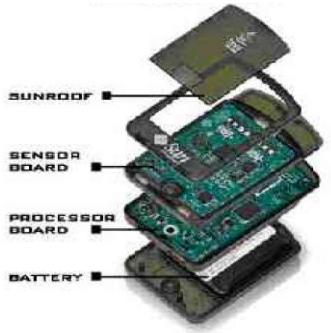
A Wavelet Transform for Efficient Consolidation of Sensor Relations with Quality Guarantees

Mirco Stern, Erik Buchmann, Klemens Böhm
Universität Karlsruhe (TH)
Germany

VLDB ‘09

Data Collection in Sensor Networks



- Sensor nodes monitor
  physical parameters of their environment
- Example:
  - Industrial setting: instrument production lines
  - Interests: maintenance of machinery; quality management; process compliance
- Our concern: (Efficient) Data acquisition
  - Query-interface to hide low-level programming and networking from user
  - Efficient query processing
    - Nodes are battery powered
    - Communication most expensive operation

⇒ Answer queries using a minimum amount of communication

Approximating Sensor Relations

- Common data acquisition task: Consolidation of current readings
  - Simplest form: “SELECT * FROM Sensors SAMPLE PERIOD x”
  - Monitoring, scientific environments, etc.

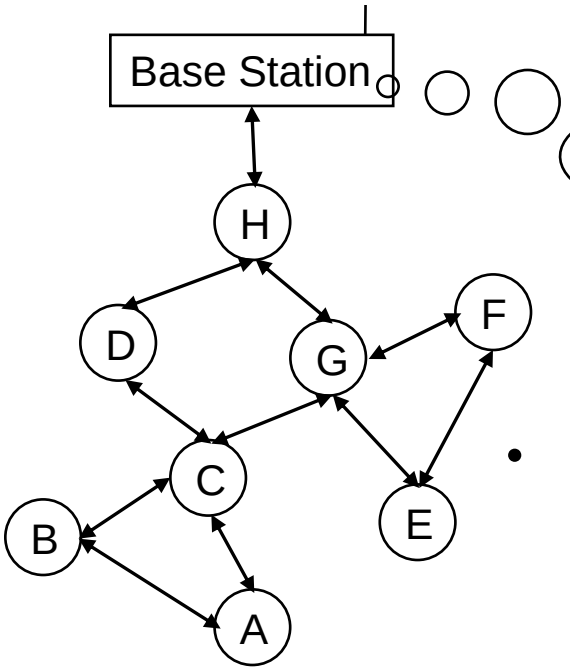
- Problem: High data volume

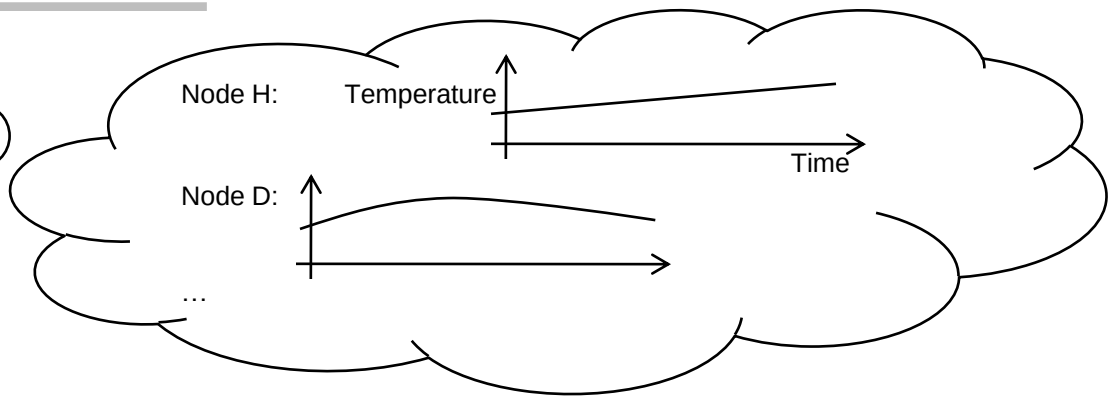
⇒ Data acquisition is communication-intensive

- Common observation: Accuracy tends to be less critical
  - As long as error bounds are provided (e.g., 22.4°C instead of 22.435…°C)
  - Chance of trading accuracy for communication

- Goal: Approximate snapshots of sensor relations under user-controlled error guarantees

Approximate Query Processing





- State-of-the-Art: Model-based approximations
  - Idea: Predict sensor readings at the base station
  - If accuracy of prediction is sufficient ⇒ no need to acquire current readings
  - Problem:

    Performance critically depends on the error that an application can tolerate
  - Further problem: Extensive training costs

Our Approach: SNAP

- To overcome these problems:
  Build upon different kind of synopsis – wavelet synopsis
  - Very effective at compressing data and providing accurate answers
- Challenge:
  Synopsis has to be constructed incrementally during data collection

⇒ Distributed construction of wavelet synopses in sensor networks

- SNAP (SNapshot APproximation)
  - Efficient consolidation of sensor readings
  - User-controlled error guarantees

SELECT $Att_1 \pm e_1, \ldots, Att_n \pm e_n$
FROM Sensors
WHERE predicates($Att_1, \ldots, Att_n$)
{SAMPLE PERIOD x | ONCE}

Agenda

- Motivation
- Background on Wavelet Synopses
- Distributing the Wavelet Transform
- Distributing the Thresholding
- Evaluation
- Conclusions

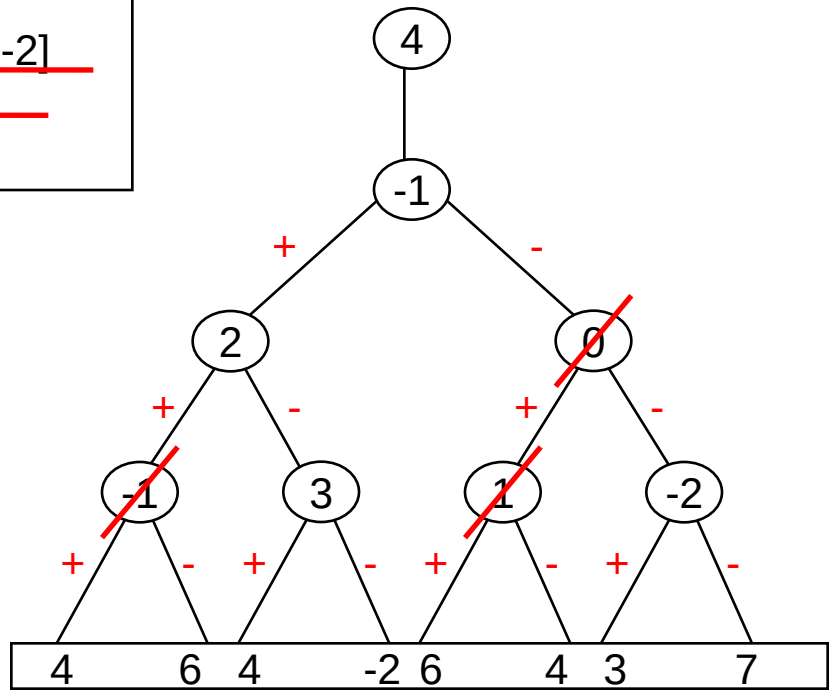
Background: Wavelet Transform

- Mathematical tool to decompose functions hierarchically
- Wavelet transform represents the function as
  - Coarse overall shape (approximation) plus
  - Detail coefficients (at increasing levels of granularity)
- Example: Haar Wavelet
  - Approximation: Pairwise averaging ($\frac{x+y}{2}$)
  - Corresponding "detail coefficients": Information that has been lost ($\frac{x-y}{2}$)
  - D = [4, 6, 4, -2, 6, 4, 3, 7] $\Rightarrow$ Transform: [4, -1, 2, 0, -1, 3, 1, -2]

Level	Approximations	Detail coefficients
3	[4, 6, 4, -2, 6, 4, 3, 7]	
2	[5, 1, 5, 5]	[-1, 3, 1, -2]
1	[3, 5]	[2, 0]
0	[4]	[-1]

Thresholding

Level	Approximations	Detail coefficients
3	[4, 6, 4, -2, 6, 4, 3, 7]	
2	[5, 1, 5, 5]	[-1, 3, 1, -2]
1	[3, 5]	[2, 0]
0	[4]	[-1]



Then: Thresholding

Compact synopsis by discarding detail coefficients

(as many as possible while guaranteeing the max. error!)

Wavelet Synopses – Concluding Remark

- Thresholding shrinks the data volume
- Why do we transform the data prior to thresholding?
  - Separated
    - “Important” information (overall view/ approximation)
    - Details

  ⇒ Discard only details – introduce only small errors

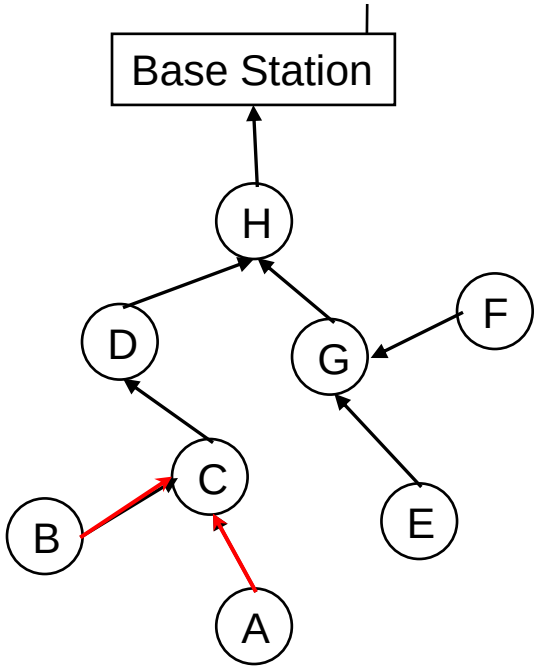
- Reason for applying wavelet transform (in our context):
  Obtain small detail coefficients

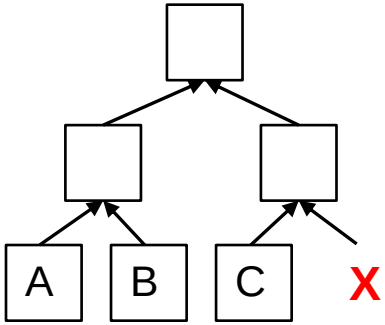
Agenda

- Motivation
- Background on Wavelet Synopses
- Distributing the Wavelet Transform
- Distributing the Thresholding
- Evaluation
- Conclusions

Problem in Distributing the Transform

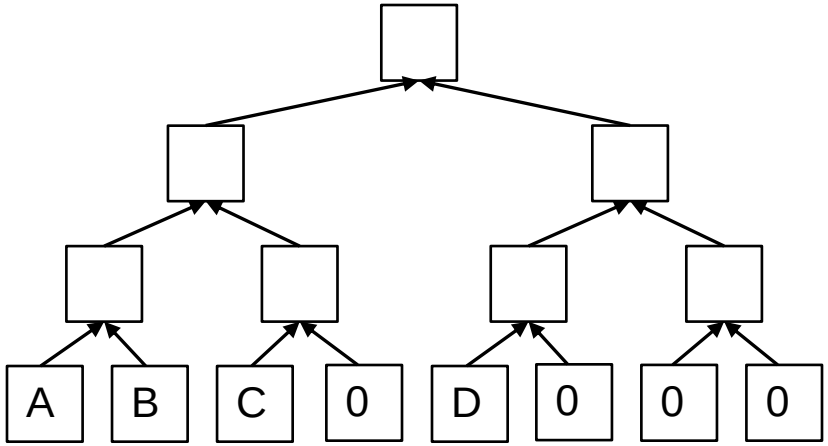
- Build upon a collection tree
- Apply the transform during forwarding
- Problem: Strong mismatch between
  - The topology of the network
  - Data flow of the transform

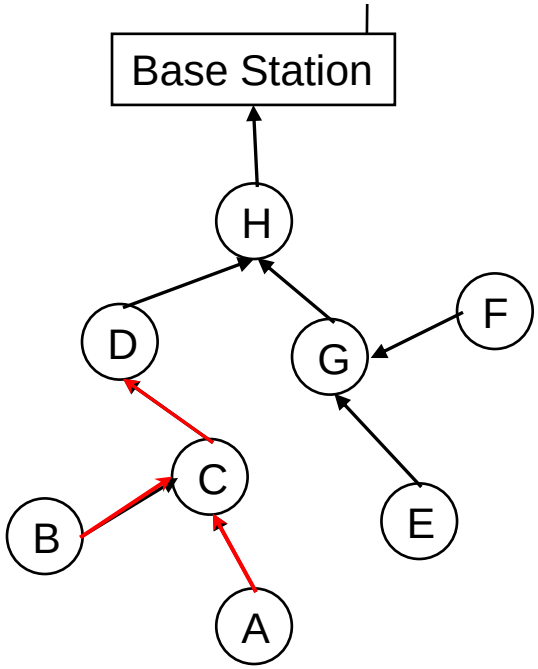




Approaches for Distributing the Transform

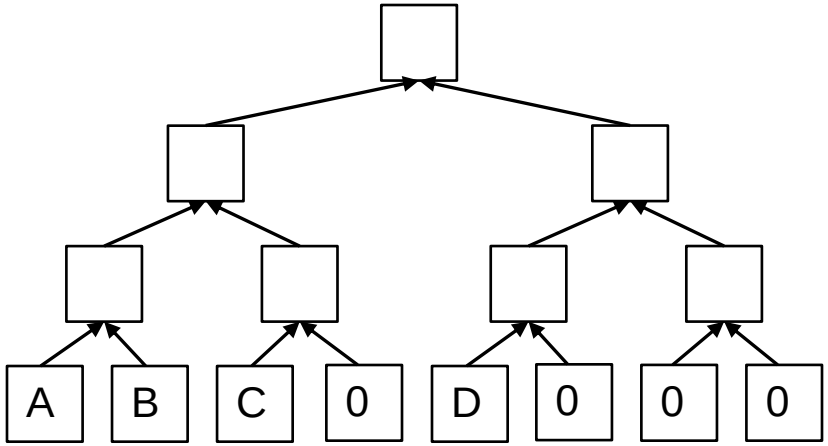
- Build upon a collection tree
- Apply the transform during forwarding
- Attempt (literature): Adjust the input data
  - "Zero Padding"

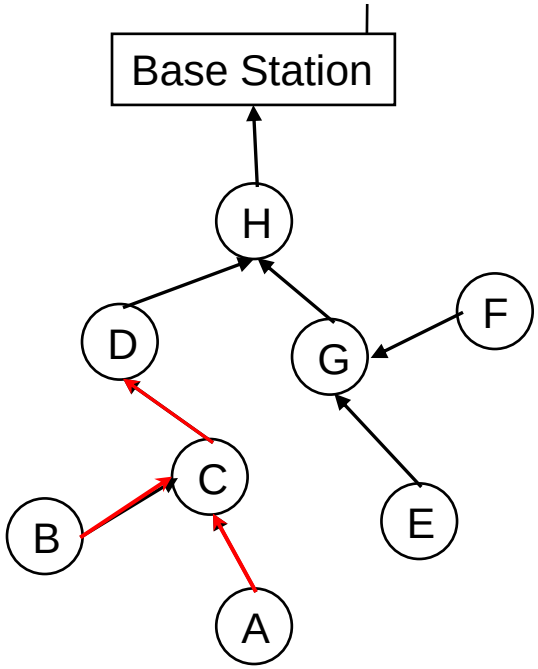




Approaches for Distributing the Transform

- Problem: Zero Padding adds a lot of data!
  - Many large detail coefficients!
- But: Goal is to REDUCE data

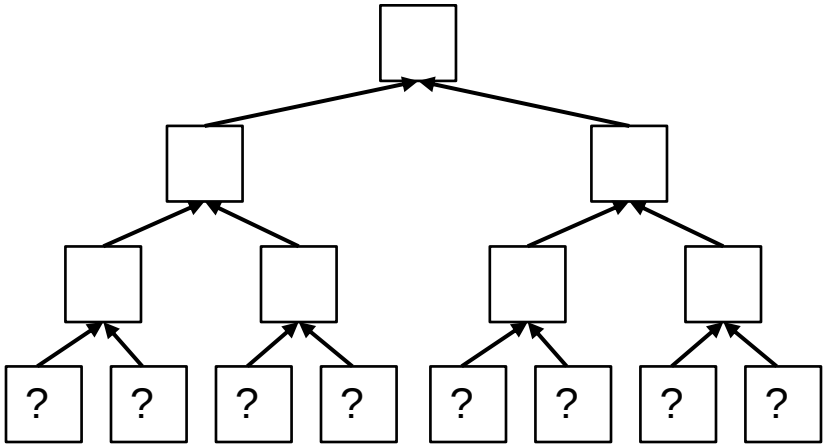


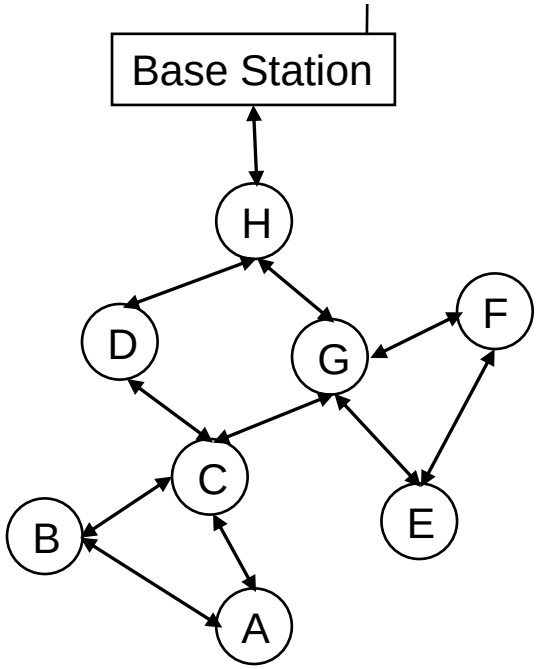


Alternative Approaches

- Alternative:

  Do not work on a collection tree
  - Choose the routing according to the transform
  - Map the transform onto the network

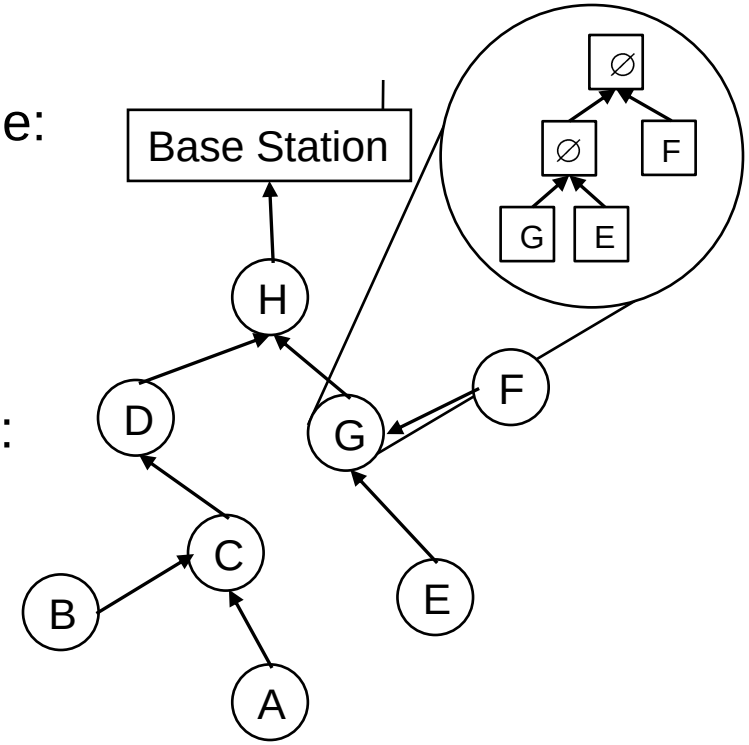




⇒ Results in longer path (a lot of forwarding)

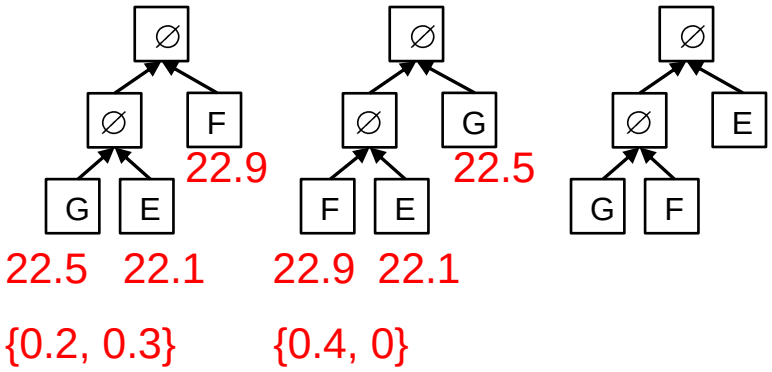
Our Approach

- Integrate the transform into an optimal routing structure
  - I.e., we build upon an optimal routing structure (shortest path)
- To cope with irregularity of the routing tree:
  Adjust the transform to routing tree
  - (≠ input data)
- Key for distributing the wavelet transform:
  Allow for an irregular transform
  - I.e., an unbalanced (binary) transform

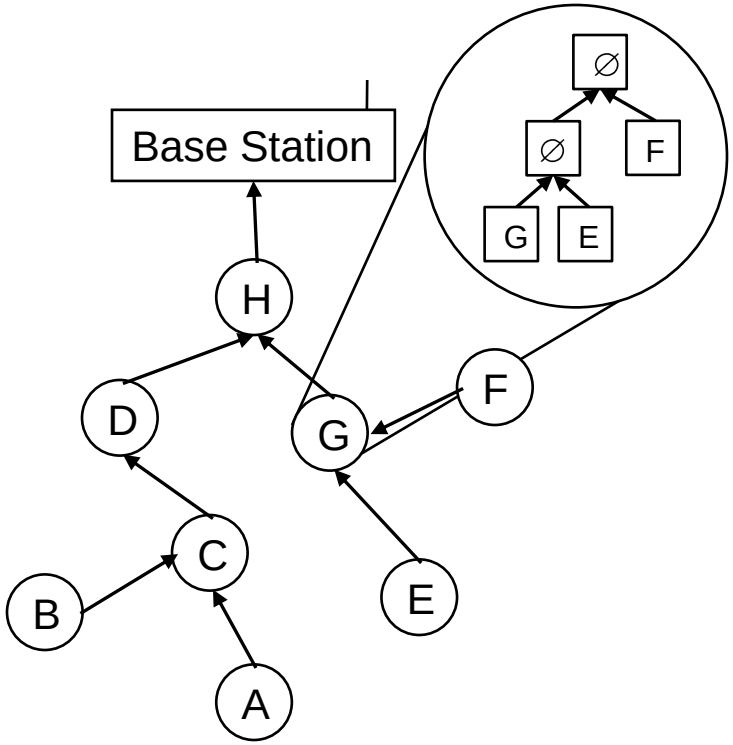


Finding an Optimal Integration

- How to choose among alternative transforms:



- Different alternatives yield different sets of detail coefficients
  - Influences size of the synopsis



Exact vs. Heuristical Optimization

- Finding the optimal integration is NP-hard
  - Decision is made by sensor nodes
    (integration is performed on-the-fly; ≠ precomputed)
- Propose a greedy heuristic:
  - In each step: Combine approximations that yield the smallest detail coefficient
- Also:

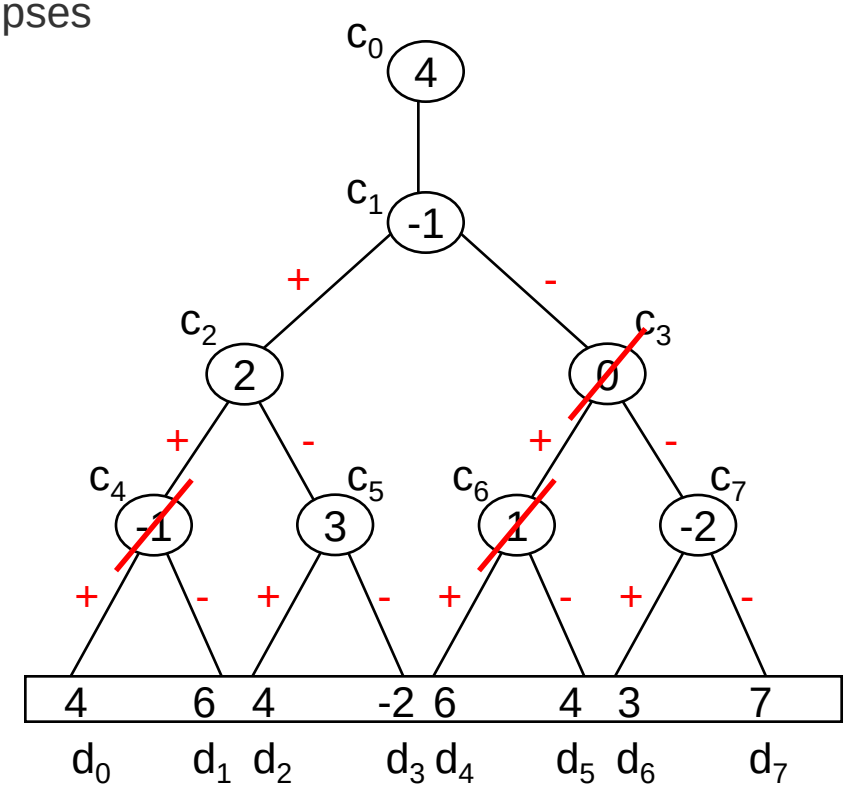
  Designed a DP-based algorithm that constructs the optimal solution
  - To evaluate the heuristic
  - Performs within 5% of optimum

Agenda

- Motivation
- Background on Wavelet Synopses
- Distributing the Wavelet Transform
- Distributing the Thresholding
- Evaluation
- Conclusions

Distributing the Thresholding

- Constructing a synopsis is different in a distributed setting
  - Costs of forwarding intermediate synopses
  - ⇒ Minimize sum of data volumes
- Problems:
  - Discarding coefficients leads to suboptimal synopses
  - **Substantial overhead**:
  1. Metadata for controlling the error (state of the computation)
  2. Coordinates



Entropy Coding for Data Reduction

⇒ We do not discard coefficients!

- Build upon a different mechanism for data reduction: Compression

- Underlying Mechanism:
  - Entropy Coding
    - Key for data reduction
    - Frequent coefficients: short codes;
      longer codes for less frequent coefficients
    - Works well if:
      (1) Distribution of detail coefficients is skewed → Wavelet Transform
      (2) Small number of different coefficients → Quantization

Entropy Coding for Data Reduction

⇒ We do not discard coefficients!

- Build upon a different mechanism for data reduction: Compression

- Underlying Mechanism:
  1. Quantization
  2. Wavelet Transform
  3. Entropy Coding
     - Key for data reduction
     - Frequent coefficients: short codes;
       longer codes for less frequent coefficients
     - Works well if:
       (1) Distribution of detail coefficients is skewed → Wavelet Transform
       (2) Small number of different coefficients → Quantization

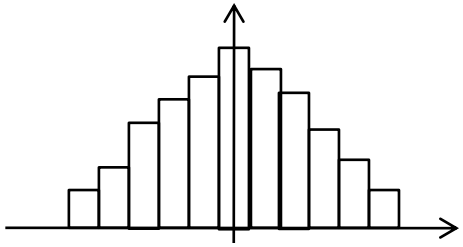
Role of Quantization

- Underlying Mechanism:
  1. Quantization
     - Use user-provided error bound:
       Large error bound ⇒ small number of detail coefficients!
     - No state to forward
       - At each node
  2. Wavelet Transform
  3. Entropy Coding

Role of Wavelet Transform

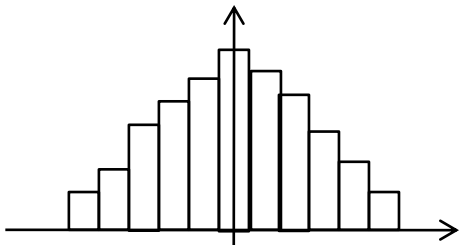
- Underlying Mechanism:
  1. Quantization
  2. Wavelet Transform
     - Generates a skewed distribution

     (Goal of transform is to minimize detail coefficients)
  3. Entropy Coding



Distributing the Data Reduction

- Underlying Mechanism:
  1. Quantization
  2. Wavelet Transform
  3. Entropy Coding
     - Frequent coefficients: short codes;
       longer codes for less frequent coefficients



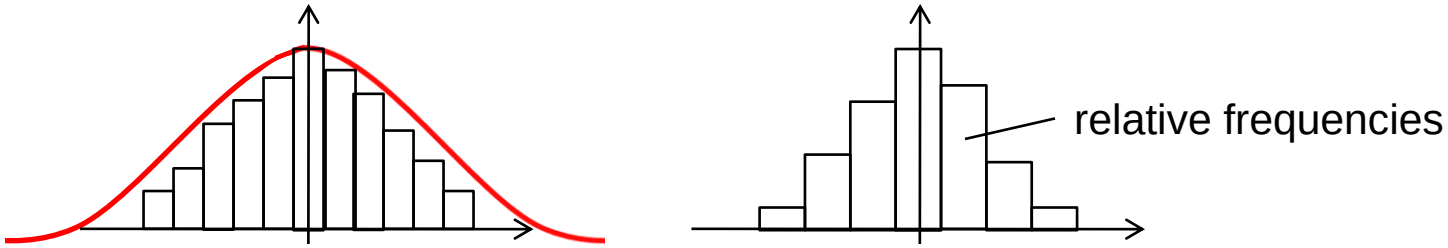
- So far:

  Decided to build upon a different mechanism for data reduction
  - Entropy Coding instead of discarding coefficients
- Challenge: Devise a **distributed** approach
  - Encoding requires each node to know the frequencies of the detail coefficients

Estimating Frequencies

2 problems in knowing the frequencies of detail coefficients:

1. Overall synopsis is unknown by the time of encoding
   - Use estimation based on experience (similar to selectivity estimation)
2. Frequency depends on error bounds
   - Larger error bounds ⇒ smaller number of different coefficients ⇒ different distributions



   - Estimate relative frequencies by discretizing continuous distribution (max. error)
     - “Continuous distribution”: error bound = 0
     - Mathematically sound

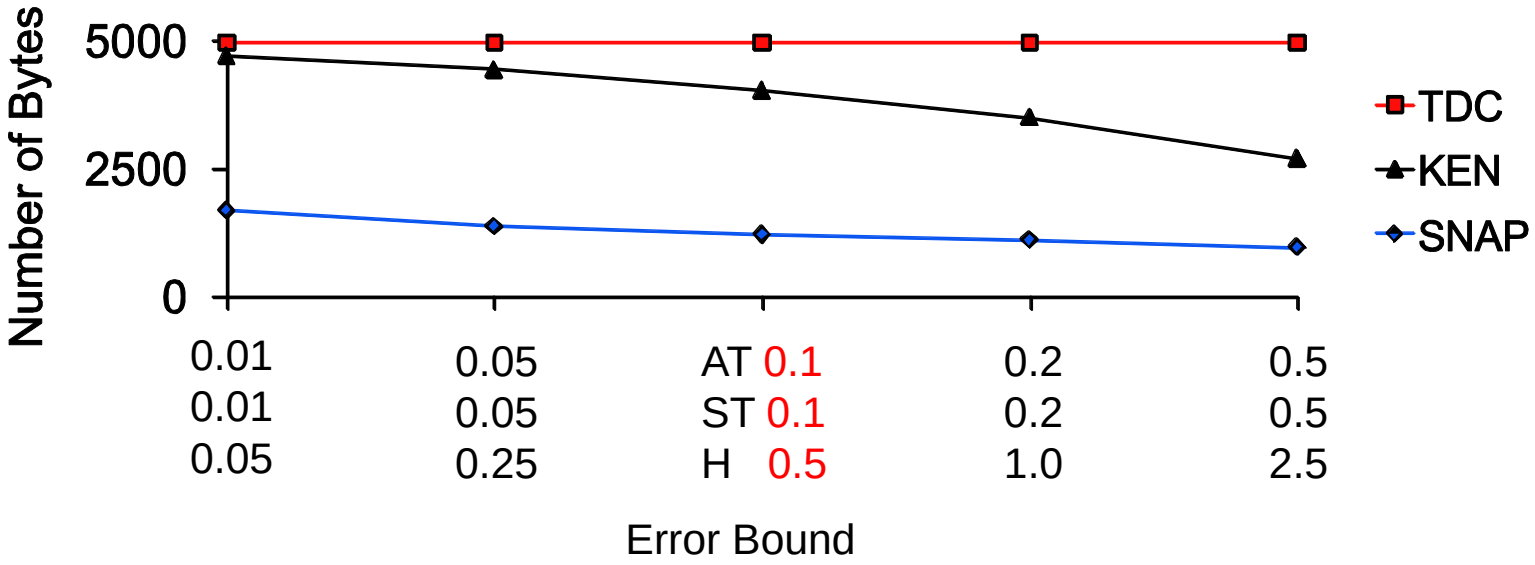
Agenda

- Motivation
- Background on Wavelet Synopses
- Distributing the Wavelet Transform
- Distributing the Thresholding
- Evaluation
- Conclusions

Experimental Setup

- Goal: Demonstrate data reduction capability of SNAP
  - Even for tight error bounds
- Implemented a prototype in ns-2
  - Controlled environment, repeatability
- Comparison schemes
  - Wavelet-based approaches (zero padding, etc.)
  - Model-based approaches (KEN)
  - Tree-based Data Collection (TDC)
- Data Sets and Setting
  - Mostly real world data (traces from sensor networks); organized setting to reflect original data collection
  - Synthetic data for scalability and extreme data sets

Comparison to Model-based Approaches



SELECT ID, Ambient Temp ± 0.1°C, Surface Temp ± 0.1°C, Humidity ± 0.5%
FROM Sensors
SAMPLE PERIOD 300s

Agenda

- Motivation
- Background on Wavelet Synopses
- Distributing the Wavelet Transform
- Distributing the Thresholding
- Evaluation
- Conclusions

Conclusions

- SNAP: Wavelet-based approach
  - Handling queries with a low selectivity in sensor networks
  - SNAP constructs synopsis during data collection incrementally
- Main contributions
  - Distribute the wavelet transform
    - Integrated the transform into an unmodified routing tree
  - Distribute the construction of the synopsis
    - Underlying mechanism: Compactly encode coefficients instead of discarding
    - For distribution: Estimate the frequencies of the detail coefficients
- Experiments indicate that SNAP
  - Can reduce the data volume by up to a factor of five
  - Improves accuracy for data consolidation by more than an order of magnitude