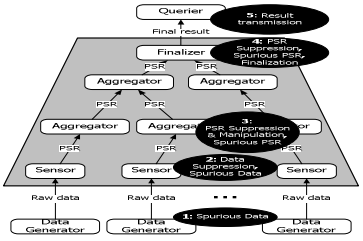
Proof Sketches: Verifiable In-Network Aggregation



Minos Garofalakis **Joe Hellerstein** **Petros Maniatis**

Yahoo! Research, UC Berkeley, Intel Research Berkeley

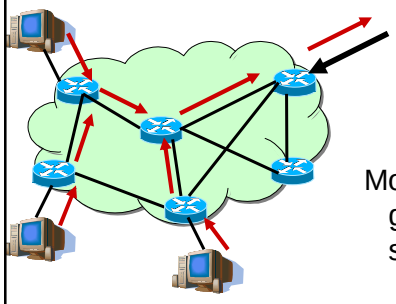
minos@yahoo-inc.com, minos@cs.berkeley.edu

YAHOO! RESEARCH

Introduction & Motivation

Context: Distributed, in-network aggregation

- Network monitoring, sensornet/p2p query processing, …
- Data is distributed – cannot afford to warehouse!
- Approximations are often sufficient
  - Can tradeoff approximation quality with communication



Querier: "How many Win-XP hosts running patch X have CPU utilization > 95%?"

"Predicate poll" query

More general aggregate queries (SUM, AVG), general-purpose summaries (e.g., random samples) of (sub)populations

YAHOO! RESEARCH

In-Network Aggregation

Typical assumption: Benign aggregation infrastructure

- Aggregator nodes cannot "misbehave"

BUT, aggregators are often untrusted!

- 3rd party hosted operations (e.g., Akamai), shared infrastructure, viruses/worms, …

Challenge: *Verifiable*, efficient, in-network aggregation

- Provide trustworthy, guaranteed-quality results with potentially malicious aggregators

YAHOO! RESEARCH

Our Contributions

Proof Sketches: Family of certificates for verifiable, approximate, in-network aggregation

- Concise sketch synopses ➔ Communication-efficient
- Guarantee detection of malicious tampering whp if result is perturbed by more than a small error bound

Basic Technique: Combines FM sketch with compact Authentication Manifest (AM)

- Prevents inflation through crypto signatures; bounds deflation through *complementary deflation detection*

Extensions: Verifiable random sampling; verifiable aggregates over *multi-tuple* nodes

YAHOO! RESEARCH

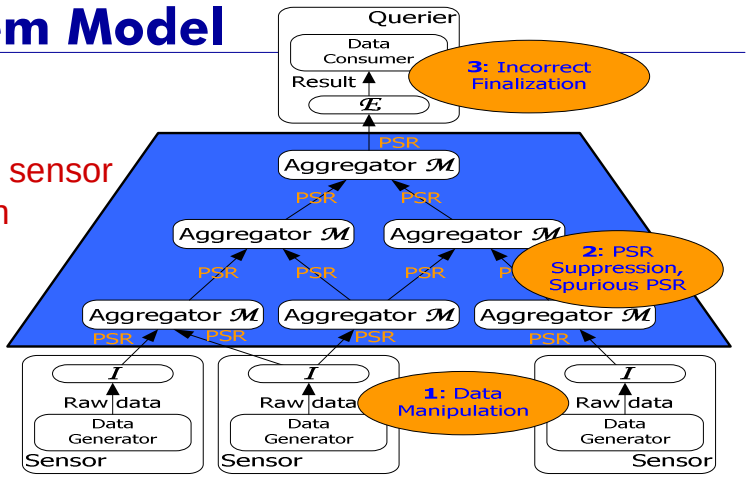
Talk Outline

- *Introduction and Motivation*
- *Overview of Contributions*
- System Model
- AM-FM Proof Sketches
- Extensions
  - Verifiable random samples
  - Verifiable aggregation over multi-tuple nodes
- Experimental Results
- Conclusions

YAHOO! RESEARCH

System Model

U = size of sensor population



Inflation Attacks: Aggregators can manipulate or inject spurious PSRs

Deflation Attacks: Aggregators can suppress valid PSRs

YAHOO! RESEARCH

A Naïve Inflation Detector

Straightforward application of crypto signatures

- Each sensor node crypto signs each tuple satisfying the predicate poll, and sends up the tuple + signature
- Aggregators simply union the signed tuple sets and forward up the tree

Aggregators cannot forge sensor tuples

- Within crypto function guarantees

BUT, size of Authentication Manifest (AM) = size of answer set

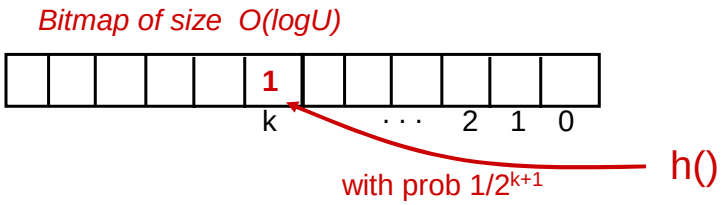
- ***O(U) in general!***

YAHOO! RESEARCH

Solution: AM-FM Proof Sketches

Sketch and AM structure of size only O(logU)

Based on the *FM sketch* for distinct-element counting



P[h(x)=0] = ½, P[h(x)=1] = ¼, P[h(x)=2] = 1/8, …

Index of rightmost zero ~ log(Count)

O(log(1/δ)/ε²) sketches to get an (ε,δ)–estimate of the Count

YAHOO! RESEARCH

Adding AM to FM: Inflation Prevention

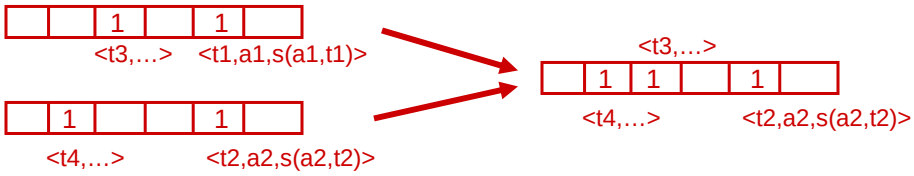
Observation: Each FM sketch bit is an independent function of the input tuples

AM = Authenticate each 1-bit in the FM sketch using a *signed "witness"/exemplar sensor tuple*

- Crypto-signed tuple that turns that bit on

Aggregators: Merge input PSRs (AM-FM sketches)

- *OR* the FM sketches
- Keep a single exemplar for each 1-bit



- *Size = O(logU)* – Cannot forge 1-bits

AM-FM Proof Sketches: Bounding Deflation

Malicious aggregator can omit 1-bits & witnesses from sketch → Underestimate predicate poll count

Approach: Complementary Deflation Detection

- Assumes that we know sensor count U
- Use AM-FM to estimate count for both pred and !pred
- Check that $C_{pred} + C_{!pred}$ is close to U (based on sketching approximation guarantees)
  - Adversary cannot inflate $C_{!pred}$ to compensate for deflating C_{pred}
  - Sum check will catch significant deviations

More Formally...

Assume $O(\log(2/\delta)/\varepsilon^2)$ AM-FM proof sketches to estimate C_{pred} and $C_{!pred}$

Verification Condition: Flag adversarial attack if $C_{pred} + C_{!pred} < (1-\varepsilon)U$

Theorem: If verification step is successful, the AM-FM estimate is within $\pm 2\varepsilon U$ of the true C_{pred} whp

- Adversary cannot deflate the result by more than $2\varepsilon U$ without being detected whp
- Relative error guarantees for *high-selectivity predicates*

YAHOO! RESEARCH

Verifiable Random Sampling

Build a general-purpose, verifiable synopsis of node data

- Can support arbitrary predicates, quantile/heavy-hitter queries, …

Traditional (eg, reservoir) sampling + authentication fails

- Adversary can arbitrarily bias the sample

Solution: AM-Sample Proof Sketches

- Use FM hashing to sample, retain tuples + AMs for all tuples mapping above a certain level
- *A la* Distinct Sampling *[Gibbons'01]* – adapt level based on target sample size
- Easily merged up the tree using max-level
- Verification condition and error guarantees based on target sample size and knowledge of U

YAHOO! RESEARCH

Aggregates over Multi-Tuple Nodes

So far, focus on predicate poll queries

- Each sensor contributes $\leq$ 1 tuple to result

Key Issue: Knowing the *total number of tuples M*

- With known M, our earlier results and analysis apply

Approach: Verifiable approximate counting algorithm

- Estimate M using a logarithmic number of simple AM-FM predicate polls
- To within a given accuracy θ, using predicate polls of the form

  Fraction of sensors with #tuples $\geq (1+\theta)^k$
- Detailed algorithm, analysis, … in the paper

YAHOO! RESEARCH

Other Extensions / Issues

Discuss "generalized template" for proof sketches to support verifiable query results

- E.g., Bloom-filter proof sketch

Accountability: Trace-back mechanisms for pinpointing attackers

Only approximate knowledge of population size U

YAHOO! RESEARCH

Experimental Study

Study *average-case behavior* of AM-FM proof sketches for verifiable predicate polls

Population of 100K sensors, fixed number of sketches to 256

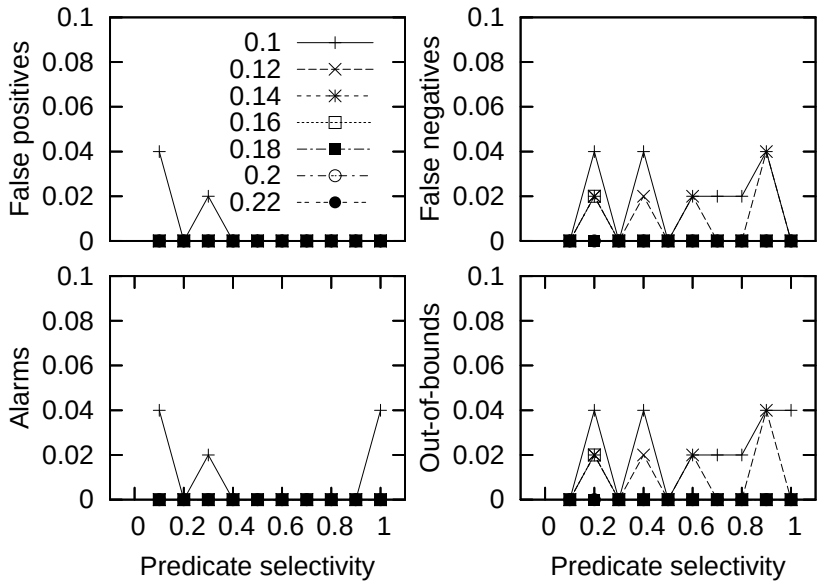
- About 4% of space for "naïve"
- $\varepsilon \approx 0.15$ wp 0.8

Parameters

- Predicate selectivity
- "Coverage" of malicious aggregators
- Two adversarial strategies (Targeted, Safe)

YAHOO! RESEARCH

Some Results: Benign Population



YAHOO! RESEARCH

Experimental Summary

Average case behavior is better than (worst-case) bounds suggest

- Adversary has even less "wiggle room" to deflate result without being detected

Bounds based on worst case for sketch approximation and combination of pred/!pred estimates

Adversary typically has limited coverage in the aggregation tree

- Can only affect a small fraction of the aggregated results

YAHOO! RESEARCH

Conclusions

Introduced *Proof-Sketches* – *first* compact certificate structure for verifiable, in-network aggregation

Basic technique: AM-FM proof sketch

- Adds concise AM to basic FM sketch; prevents deflation through *complementary deflation detection*

Extensions

- Verifiable random sampling
- Approximate verifiable counting for general aggregates over multi-tuple nodes

Future: Extending ideas and methodology to more general approximate in-network queries (e.g., joins)

YAHOO! RESEARCH

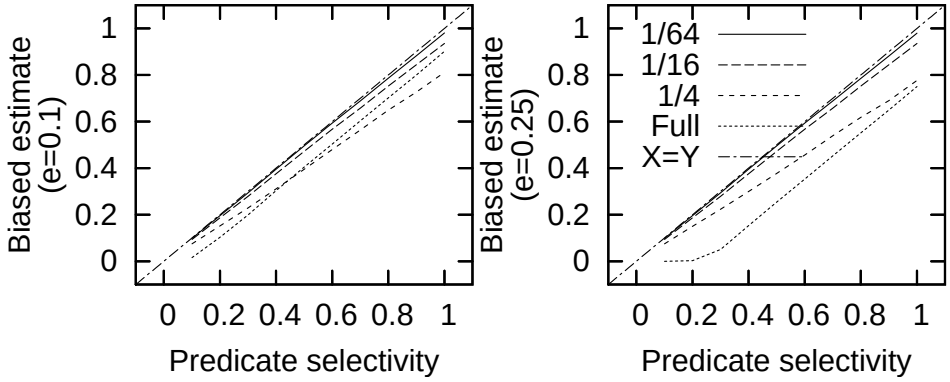
Thank you!

http://www.cs.berkeley.edu/~minos/

minos@yahoo-inc.com, minos@cs.berkeley.edu

YAHOO! RESEARCH

Some Results: Safe Adversary



YAHOO! RESEARCH