Tributaries and Deltas: Efficient and Robust Aggregation in Sensor Network Streams

Amit Manjhi, Suman Nath, Phillip B. Gibbons
Carnegie Mellon University Intel Research Pittsburgh

Presented by:
Simrat Wason
Duke University

Acknowledgements for slides:
Amit Manjhi, SIGMOD’05

Outline
- Background and Motivation
- Tributary-Delta Approach
- Simple Aggregates in TD framework
- Frequent items in TD Framework
- Evaluation
- Related work and conclusion

Sensors
- Battery operated tiny devices

Constraints:
- Conserving battery power is important
- Comm. Consumes more energy than local computation
- Operate in dynamic, harsh environments

Sensor networks
Important type of query is computing aggregates e.g., total number of live sensors
(route all the sensor readings to the base station)

In-network aggregation is performed to save communication

Existing energy-efficient in-network approaches: Tree and Multi-path

Tree
[TinyDB, Cougar]
- Non-robust topology
- Exact answer

Multi-path
[Canisim et al. ICDE’04]
+ Robust Topology
- Approximate answer
Tree and Multi-path Tradeoffs

Can we get the best of both by adapting to changing loss rates?

Loss rate varies with change in conditions

SOLUTION: Tributary-Delta

Approach to In-network aggregation in sensor networks

Run tree and multi-path simultaneously in different parts of the Network
- as Energy-efficient as tree or multi-path
- Significant error reduction
- Multi-path region adapts to loss rate

Outline
- Background and motivation
- Tributary-Delta
- Simple aggregates in TD framework
- Frequent Items in TD framework
- Evaluation
- Related work and conclusion

How does Tributary-Delta work?
- Correctness: A tree node should not receive aggregates from a multi-path node
- Gives rise to a delta at the centre (multi-path aggregation is used in the nodes at the centre)

Computing Aggregates in the Tributary-Delta Framework

1. Each tree node
   - Tree Algorithm: Generate multi-path partial results

2. Each multi-path node
   - Multi-path Algorithm: Generate multi-path partial results

3. Nodes at the boundary
   - Conversion Function: Convert tree results to multi-path results

Expand or shrink the delta region
- Expand delta → increases robustness
- Shrink delta → lowers approximation error
Example Aggregates
- Many useful aggregates can be readily computed within the Tributary-Delta framework
- Missing piece: a suitable conversion function
- Conversion functions for several aggregates:
  - Count
  - Sum, Average
  - Top-k
  - Uniform sample

Outline
- Background and motivation
- Tributary-Delta
- Simple aggregates in TD framework
- Frequent Items in TD framework
- Evaluation
- Related work and conclusion

Finding Frequent Items
- Tree Algorithm:
  - Previous work [Greenwald, Khanna PODS '04, Manjhi et al. ICDE '05]
  - Tree algorithm achieves optimal bound for total communication
- Multi-path Algorithm:
  - Previous work [Nath et al. SenSys '04]
  - Multi-path algorithm is more accurate than previous work
- Conversion Function

Framework for finding Freq. Items
1. **Add** frequency counts from children
2. **Decrement** frequency counts
3. **Drop** counters that are below zero

Multi-path Algorithm for Freq. Items
1. **Add** → Duplicate insensitive addition
2. **Decrement** → Duplicate insensitive subtraction
3. **Drop counters below zero** → (rising) threshold

These steps are repeated at each internal node; decrements depend on height in the tree

How much to decrement at different levels?
- Max possible error ε
- Minimizes total communication Error
- Minimizes geometric decrease in decrement, e.g.: 0.5%, 0.25%, 0.125%, ...
- Geometric decrease in decrement, ε=0.1%
- Late Drop
- Early Drop

Threshold is maintained based on careful analysis
Paper has details on lowering communication
Outline

- Background and motivation
- Tributary-Delta
- Simple aggregates in TD framework
- Frequent Items in TD framework
- Evaluation
- Related works and conclusion

Evaluation Methodology

- The TAG Simulator [Madden et al. OSDI '02]
- Topology: 600 random sensors in 20 x 20
  - Base station is at the center
- Approaches:
  - Tree-based scheme: TAG
  - Multi-path scheme: Synopsis Diffusion [Nath et al. SenSys '04]
  - TD-Coarse: uniform expansion

Effects of regional loss rate

![Graph showing effects of regional loss rate]

**All four approaches use same energy**

Effects of global loss rate

![Graph showing effects of global loss rate]

1. Methods effectively combine the benefits: perform better than either existing approach
2. All four approaches use same energy

Computation of frequent items

![Graph showing computation of frequent items]

**False positives < 3%**

**Data from real sensor deployment**

**Loss rate = 0.05**

**Varying loss rate**

RELATED WORK:

- Existing in-network aggregation algorithms
  - Tree: TinyDB [Madden et al. SIGMOD '03]
  - Multi-path: Considine et al. ICDE '04, Bawa et al. SIGMOD '04, Nath et al. SenSys '04
- Adapting to changes in the environment
  - Directed Diffusion [Intanagowiwat et al. MobiCOM '00], TAG [Madden et al. OSDI '02]
- Frequent items and quantiles
  - Manku, Motwani VLDB '02, Greenwald, Khanna PODS '04, Manjhi et al. ICDE '05
Discussion

- Are frequent item queries interesting in sensor networks?
- Can we lift the constraint that an M edge can never be incident on a T vertex?
- Threshold on the minimum percentage of nodes that should contribute to the aggregate answer
- Overhead, convergence, and speed of adaptivity

CONCLUSION:

- Tributary-Delta: energy-efficient, and robust solution
  - Combines benefits of existing tree- and multi-path based approaches
  - Adapt to changing network conditions
- Algorithms for finding frequent items
- Results confirm the advantages
  - Error reduction is up to a factor of 3

FUTURE WORK:

- Deployment in a real scenario — incorporate in TinyDB

- Add other aggregates to the suite of aggregates