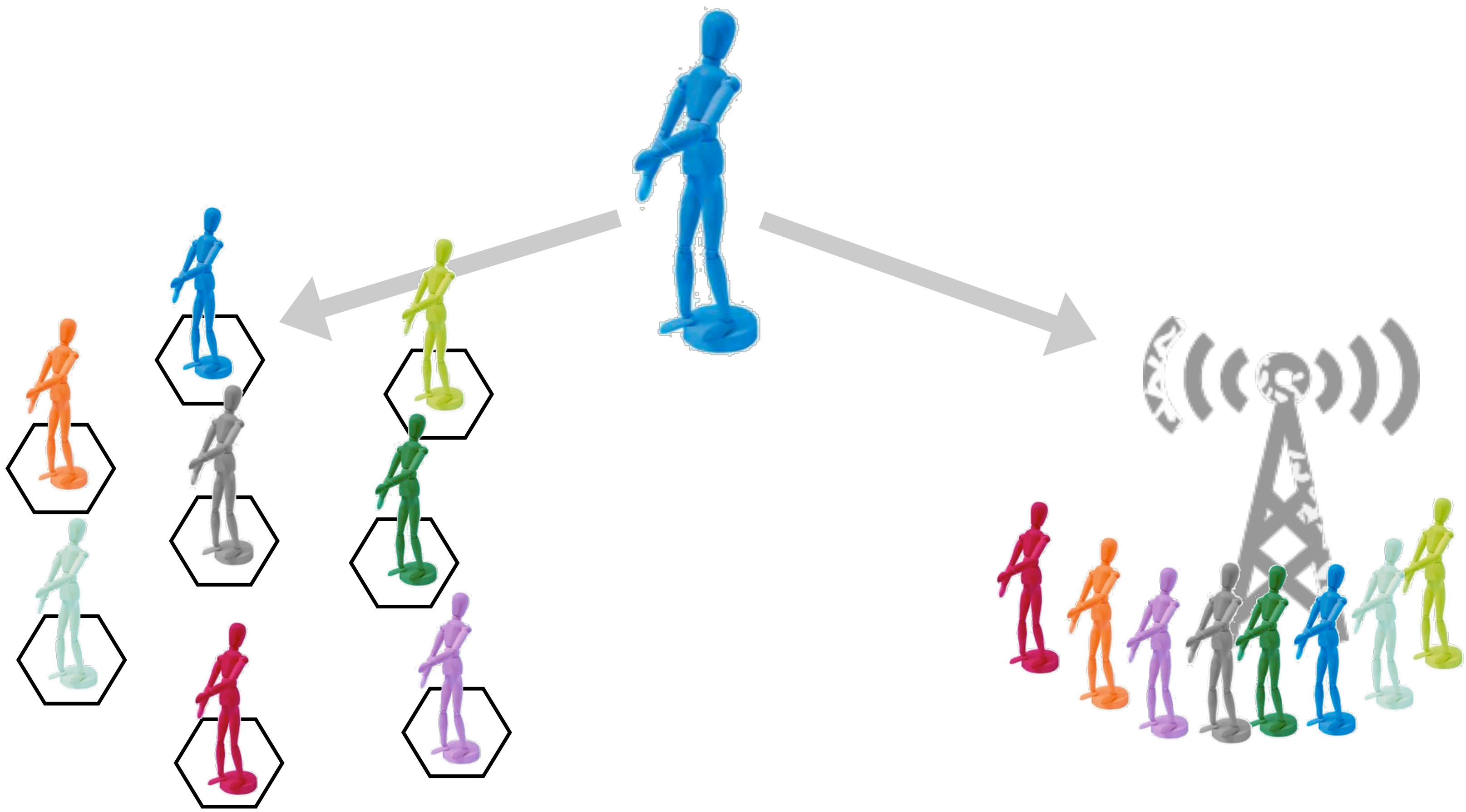


ken

ken

1. Barbie’s boyfriend
2. bounded-loss in-network data reduction
3. the range of perception, understanding, or knowledge

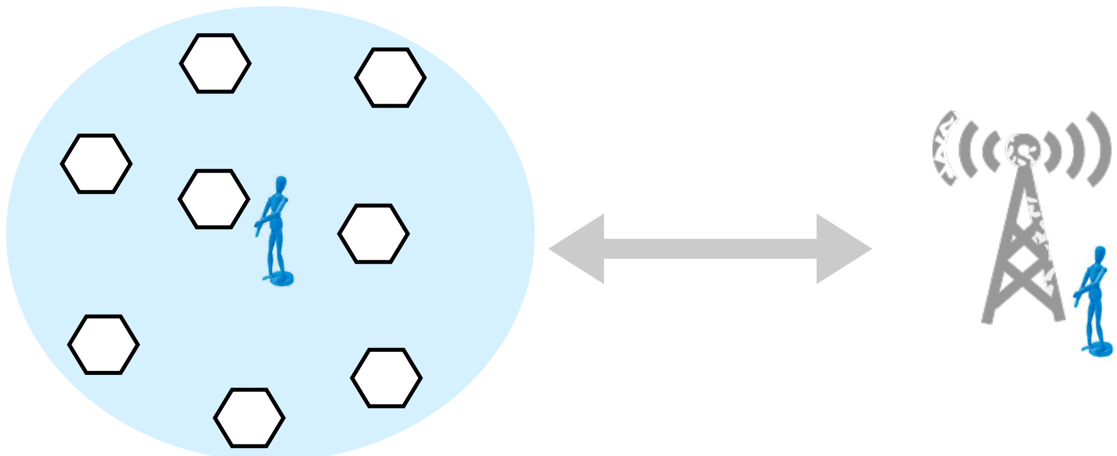
example



properties

- Nodes report to base
  all anomalous samples
- Base delivers to user
  samples within user-tolerated error bound
- Online bounded-loss data reduction using <u>time</u> correlations. What about <u>space</u>?

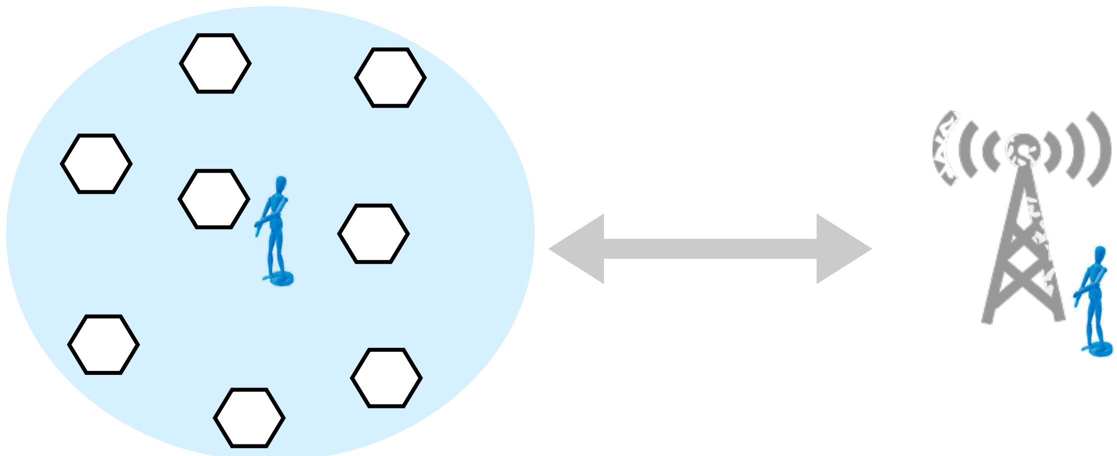
space



Use multi-dimensional prediction models

$$(X_1^t, X_2^t, X_3^t, \ldots) \rightarrow X_1^{t+1}, X_2^{t+1}, X_3^{t+1}, \ldots$$

space

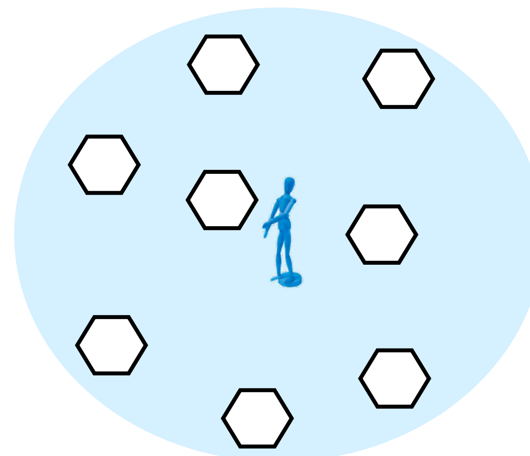


But, must collect all sensor readings at a single sensor node to do prediction

Almost as expensive as bulk data collection

Also infeasible given the computation limitations at each node

two extremes



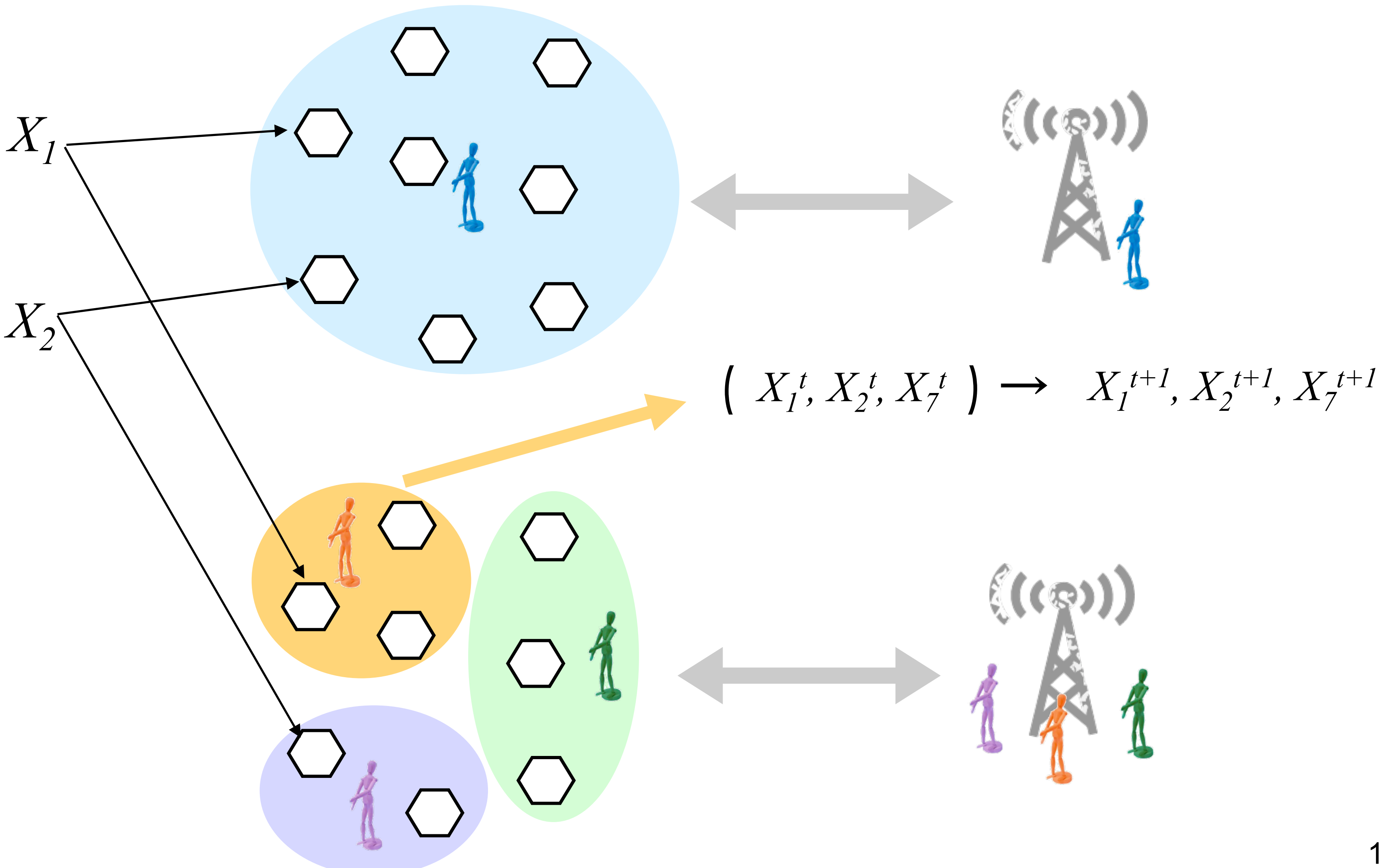
1 node — entire network

no spatial correlations — full spatial correlations

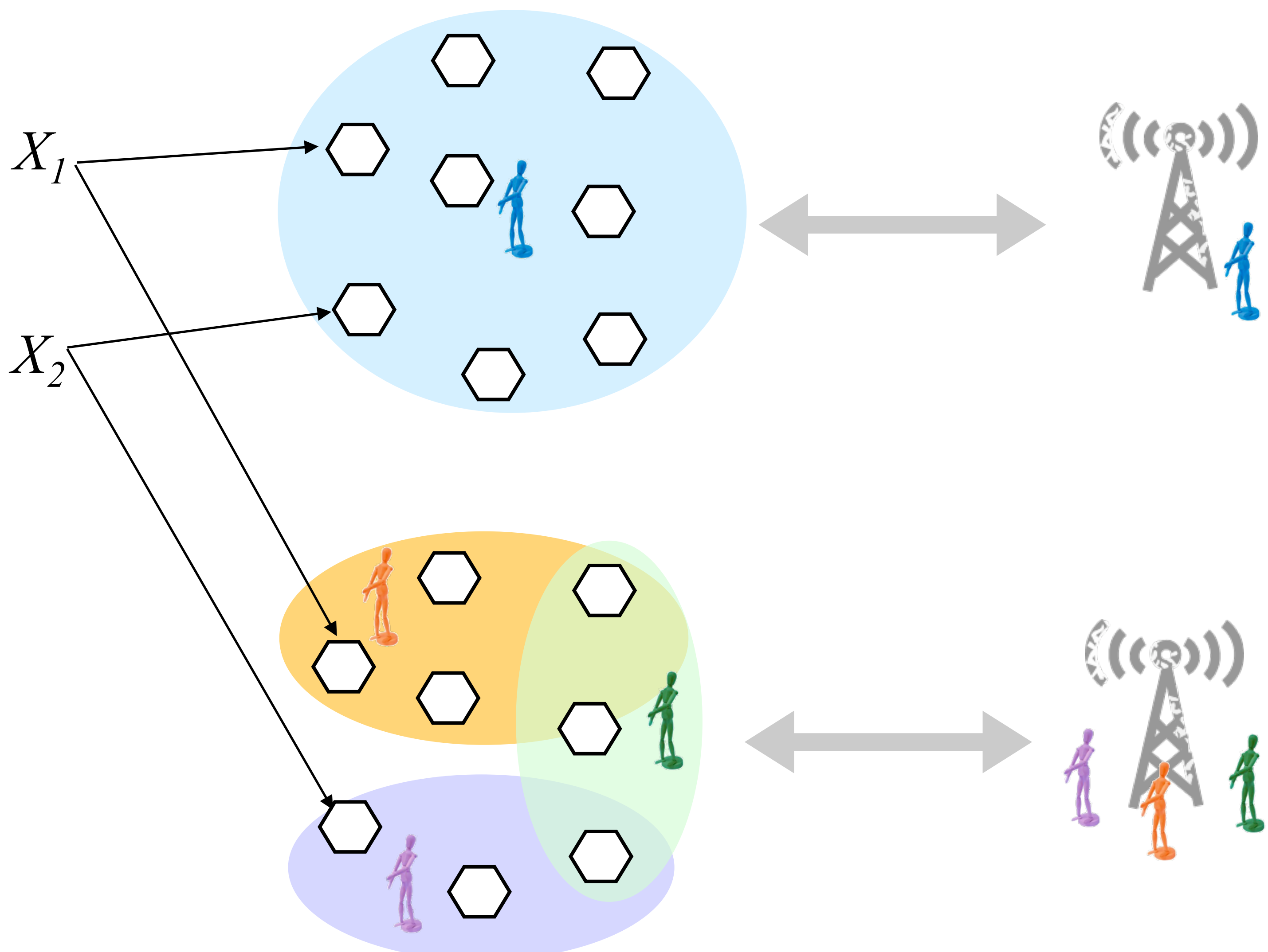
low overhead — high overhead

Ken explores the spectrum between these two

example 1

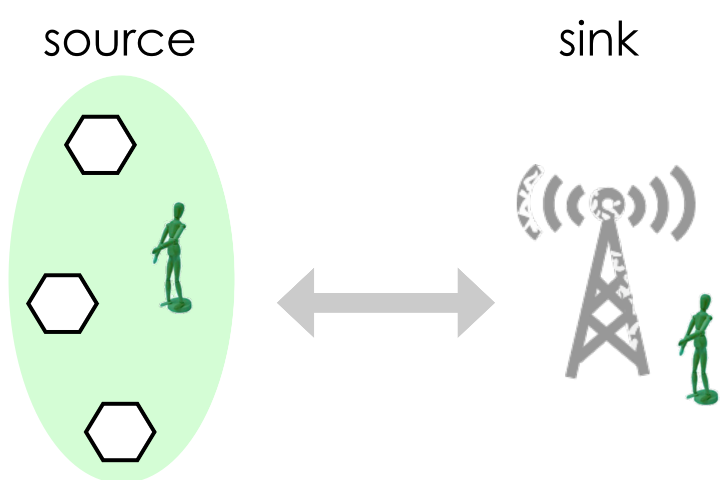


example 2



need to specify

- prediction model
  - someModel ([INPUT] ...) → EXPECTED_VAL
  - How good is the model?
    - How much does it deviate from sampled?
    - What is the cost of correcting the deviation?
  - *Data reduction factor*
- communication structure
  - Where do we collect INPUTs?
  - *Total Communication cost*
    - intra-source
    - source-sink

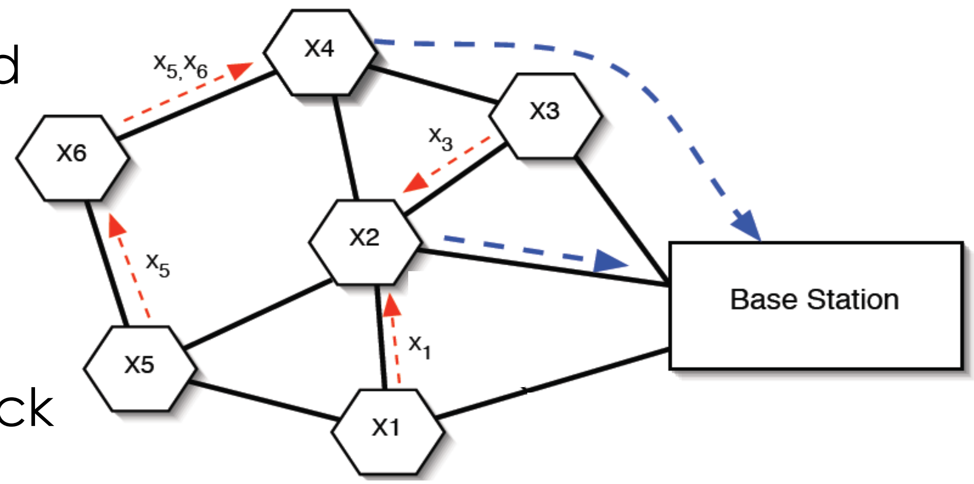


structure: disjoint cliques

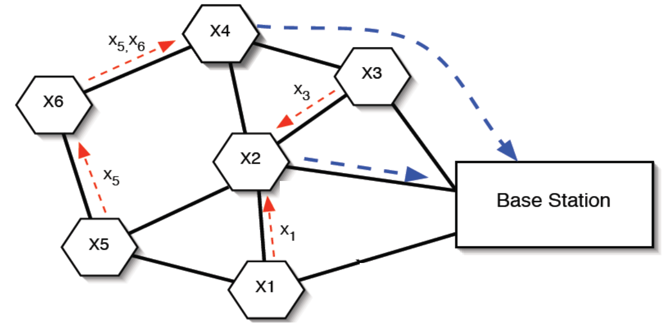
- Allow multiple nodes (a clique) to collect data in-network at a *clique root* and perform inference over multiple sensor readings.

- Clique root decides which readings (if any) to send back to base station.

- Not fully specified: clique members, clique roots ?



disjoint cliques

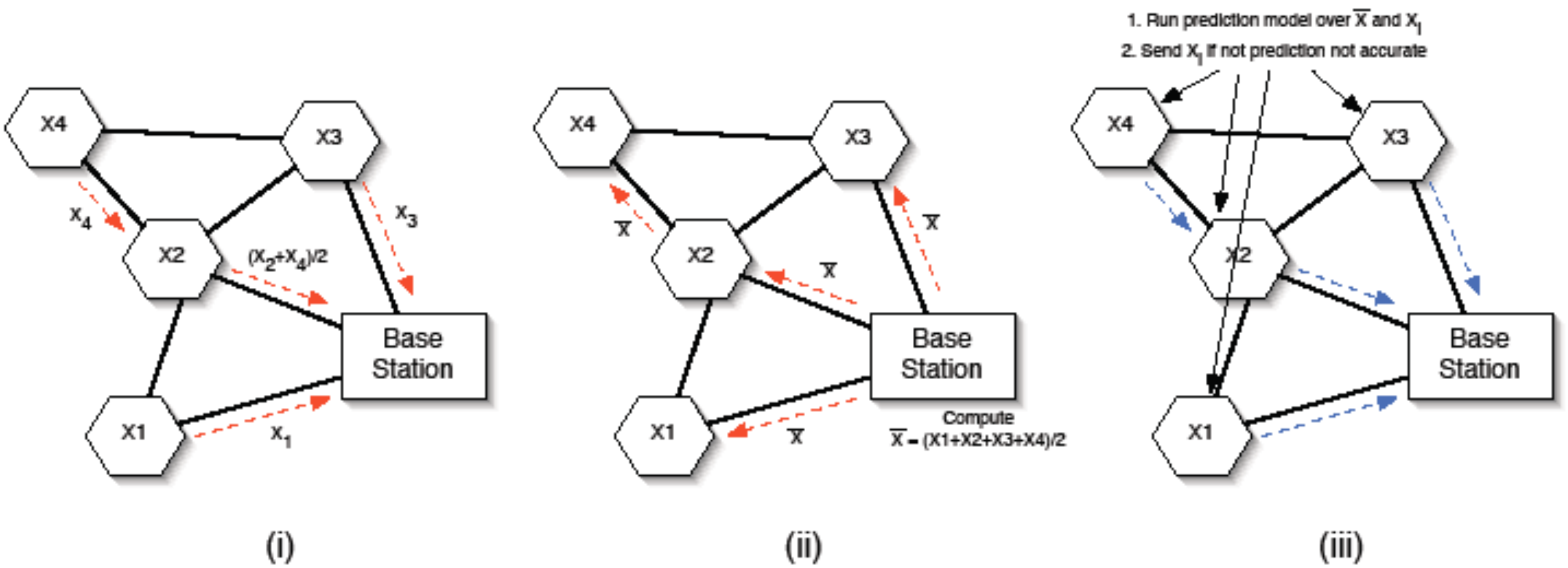


- goal: find the cliques and clique roots with lowest expected communication
- cost factors: data reduction factors, intra-source and source-sink
- exhaustive algorithm
  - find optimal node partitioning (NP-hard)
- greedy heuristic
  - prune unlikely candidates

structure: composite-value

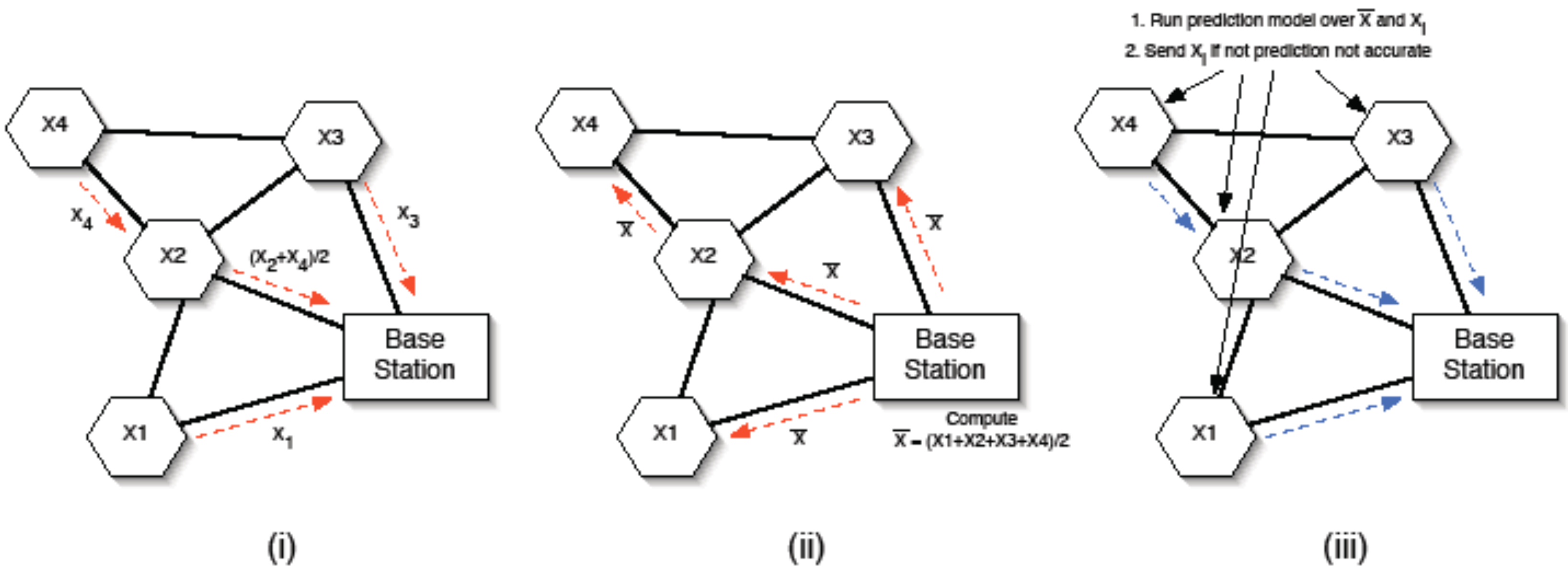
- Compute a composite value (e.g. average) in-network, then disseminate computed composite.

- Run *n* models over two variables each: $X_i, \bar{X} = \frac{\Sigma_{i=1}^{n} X_i}{n}$



structure: composite-value

- Average is likely to be highly correlated with individual readings
- Communication cost of average computation is only *O(n)* messages



evaluation

- input
  - Intel Lab dataset
  - UC Botanical Gardens dataset
- compare
  - Ken w/ average-value
  - Ken w/ disjoint cliques
  - bulk collection
  - caching
  - single node models
- error bounds
  - ±0.5°C
  - ±2% humidity
  - ±0.1V battery

results at a glance

data reduction

- <u>60% with 2-node clique</u>
- <u>82% with 5 –nodes clique</u>

communication reduction

- <u>28% with 2-node</u>
- <u>45% with 5-node clique</u>

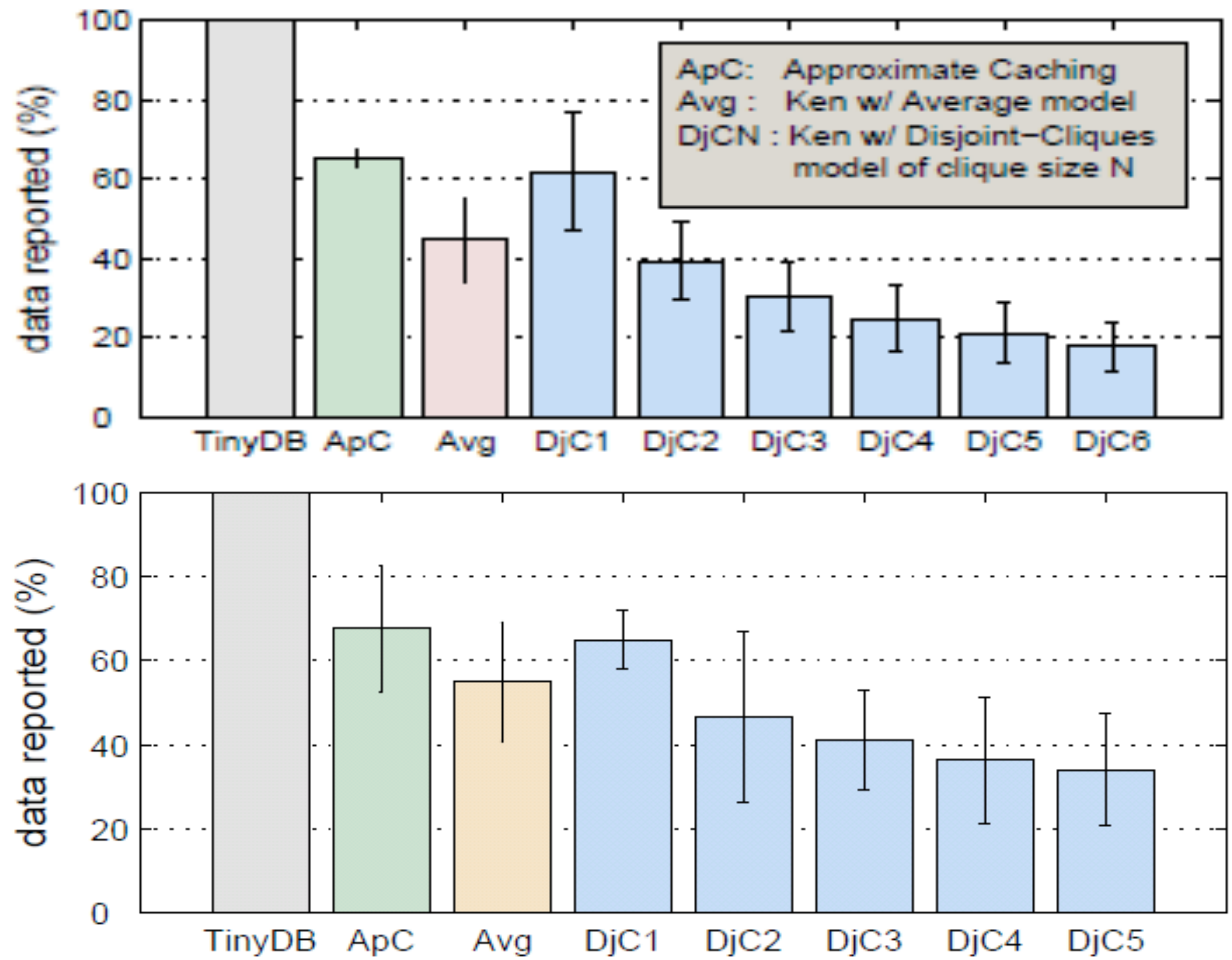
multi-attribute reduction

- <u>65% with 3 attributes</u>

evaluation: data reduction

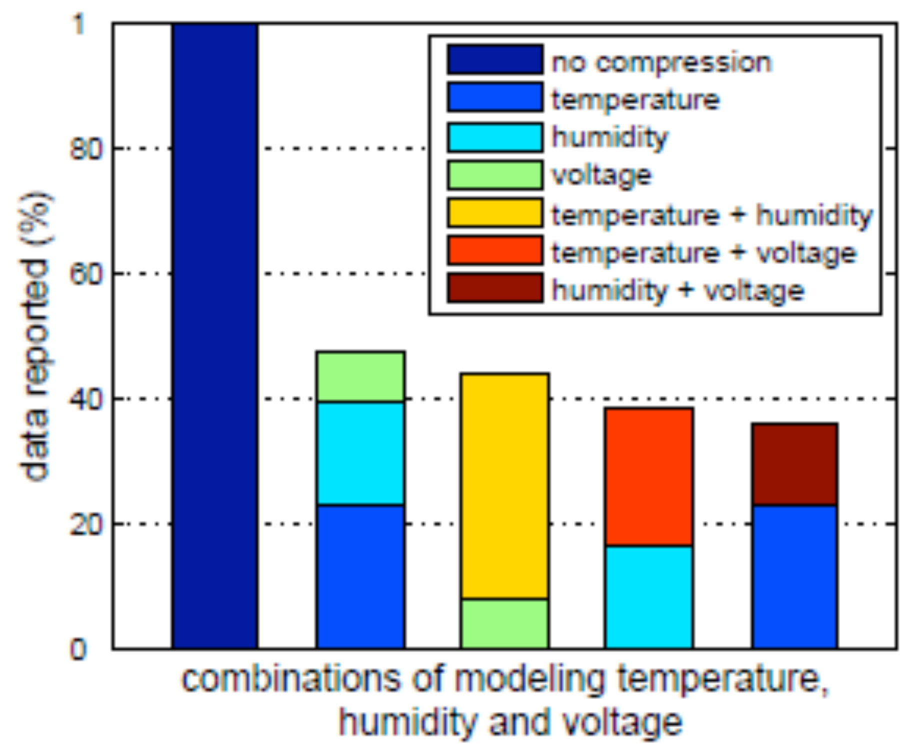
garden

lab



evaluation: multiple attributes

- spatial correlations across attributes
- no additional communication
- mix-and-match with overlay of choice



related work

- Approximate caching [Olston, et al.]: model-less caching
- Stream resource management using Kalman Filters [Jain, et al.]: temporal only
- BBQ [Deshpande, et al.]: pull-based query driven approach; probabilistic guarantees only
- TinyDB, TAG [Madden, et al.]: data service for sensor networks

conclusion

- exploiting both temporal and spatial correlations in real-world datasets
- Find the right communication structure → substantial data reduction achievable
  - 60% with only two node clique and simple model
- communications savings appreciable, even for simple models
  - 28% with only two node clique and simple model
- guarantee of desired accuracy independent of model

thanks!

- questions?