Adaptive Sampling: From Data Streams to Graph Streams

Nick Duffield

Texas A&M University

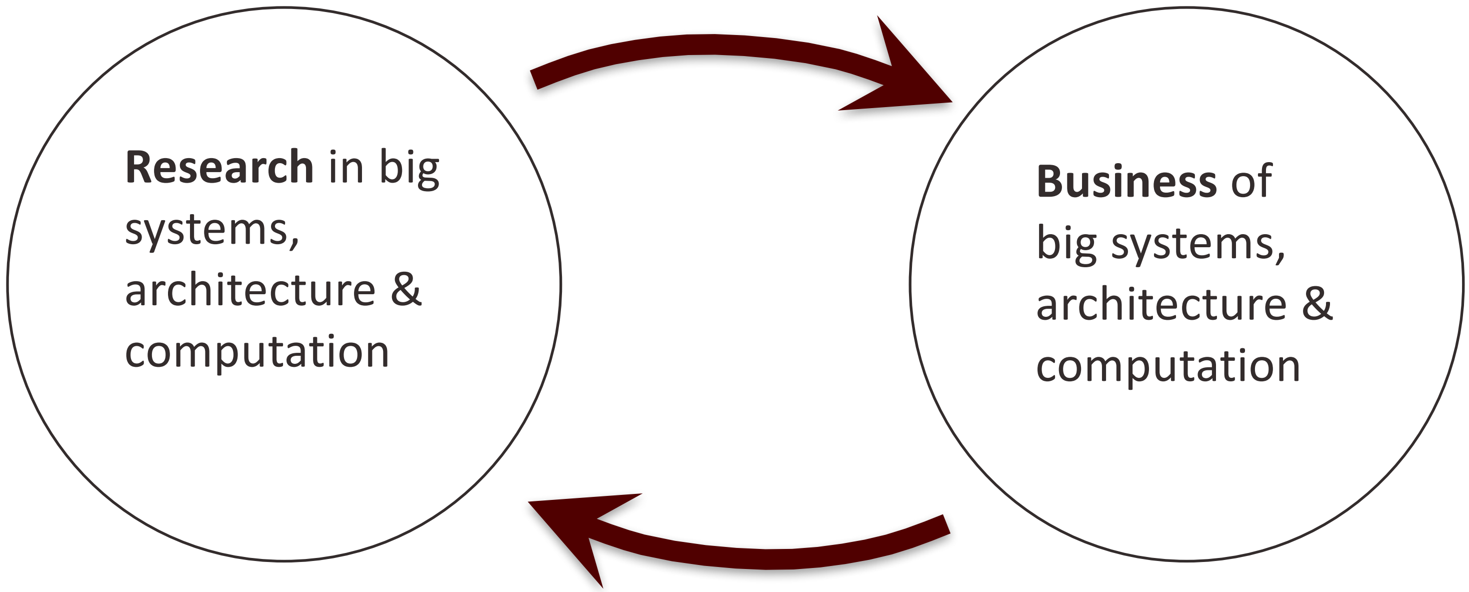
http://nickduffield.net/work

Big Data is Big Business

- Exciting advances in systems, architecture, computation

- Should research focus on big solutions to big problems?

Does Big Data need Big Systems?



- Ingenious ways of throwing resources at problems
- Are cycles and hardware no longer cost limited?
- Or is this a cycle of resource addiction?

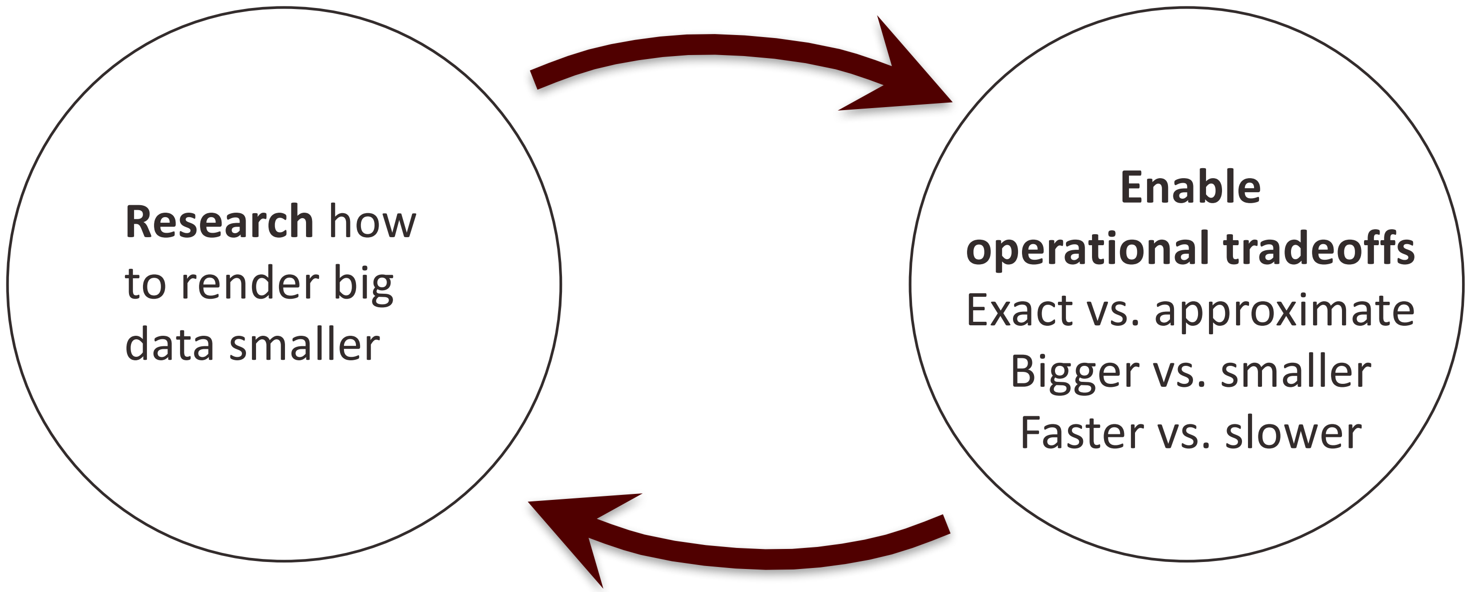
"But I had to grow bigger. So bigger I got"

The Lorax
Dr. Suess
1971

"Just because you can, doesn't mean you should"

Popular wisdom

Research Big to Execute Small

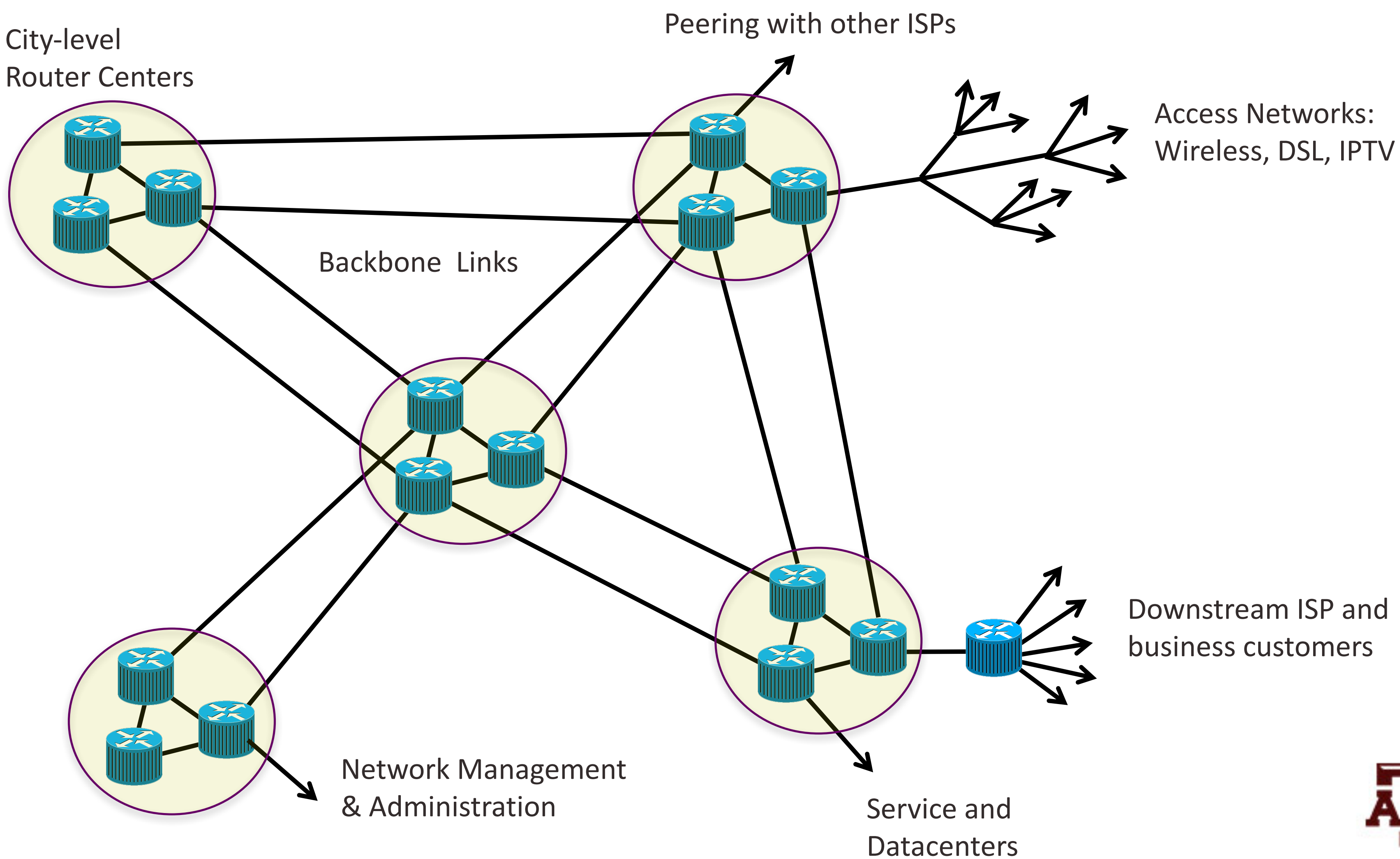


- Often want fast answers to retrospective queries
  - Data-driven automated control, interactive data analysis
- Approximate answers to queries often sufficient
  - Compare with modeling uncertainties, uncontrolled variables
- Experience? ISPs have worked with Big Data at network scale for years:
  - Operational datasets used to manage network over range of timescales
    - capacity planning (months), …. , detecting network attacks (seconds)

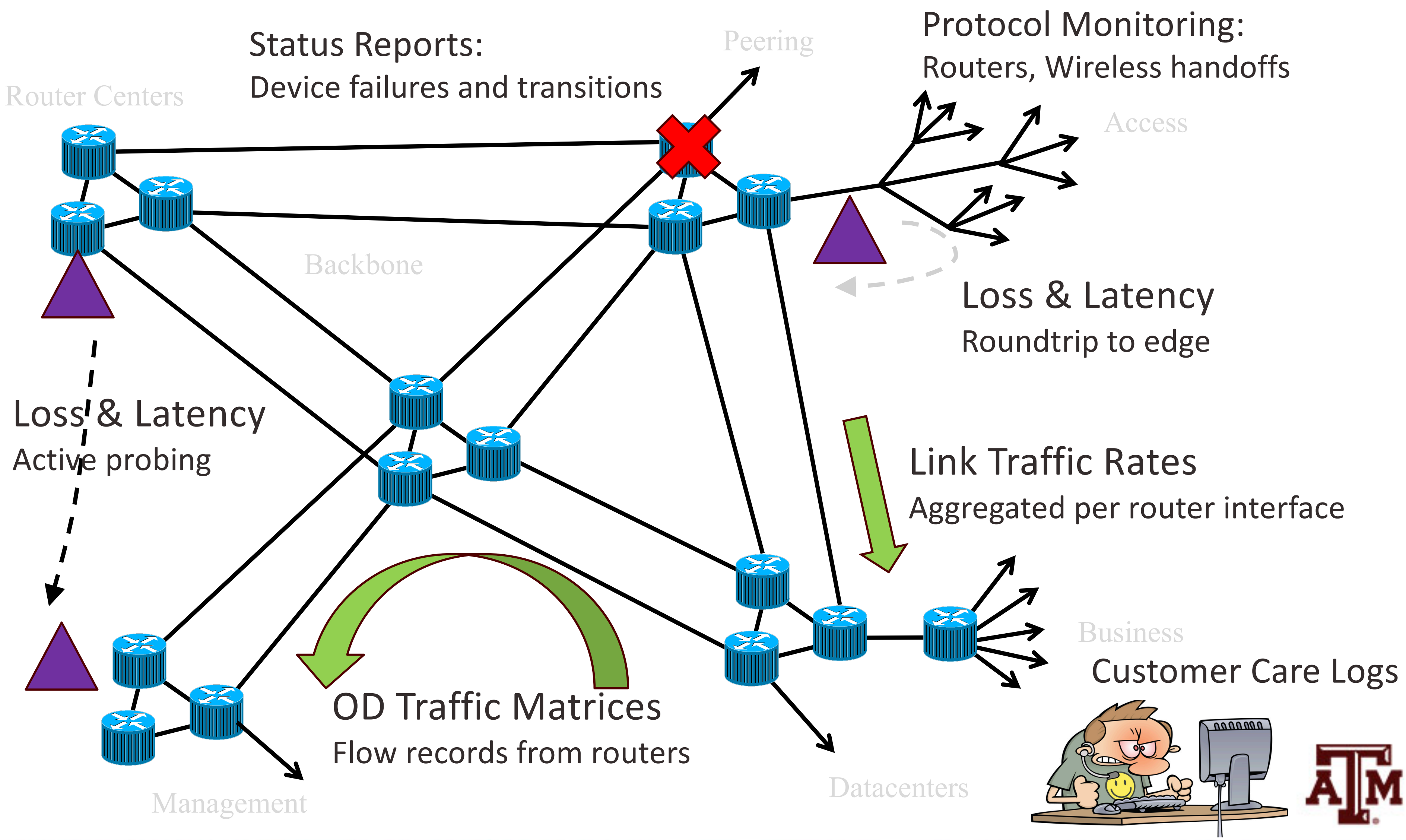
This talk

- Adaptive sampling in data streams
  - Aim: Constructing a reference sample for queries
  - Matching data characteristics to queries
    - Non-uniform sampling for heavy tails
    - Setting: stream sampling in ISP measurements
- Develop approach from graph stream sampling
  - Target queries: subgraph counts
  - Adapt sampling probabilities for arriving edges
    - Depends on role in sampled topology
    - Enhance ability to query target subgraphs

Structure of Large ISP Networks



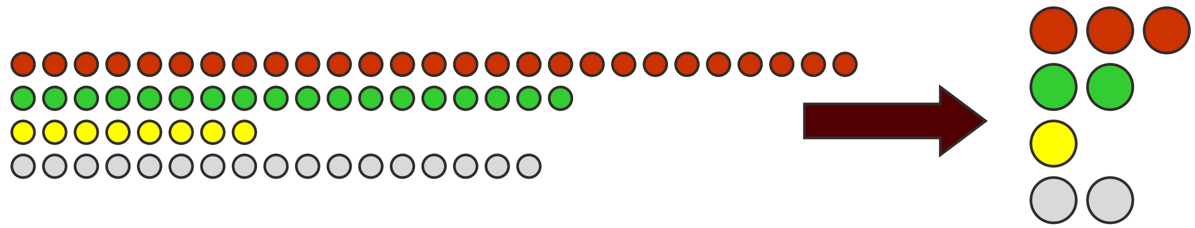
Operational ISP Network Data



Why Summarize ISP Operational Data?

- Limited bandwidth
  - Processing cycles within measurement devices
  - For transmission of data to collectors
- Limited storage
  - Infeasible to accumulate raw data streams over extended periods
- Limited time
  - Need fast query response
  - Infeasible to run exploratory queries over full data

Why Sample?



- Sampling has an intuitive semantics
  - We obtain a smaller data set with the same structure
- Estimating on a sample is often straightforward
  - Run the analysis on the sample that you would on the full data
- Futureproof
  - Don’t need to know queries at time of sampling
    - “Where/where did that suspicious UDP port first become so active?”
    - “Which is the most active IP address within than anomalous subnet?”
  - Contrast with other types of summary:
    - can’t drill down into aggregates

Sampling as a Mediator of Constraints

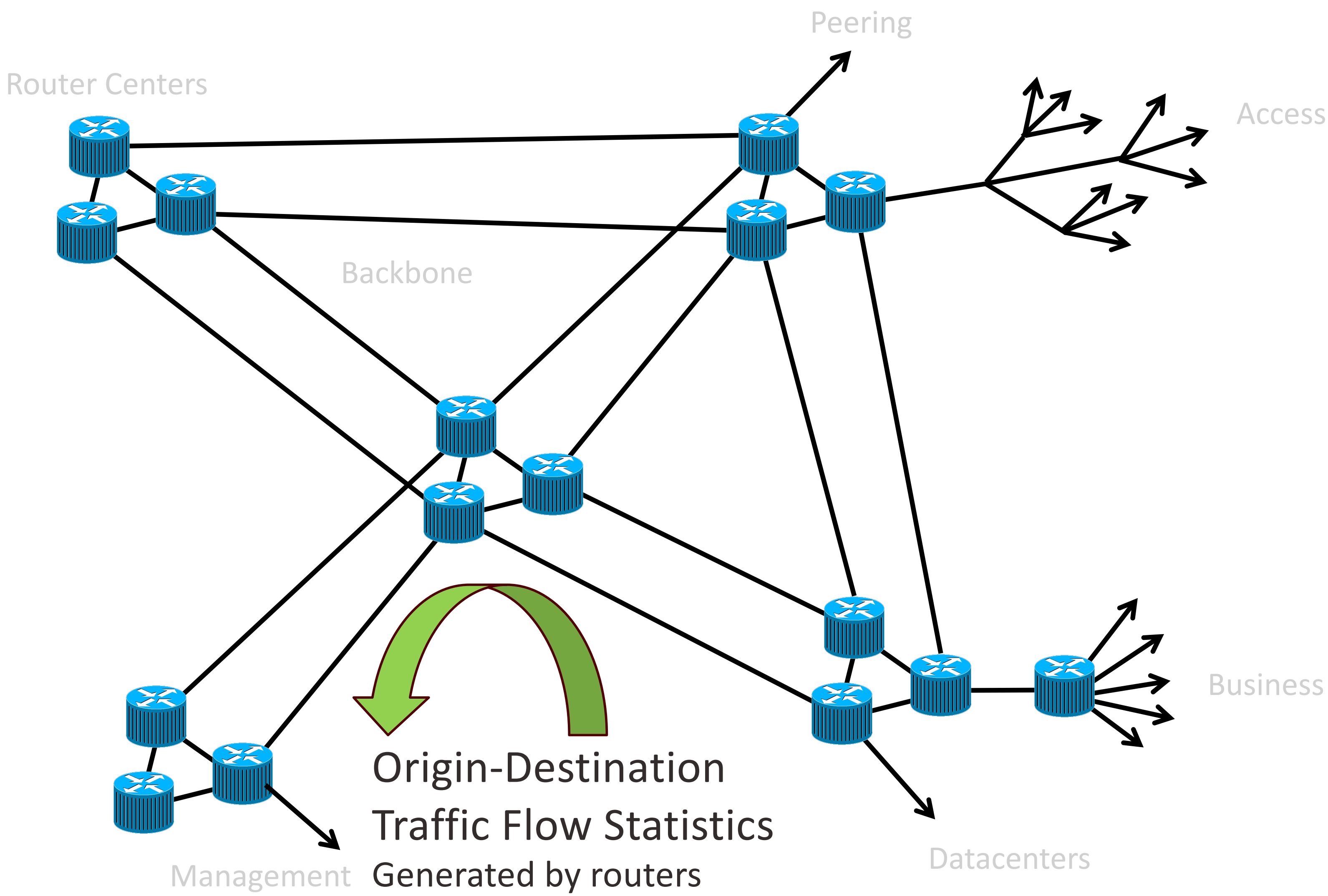
Data Characteristics
(Heavy Tails, Correlations)

Sampling

Resource Constraints
(Bandwidth, Storage, CPU, Cloud Cost)

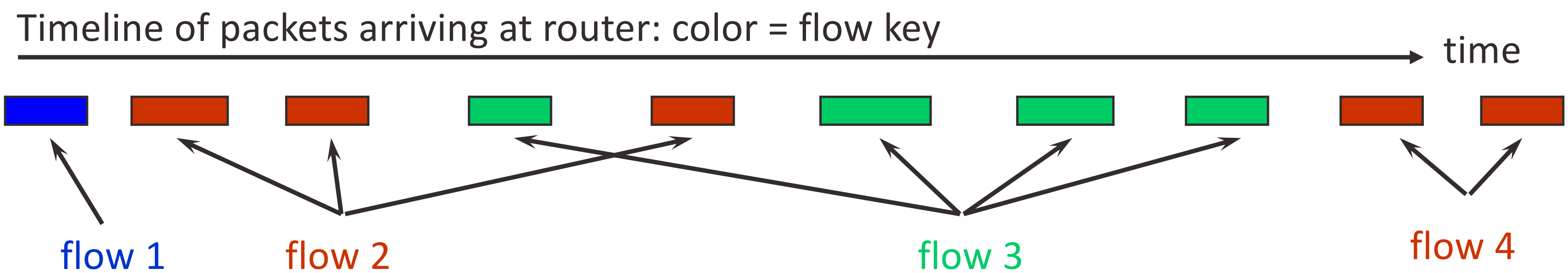
Query Requirements
(Ad Hoc, Accuracy, Speed)

ISP Data: Traffic Flow Statistics



Origin-Destination
Traffic Flow Statistics
Generated by routers

Flow Records



- IP Flow:
  - Set of packets with common flow **key** observed close in time
- Flow Key:
  - Origin/Destination IP addresses, TCP/UDP ports of packets in the flow,
- Flow Records:
  - Summaries of flows (key, #packet, #byte, first/last packet time, …)
  - Continuously compiled by routers, exported to collector
- 10’s PetaBytes daily network traffic ➔ 100’s TeraBytes flow records
  - Applications
    - Routine: compute time series of aggregates over pre-defined selectors
    - Challenge: real-time detection of botnet victim acquisition, communications, attacks

Abstraction: Keyed Data Streams

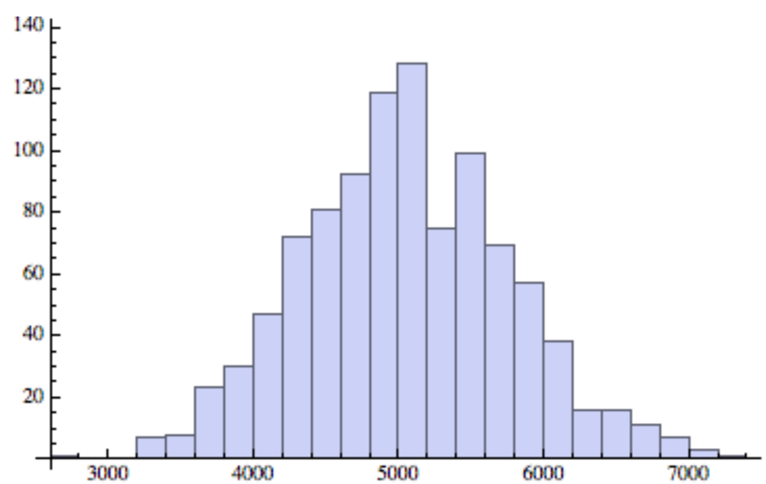
- Data Model: items are keyed weights
  - Item (x,k): Weight x; key k
    - x = flow bytes, k = flow key (common endpoints of packets)
- Stream of keyed weights
  - $\{(x_i , k_i): i = 1,2,\ldots,n\}$)
- Generic query: subset sums
  - $X(S) = \Sigma_{i \in S} x_i \quad S \subset \{1,2,\ldots,n\}$ i.e. total weight of index subset S
  - Typically $S = S(K) = \{i: k_i \in K\}$: items with keys in K
    - X(S(K)) = e.g. total bytes to given IP dst address / port
- Aim:
  - Compute fixed size summary of stream that can be used to estimate arbitrary subset sums with known error bounds

Inclusion Sampling and Estimation

- Horvitz-Thompson (1952)
  - Item i of size x_i is sampled with probability p_i
  - Estimate $x'_i = x_i / p_i$ (if sampled), 0 if not
  - Unbiased: $E[x'_i] = x_i$
- Linearity
  - Estimate of subset sum = sum of corresponding estimates
  - Subset sum $X(S)= \Sigma_{i\in S} x_i$ has estimate $X'(S) = \Sigma_{i\in S} x'_i$
    - Query on S: find matching items in sample and sum estimates
- Accuracy
  - Exponential Bounds: $\Pr[\ |X'(S) - X(S)| > \delta X(S)\ ] \leq e^{-g(\delta)X(S)}$
  - Translate into confidence intervals for X(S)

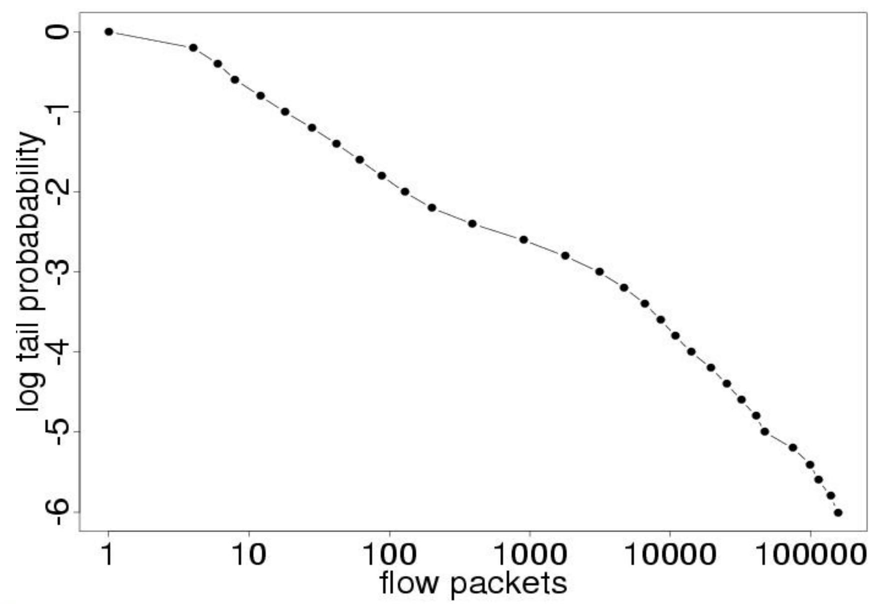
Matching Data to Analysis with Sampling

- Generic problem 1: Counting items: use weight $x_i = 1$
  - Uniform sampling with probability p works fine
  - Estimated subset count X'(S) = #{samples in S} / p
  - Accuracy?
    - relative variance of X'(S) = (1/p -1)/X(S)
    - given p, get any desired accuracy for large enough S



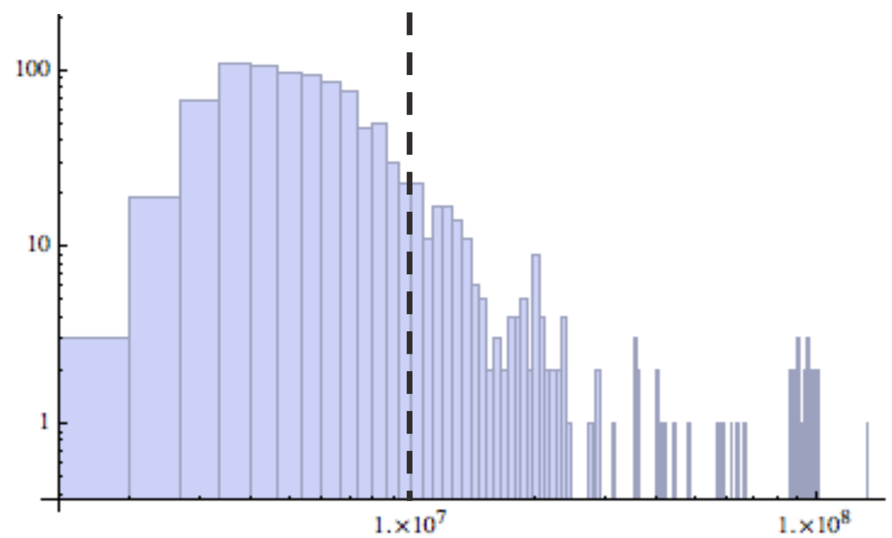
Heavy Tails in the Internet and Beyond

- Heavy tailed distribution
  - E.g. Pareto, $P[X > x] \sim x^{-\alpha}$
  - 80-20 Laws: Small fraction of items have large fraction of weight
- Many examples
  - Degree distributions in web graph, social networks
  - Bytes and packets per network flow
  - Files sizes in storage



Matching Data to Analysis with Sampling

- Generic problem 2: x_i in Pareto distribution
- Uniform sampling?
  - Likely to omit heavy items $\Rightarrow$ big hit on accuracy
  - Making selection set S large doesn't help
- Select m largest items ?
  - biased & smaller items systematically ignored



Sample Cost Optimization

- Independent sampling from n items with weights $\{x_1, \ldots, x_n\}$
- Goal: find the "best" sampling probabilities $\{p_1, \ldots, p_n\}$
- Horvitz-Thompson: unbiased estimate of each x_i by

$$x'_i = \begin{cases} x_i / p_i & \text{if weight } i \text{ selected} \\ 0 & \text{otherwise} \end{cases}$$

- Two costs
  1. Sampling variance from Horvitz-Thompson : $\mathrm{Var}(x'_i) = x^2_i\,(1/p_i - 1)$
  2. Expected Sample Size: $\Sigma_i p_i$
- Minimize Linear Combination Cost: $\Sigma_i\,(x_i^2(1/p_i - 1) + z^2\, p_i)$
  - z expresses relative importance of small sample vs. small variance

Minimal Cost Sampling

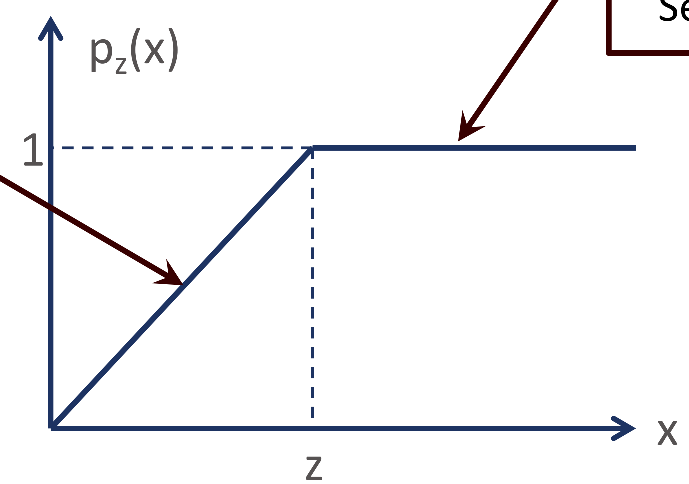
- Minimize
  - Cost $\Sigma_i (x_i^2 (1/p_i - 1) + z^2 p_i)$ subject to $1 \geq p_i \geq 0$
- Solution
  - IPPS: Inclusion probability proportional to size
  - $p_i = p_z(x_i) = \min\{1, x_i / z\}$
  - Call z the "sampling threshold"
  - Unbiased estimator $x_i/p_i = \max\{x_i, z\}$

Large items ($x_i > z$):

Selection Probability = 1

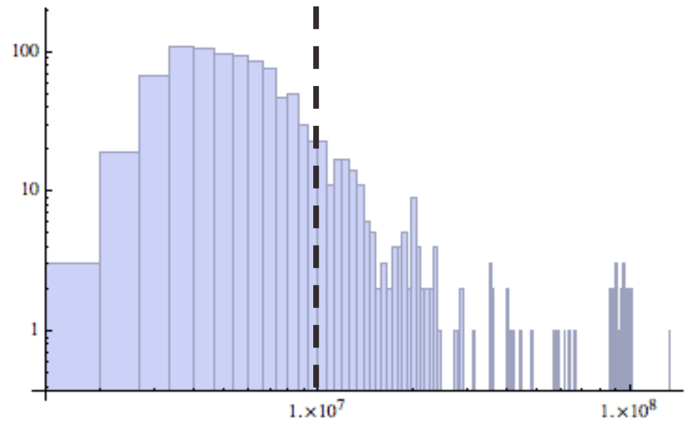
Small items ($x_i \leq z$): PPS

Probability Proportional to Size

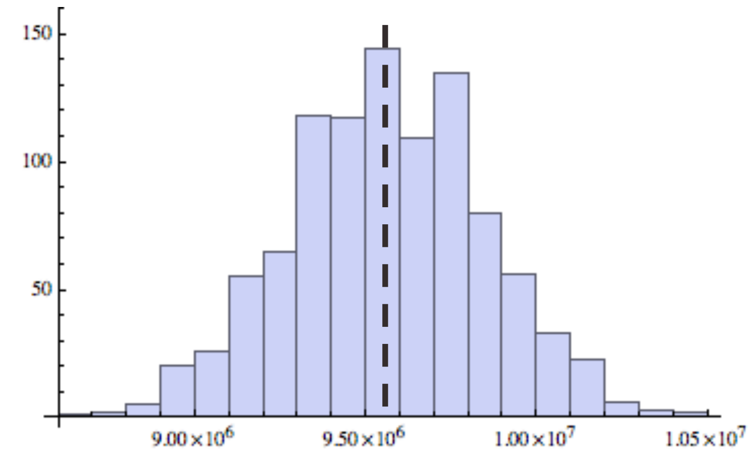


Taming the Heavy Tail

- Distribution of packet count estimates



Uniform sampling



IPPS sampling

Variations on a Theme

- Matching sampling to estimation is versatile approach to sample design
  - Many variations expressing different resource and estimation goals
- Fixed Size Sampling: Reservoir IPPS sampling

  [Cohen, Duffield, Lund, Kaplan, Thorup; SODA 2009, SIAM J. Comput. 2011]
- Structure-Aware Sampling
  - Minimize variance only for Range Queries (e.g. IP prefixes)

  [Cohen, Cormode, Duffield, PVLDB 2011]
- Fair Sampling over subpopulation streams of different rates
  - Minimizing *Relative* variance of subpopulation subset sums

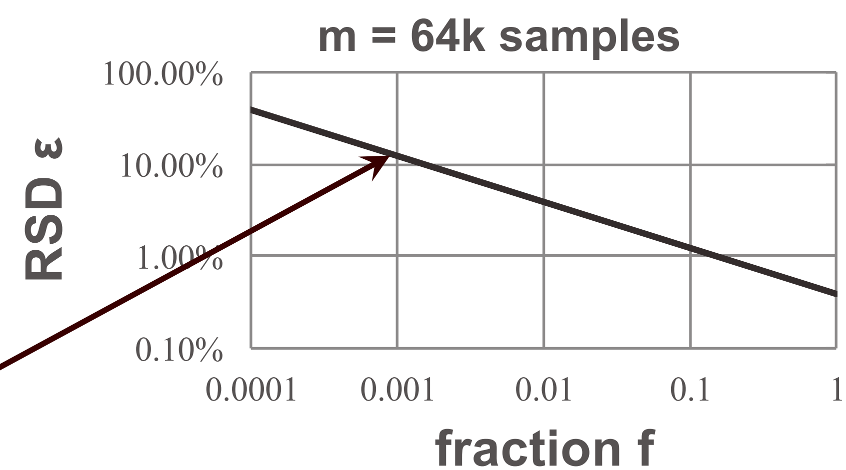
  [Duffield, Sigmetrics 2012]
- Stable Sampling
  - Minimize churn in sample set

  [Cohen, Cormode, Duffield, Lund, TALG 2016]
- IPPS sampling & variations used in ISP measurement today

Estimation Accuracy in Practice

- Aim: estimate heavy hitters
  - any subset sum comprising at least some fraction f of total weight
- Suppose: sample size m
- Analysis: typical estimation error ε (relative standard deviation) obeys

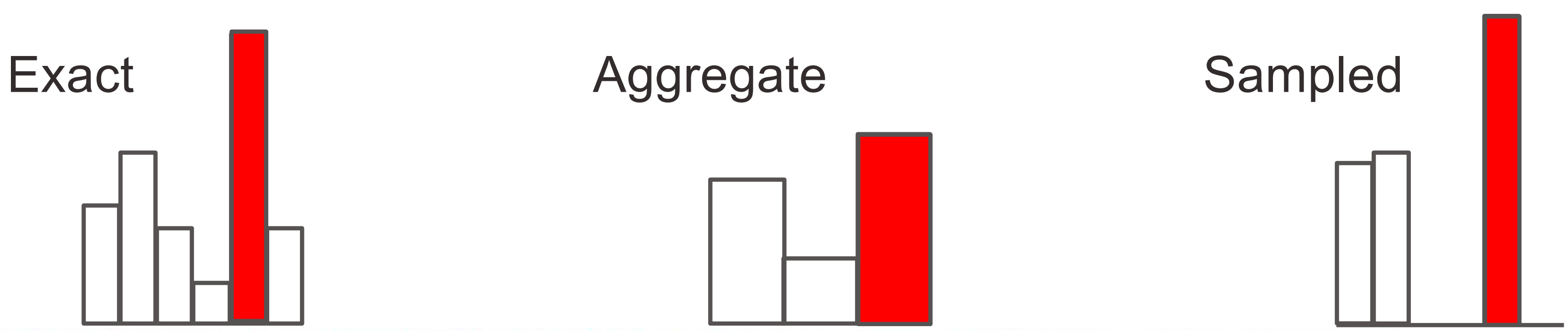
$$\varepsilon < (f\ m)^{-1/2}$$



Estimate fraction f = 0.1% with typical relative error ~ 10%

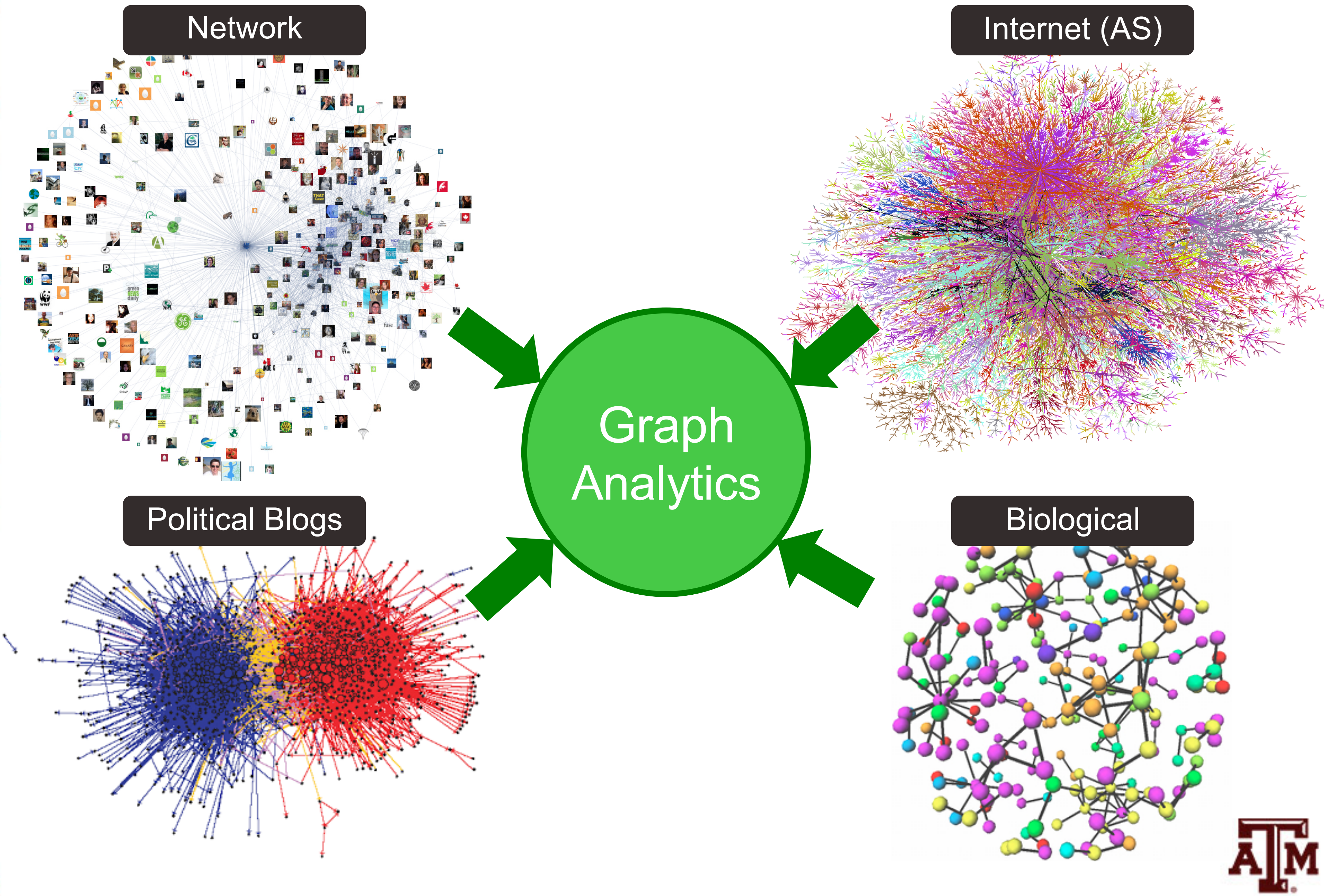
Heavy Hitters: Exact vs. Aggregate vs. Sampled

- Sampling does not tell you where the interesting features are
  - But does speed up the ability to find them with existing tools
- Example: Heavy Hitter Detection
  - Setting: Flow records reporting 10GB/s traffic stream
  - Aim: find Heavy Hitters = IP prefixes comprising ≥ 0.1% of traffic
  - Response time needed: 5 minute
- Compare:
  - Exact: 10GB/s x 5 minutes yields upwards of 300M flow records
  - 64k aggregates over 16 bit prefixes: no deeper drill-down possible
  - Sampled: 64k flow records: **any** aggregate ≥ 0.1% accurate to ~10%



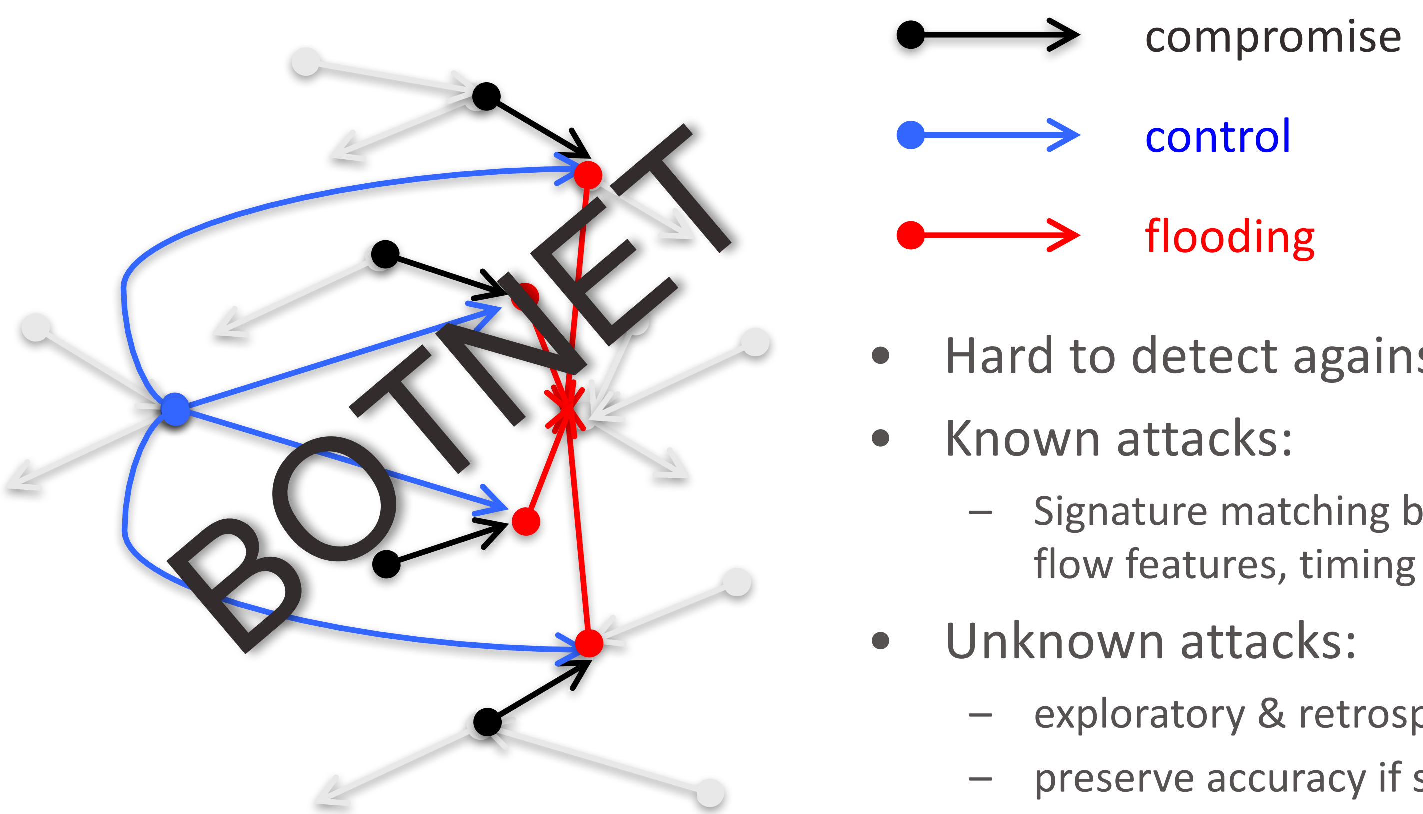
Graphs = Really Big Data

- Operational Graph Data
  - Search providers: web graphs (billions of pages indexed)
  - Online social networks:
    - Facebook: ~10^9 users (nodes), ~10^{12} edges (relationships)
  - ISPs: communications graphs
    - From flow records: node = IP src/dst, edge if traffic flows between them
- Graph Streaming Data
  - Transactional edge data often not maintained in graphical form
    - Real time streams e.g. flow records, or stored transactions e.g. retail purchases
  - Need to support fast, retrospective queries over multilayer graphs
    - IP communications graph, social networks, external resource graphs
  - Sampling needs to be representative over sets of target query objects
    - nodes, links, paths, subgraphs,…


Network
Internet (AS)
Graph Analytics
Political Blogs
Biological

Example: Streaming ISP Graphs

- Node = IP address
- Directed edge = flow from source node to destination node



- Hard to detect against background
- Known attacks:
  - Signature matching based on partial graphs, flow features, timing
- Unknown attacks:
  - exploratory & retrospective analysis
  - preserve accuracy if sampling?

Streaming Subgraph Estimation

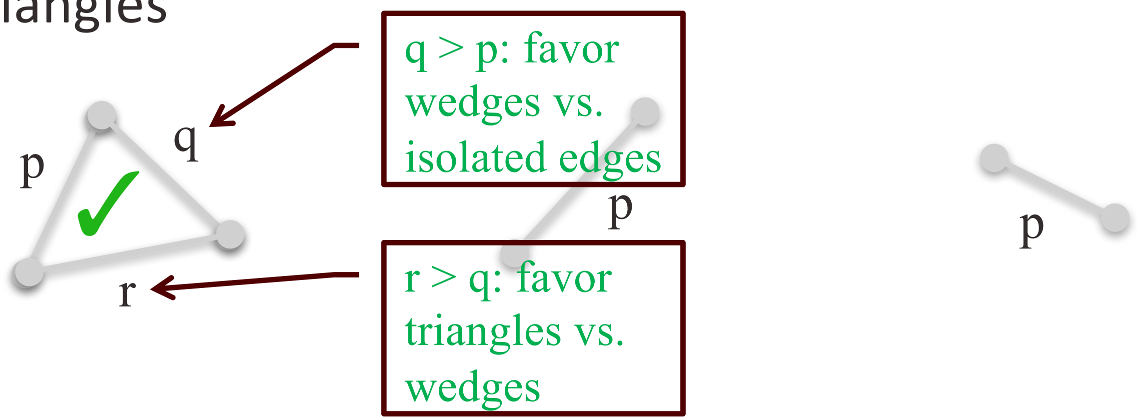
- Hot topic: sample-based subgraph counting from streams
  - Triangles: simplest non-trivial representation of node clustering
    - Regard as prototype for more complex subgraphs of interest
- Uniform sampling performs poorly:
  - Chance for random sampled edges to form subgraph is ≈ 0
  - Non-uniform edge sampling: preferentially select target subgraphs
- Prior work optimizes subgraph specific data structure
  - [Buriol et. al. 06]: sample edges, assumptions on arrival order
  - [Jha et.al. KDD 2103], [Pawan et.al. VLDB 2013]:
  - Focus e.g. on triangles
  - Has the effect of combining sampling and estimation steps

Disjoining Sampling from Estimation

- Sampling:
  - Selection of edges from graph stream
- Estimation:
  - Computation from edge sample
  - Approximate count of subgraph selections from query
  - Can be done at any time during stream
- Don’t need to know query selection when sampling
- Don’t maintain subgraphs in storage during sampling
  - Potential area for resource trade-off
    - Intermediate subgraph storage vs. computation on the fly

Graph Sample and Hold

- General framework for sampled subgraph counting
- Adaptive edge selection: arriving edge i sampled with probability p_i
  - p_i encodes importance to subgraph queries in current sampled topology
- Example: triangles



- Unbiased Subgraph Count Estimation
  - Subgraph J sampled ⇔ All edges $\{j \in J\}$ sampled
  - Horvitz-Thompson Estimator $1/p_J = \prod_{j \in J} 1/p_j$ for sampled subgraphs
    - Triangle: $(pqr)^{-1}$ Wedge: $(pq)^{-1}$ Edge: p^{-1}

Framework for Adaptive Edge Selection

- L = stream of edges {1, 2 , 3, }
  - L_n = first n edges
  - L'_n = edges sampled from L_n

- Adaptive edge selection
  - Conditional sampling probabilities
    - p_n = Pr[sample egde n | L'_{n-1}]

Framework for Subgraph Estimation

- Edge sampling indicator $I(n \text{ in } L'_n)$
  - 1 if n is sampled, 0 if not

- Single edge counter
  - $S_n = I(n \text{ in } L'_n) / p_n$

- Unbiasedness
  - $E[S_n \mid L_{n-1}] = 1$ hence $E[S_n] = 1$

- Subgraph counter $S_J = \prod_{j \in J} S_j$ for $J \ni j_1 < \ldots < j_m$

- Unbiased by chaining conditional expectations
  - $E[S_{j1} \ldots S_{jm} \mid L_{jm-1}] = S_{j1} \ldots S_{j(m-1)}$

Comparison with Previous Work

- Comparison to Streaming-Triangles [Jha et. al-KDD'13]
  - Metric: relative error on triangle count

graph	Jha *et al.* $\frac{\lvert \widehat{N}_T - N_T \rvert}{N_T}$	Jha *et al.* $SSize$	gSH $\frac{\lvert \widehat{N}_T - N_T \rvert}{N_T}$	gSH $SSize$	Sample size
web-Stanford	≈ 0.07	40K	0.0023	14.8K	5% of graph edges
web-Google	≈ 0.04	40K	0.0029	25.2K	0.6% of graph edges
web-BerkStan	≈ 0.12	40K	0.0063	39.8K	0.57% of graph edges

92% - 96% Improvement in relative error in same storage

[Ahmed, Duffield, Kompella, Neville, SIGKDD 2014]

Estimation Variance

- Horvitz-Thompson formalism provides unbiased estimates of (co)-variance of subgraph counts
- $\mathrm{Cov}(S'_J , S'_K)$ has unbiased estimator

  $C'(J,K) = S'_{J\backslash K} (S'_{J\cap K} - 1) S'_{K\backslash J}$
  - Computable from sampled subgraphs
- Can approximate variance of rational combinations of counts using the delta-method
  - Global Clustering Coefficient = $3 N_T / N_\Lambda$
    - N_T = #{triangles}, N_Λ = #{paths of length 2}

$$\mathrm{Var}(\widehat{N}_T/\widehat{N}_\Lambda) \approx \frac{\mathrm{Var}(\widehat{N}_T)}{\widehat{N}_\Lambda^2} + \frac{\widehat{N}_T^2\,\mathrm{Var}(\widehat{N}_\Lambda)}{\widehat{N}_\Lambda^4} - 2\frac{\widehat{N}_T\,\mathrm{Cov}(\widehat{N}_T, \widehat{N}_\Lambda)}{\widehat{N}_\Lambda^3}$$

Actual

Estimated/Actual

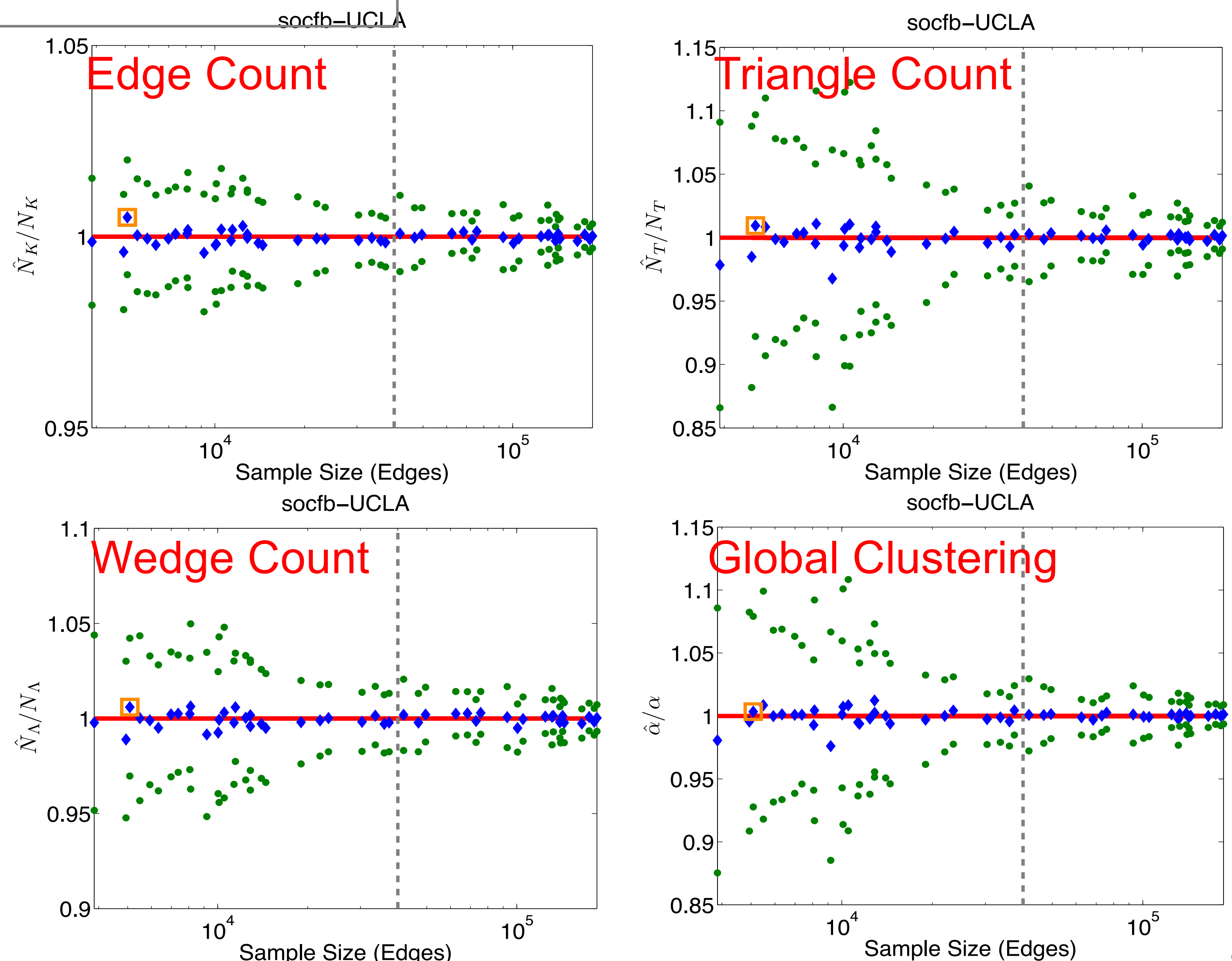
Confidence Upper & Lower Bounds

Sample Size = 40K edges

Dataset:

facebook friendship graph at UCLA

$$\frac{\text{Estimated}}{\text{Actual}}$$



Summary

- Sampling as enabler for Big Data
  - Lowers the bar for resources via cost tradeoffs
    - Size, Speed, Accuracy
  - Selection of reference sample for later subsequent queries
    - Match sampling scheme to query targets
    - Disjoin sampling from estimation
- Graph streams
  - Really big data!
    - Real-time streaming or edge transactional data stored non-graphically
  - Selection of reference sample for later subgraph queries
    - Match sampling scheme to query subgraph targets
    - Adapt edge sampling probabilities to role in target subgraphs
  - Improves trade-off between space and accuracy