Robust Message-Passing for Statistical Inference in Sensor Networks

Steve Gu and Ying Zheng
Instructor: Jun Yang
Feb 12, 2008

Outline

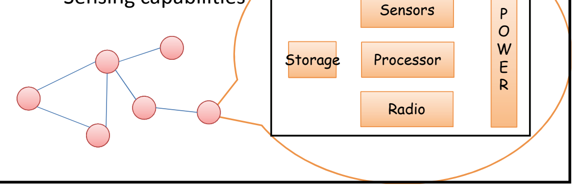
- 1. Introduction
  - Sensor Networks
  - Graphical Modeling
  - Motivation
- 2. Problem Statement
  - Inputs
  - Outputs
  - Assumptions
- 3. BP and RBP
- 4. Architecture
- 5. Implementations
- 6. Evaluation
  - Analysis of communication costs
  - Experiments

Part I: Introduction

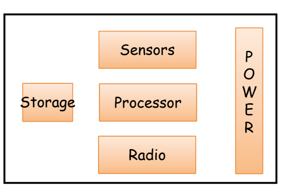
→Sensor Networks
→Graphical Modeling
→Motivation

1.1 Sensor Networks

- Networks of typically small, battery-powered, wireless devices.
  - On-board processing,
  - Communication, and
  - Sensing capabilities

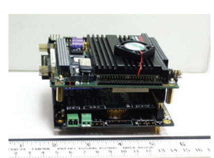


1.1 Sensor Networks

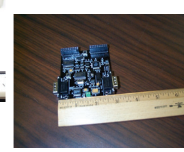


- Low-power processor.
  - Limited processing.
- Memory.
  - Limited storage.
- Radio.
  - Low-power.
  - Low data rate.
  - Limited range.
- Sensors.
  - Scalar sensors:
    - temperature, light, etc.
  - Cameras, microphones.

1.1 Sensor Networks: Examples



PC-104+ (off-the-shelf)



UCLA TAG (Girod)

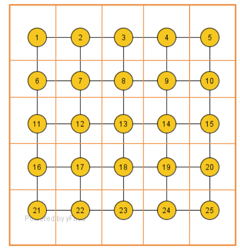
UCB Mote (Pister/Culler)

- Berkeley Mote
  - Commercially available
  - TinyOS: embedded OS running on motes.

1.2 Graphical Modeling

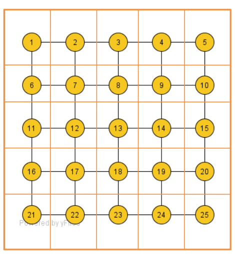
- Markov Random Fields
  - One common class of graphical models
  - Undirected graph G=(V,E)
  - Nodes represent random variables
  - Edges represent correlations between variables
  - Associated with potential functions
    - Ψ_i ~prior probabilities
    - $\Psi_{i,j}$ ~strength of correlations ~ joint probabilities

1.2 Graphical Modeling



- Each node represents temperature at a particular point in the room
- The edge potential represents the correlation between temperature of adjacent areas
- Node potentials represents prior temperature distributions at each location

1.2 Graphical Modeling



- Once we have
  - Node potentials
  - Edge potentials
  - Graphical model
- We can:
  - Calculate posterior probability distribution for each random variable
  - i.e. infer about the real temperature distributions at each location

1.3 Motivation

- Why not transmit all the data to a central based station?
- Why Graphical Models?
- Why RBP?

1.3 Motivation

- Why not transmit all the data to a central based station?
  - Heavy communication for continuous report
  - Especially bad for motes near base station
    - All the other sensor readings must be routed through
  - (if) motes need extra computations after inference
    - Redundant if retransmitting results back to motes

1.3 Motivation

- Why Graphical Models?
  - Deal with incomplete, unreliable information
  - Run in a fully distributed context

1.3 Motivation

- Why RBP?
  - Junction-Tree:
    - Exact inference…but depends on the graph structure
    - Very slow (exponential)
  - BP:
    - No guarantees on correctness, convergence
    - Works well in practice
  - RBP:
    - Modification of BP by adding suitable weights
    - Guarantees on stability and fixed point solutions

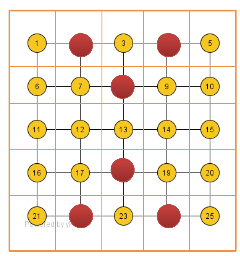
Part II: Problem Statement

→Inputs
→Outputs
→Assumptions

2.1 Inputs

- The inputs to the system
  - Global graphical model
    - Two types of nodes
    - Edge ~ correlations between nodes
  - Potentials for each node
  - Potentials for each edge
  - A mapping from nodes to motes

2.1 Inputs



- Two types of nodes
  - Evidence/Observable nodes
  - Latent/Unobservable nodes
- Observable node ~ sensor mote
- Latent nodes ~ physical location where inferred information is desired

2.2 Outputs

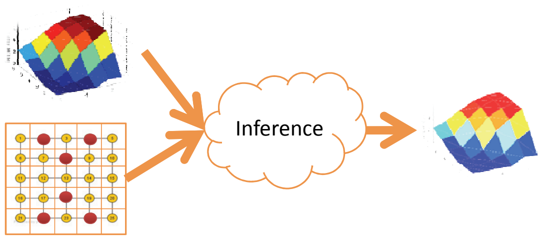
- Result of inference over the graphical model
  - Each mote contains the inferred data it owns
  - Each node yields a marginal probability for temperature at the corresponding location
  - Results are stored in a distributed fashion

2.3 Assumptions

- Assume:
  - Graphical Model is given
    - Latent vs. Observable
  - Random Variables are discrete
  - Potential functions are given
  - Mapping of nodes to motes is given

2.3 Summary

- Input: Sensor measurements + graphical model
- Output: Good “virtual measurements”



Part III: BP and RBP

3.1 Belief Propagation

- BP is used to compute:
  - Marginal Probability
  - Maximum A posterior Probability (MAP)
- For full description, please refer:
  - http://www.merl.com/reports/docs/TR2001-22.pdf

3.1 Belief Propagation

- The update equation:

$$m_{i\to j}(x_j) = \sum_{x_i} (\psi_i(x_i)\psi_{ij}(x_i, x_j) \prod_{k\in N(i)\backslash\{j\}} (m_{k\to i}(x_i)))$$

- It will be more helpful if we rewrite as:

3.1 Belief Propagation

$$m_{i\to j}(x_j) = \sum_{x_i} (\psi_i(x_i)\psi_{ij}(x_i, x_j) \prod_{k\in N(i)\backslash\{j\}} (m_{k\to i}(x_i)))$$

Incoming Message

$$M(x_i) = \prod_{k\in N(i)\backslash\{j\}} m_{k\to i}(x_i)$$

Outgoing Message

$$m_{i\to j}(x_j) = \sum_{x_i} \psi_i(x_i)\psi_{ij}(x_i, x_j) M(x_i)$$

3.1 Belief Propagation

$$M(x_i) = \prod_{k\in N(i)\backslash\{j\}} m_{k\to i}(x_i),$$

$$m_{i\to j}(x_j) = \sum_{x_i} \psi_i(x_i)\psi_{ij}(x_i, x_j) M(x_i)$$

- A recursive equation!
- Outgoing messages at each node depend on the incoming messages at that node
- These messages summarize the information from a portion of the graphical model
- May be computed sequentially in any order, or even partially or fully simultaneously

3.1 Belief Propagation

- To compute marginal probabilities:

$$p(x_i) \propto \psi_i(x_i) \prod_{k \in N(i)} m_{k \to i}(x_i)$$

- Compare to:

$$posterior \propto prior \times likelihood$$

3.2 Reweighted Belief Propagation

$$M_\mu(x_i) = \frac{\prod_{k \in N(i) \setminus \{j\}} m_{k \to i}(x_i)^{\mu_{ij}}}{m_{j \to i}(x_i)^{1-\mu_{ij}}}$$

$$m_{i \to j}(x_j) = \sum_{x_i} \psi_i(x_i) \psi_{ij}(x_i, x_j)^{\frac{1}{\mu_{ij}}} M_\mu(x_i)$$

$$RBP \to BP \text{ when } \mu = 1$$

3.2 Reweighted Belief Propagation

- How to choose μ ?
  - Ref: [M. Wainwright, AIStats 03]
  - E.g: For grid like structures

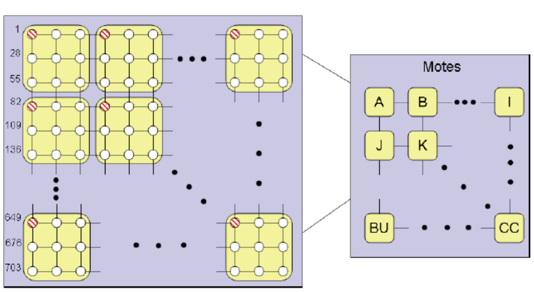
$$\mu_{ij} = \frac{|V|-1}{|E|}$$

- Guarantee convergence for proper μ_{ij}

Part IV: Architecture

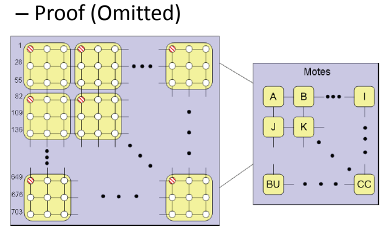
StatSense

Architecture



Architecture

- No Routing Property
- Proposition 1 [Ref to the original paper]
  - Proof (Omitted)



Architecture

- Message updating
  - SyncAllTalk
    - Limit messaging across motes when possible
    - “bursty” communication patterns
    - “Informativeness” vary greatly
  - SyncConstProb
    - Exchanges inter-mote message with probability p
  - AsyncConstProb
    - Relax the global time constraint
  - AsyncPropProb
    - p is proportional to the “informativeness” of the message

Architecture

- Handling communication failure

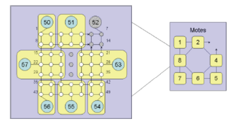


Figure 2: Dead Mote Graph

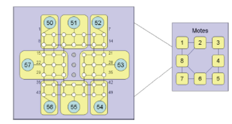


Figure 3: Dead Link Graph

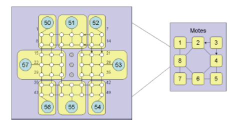


Figure 4: Asymmetric Mote Graph

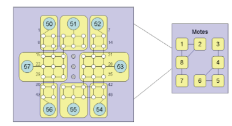


Figure 5: Partition Graph

Part V: Implementation

Implementation

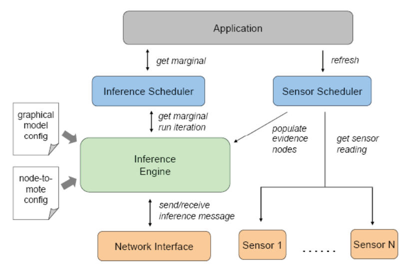
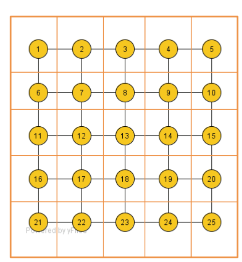


Figure 3: The StatSense architecture as implemented in nesC.

Part VI: Evaluation

6.1 Analysis of Communication Costs

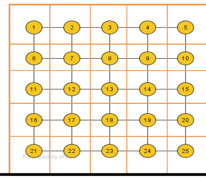


- Consider a grid of n^2 nodes
- Assume:
  - # of base stations = constant
  - # of neighbors = 4
  - # of BP iterations = t
  - size of single data reading=q

6.1 Analysis of Communication Costs

- Central based
  - O(n^2*q) total
  - O(nq) for each node
- Message passing
  - O(tq) for each node
- Save costs only when t<<O(n).
- This is possible!

- Consider a grid of n^2 nodes
- Assume:
  - # of base stations = constant
  - # of neighbors = 4
  - # of BP iterations = t
  - size of single data reading=q



6.2 Experiments

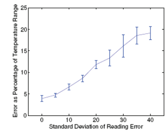


Figure 4: Average error as the gaussian error applied to the sensor observations increases.

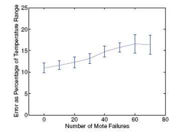


Figure 5: Average error as the number of dead motes increases.

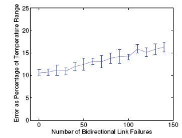


Figure 6: Average error as the number of dead symmetric links increases.

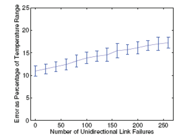


Figure 7: Average error as the number of dead asymmetric links increases.

6.2 Experiments (Cont')

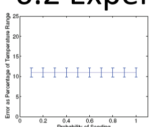


Figure 8: Average error as the probability of sending in a single iteration for SyncConstProb increases.

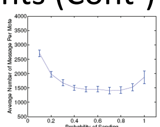


Figure 9: Number of messages sent as the probability of sending in a single iteration increases for SyncConstProb.

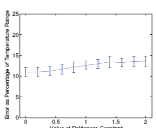


Figure 10: Average error as the exponent of the total variation difference between old and new messages increases for AyncPropProb.

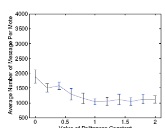


Figure 11: Number of messages sent as the exponent of the total variation difference between old and new messages increases for AyncPropProb.

Experiments (Cont')

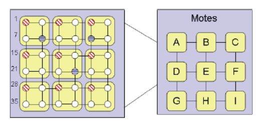


Figure 12: This figure illustrates the two layers of the Graphical Model (left) and Motes (right) which we used for our in-lab experiment. The red angle-slashed dots denote observation nodes, ie locations where we have sensor readings, and the blue-horizontal nodes denote locations where we logged unobserved temperature readings to determine error.

Node	actual temp	mean of marginal
7	36.8°C	34.7°C
10	66.0°C	55.5°C
21	33.7°C	35.1°C

Table 1: Comparison between actual and inferred temperatures for three nodes in the graphical model

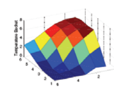


Figure 16: Temperature field for the experiment run on the deployment

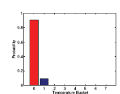


Figure 13: This is the histogram of the inferred distribution on Node 7 of our graphical model. The red bar is the bucket for the actual sensor reading.

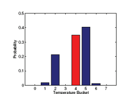


Figure 14: This is the histogram of the inferred distribution on Node 10 of our graphical model. The red bar is the bucket for the actual sensor reading.

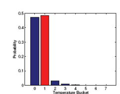


Figure 15: This is the histogram of the inferred distribution on Node 21 of our graphical model. The red bar is the bucket for the actual sensor reading.

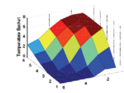
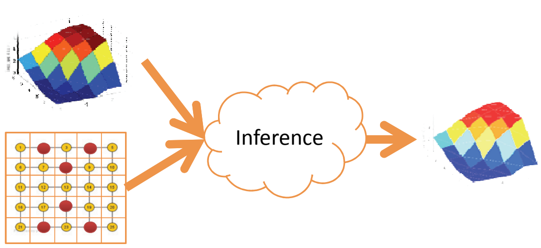


Figure 17: Temperature field for the experiment run in simulation

Summary

- Input: Sensor measurements + graphical model
- Output: Good "virtual measurements"



Discussion

Data dependence

≠

Physical Distance