# Communication-Efficient Online Detection of Network-Wide Anomalies

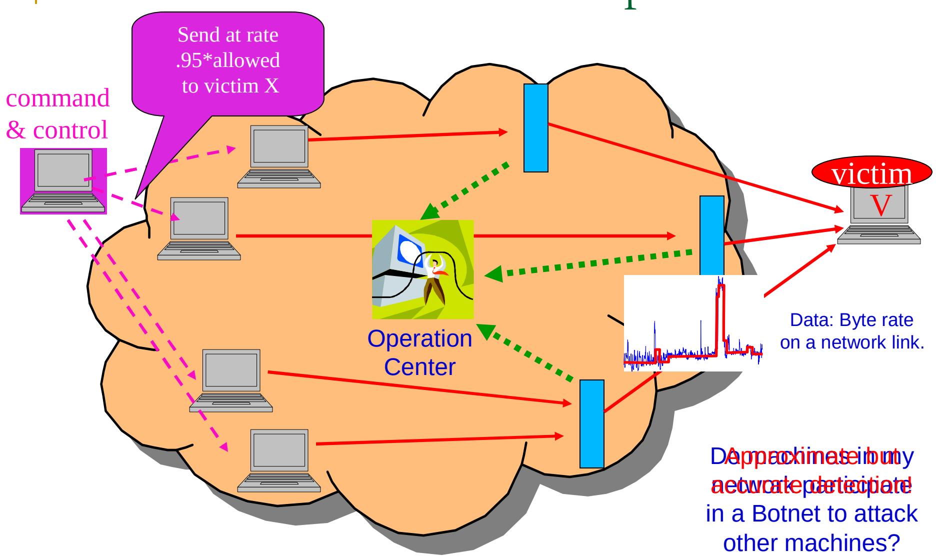
**Ling Huang***    XuanLong Nguyen*
Minos Garofalakis§    Joe Hellerstein*
Michael Jordan*    Anthony Joseph*    Nina Taft§
*UC Berkeley    §Intel Research



## Towards Decentralized Detection

- Today: Distributed Monitoring & Centralized Computation
  - Stream-based data collection
  - *Periodically* evaluate detection function over collected data
  - Doesn't scale well in network size or timescale
- Our contribution: Decentralized Detection
  - *Continuously* evaluate detection function in a decentr. way
  - Low-overhead, rapid response, accurate and scalable
  - Detection accuracy controllable by a "tuning knob"
    - Provable guarantees on detection error (false alarm rate)
    - Flexible tradeoff between overhead and accuracy

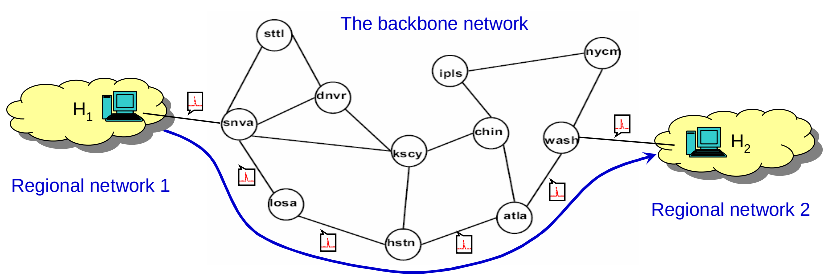
# Detection Problems in Enterprise Network



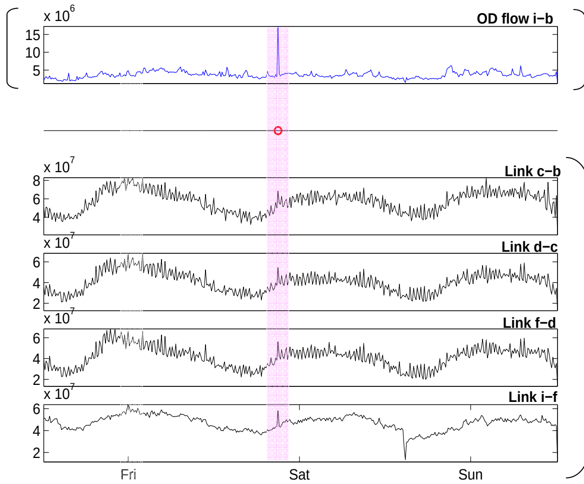

For efficient and scalable detection, push data processing to the edge of network!

# Detection of Network-wide Anomalies

- A ***volume anomaly*** is a sudden change in an Origin-Destination flow (*i.e.,* point to point traffic)
- Given link traffic measurements, ***detect*** the volume anomalies



## An Illustration




Anomalies in (unobserved) flow data

Observed network link data = aggregate of application-level flows

Each link is a dimension

Finding anomalies in high-dimensional, noisy data is difficult !

## The Subspace Method (Lakhina'04)

- An approach to separate normal from anomalous traffic based on Principal Component Analysis (PCA)
- Normal Subspace $\mathcal{S}$: space spanned by the top $k$ principal components
- Anomalous Subspace $\tilde{\mathcal{S}}$: space spanned by the remaining components
- Then, decompose traffic on all links by projecting onto $\mathcal{S}$ and $\tilde{\mathcal{S}}$ to obtain:

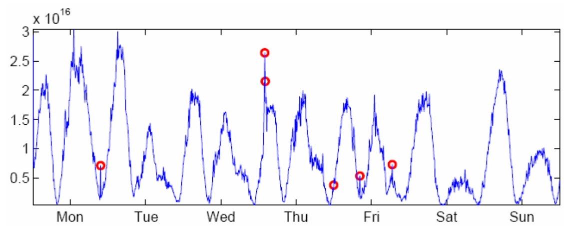
$$\mathbf{y} = \mathbf{y}_{no} + \mathbf{y}_{ab}$$

**Traffic vector of all links at a particular point in time**

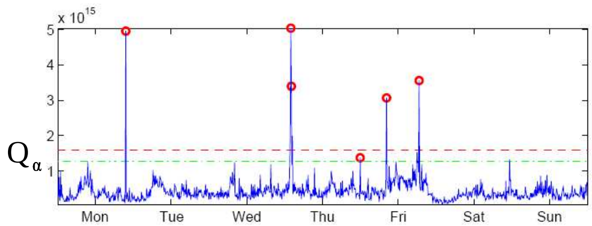
**Normal traffic vector**

**Residual traffic vector**

# Detection Illustration



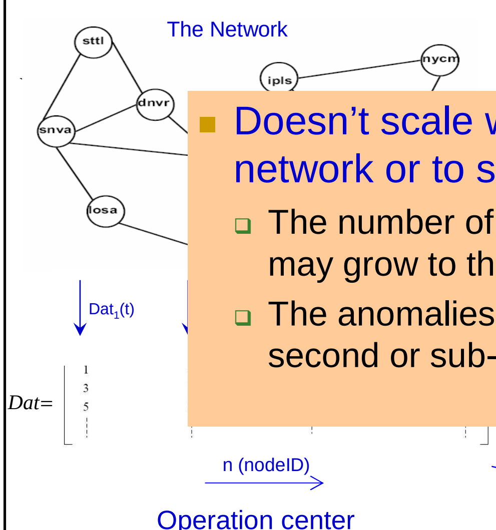

Value of $||\mathbf{y}||^2$
over time
**(all traffic)**




Value of $||\mathbf{C}_{ab}\mathbf{y}||^2$
over time

Red dots: anomalies     Blue curve: traffic data

# The Centralized Algorithm




- Data matrix *Dat*
  1) Each link produces a column of m data over time.
  ... a y at each

... > Q α

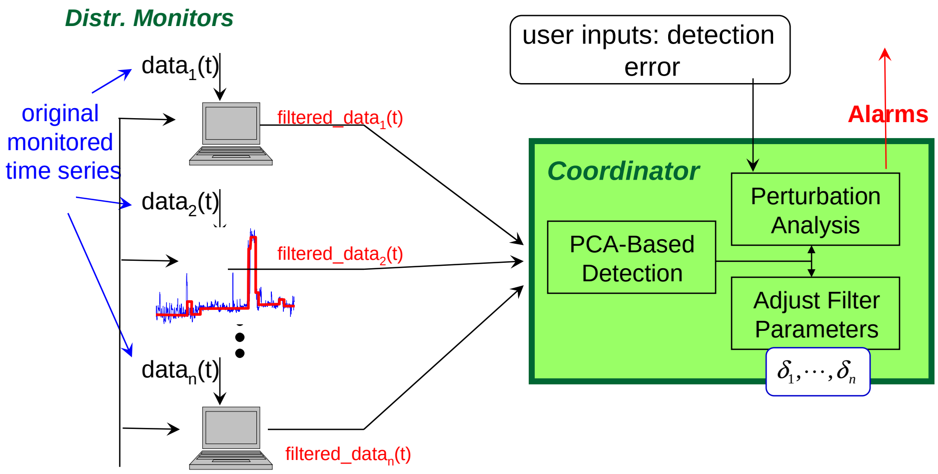
...hreshold
$Q_\alpha$

Eigen vectors     Eigen values

PCA on *Dat*

- Doesn't scale well to large network or to smaller timescales
  - The number of monitoring devices may grow to thousands
  - The anomalies may occur on second or sub-second time scales

# Our In-Network Detection Framework

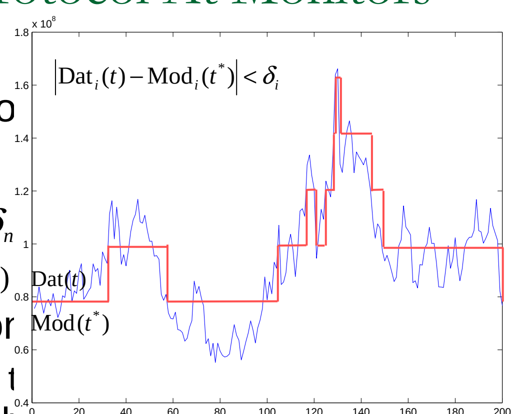



# The Protocol At Monitors

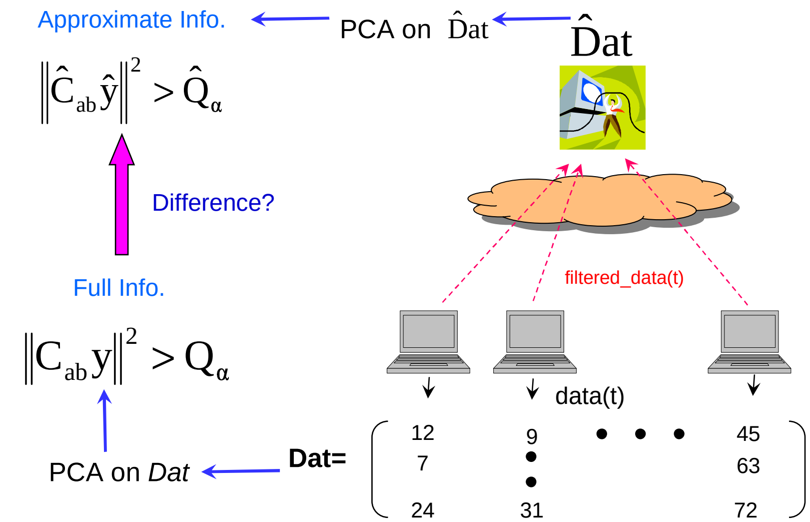
- Monito

  $\delta_1, \cdots, \delta_n$

- $\text{Mod}_i(t^*)$ tion model built or
  - e.g., t s observed locally
  - Simple but enough to achieve 10x data reduction



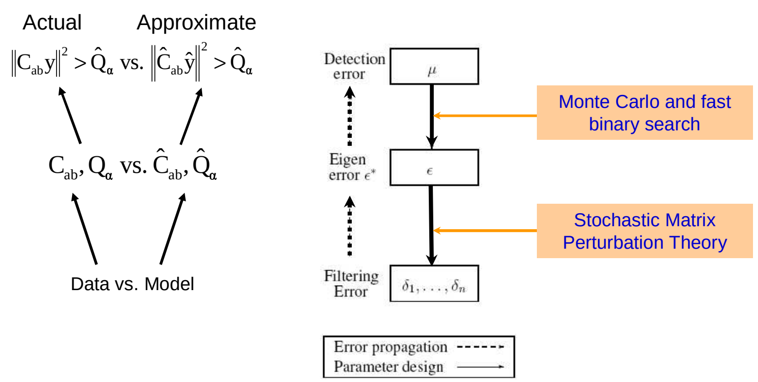
# The Communication and Error Tradeoff




The coordinator computes a set of good $\delta_1, \ldots, \delta_n$ to manage this difference.

# Parameter Design and Error Control (I)

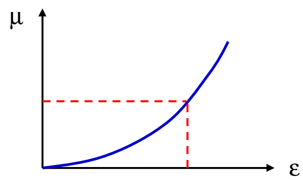
- Users specify an upper bound on false alarm rate, then we determine the filtering parameters δ's




Eigen error: $L_2$ norm of the difference between the approximate eigenvalues and the actual ones

## Parameter Design and Error Control (II)

- Detection Error μ → Eigen-Error ε
  - Mont Carol simulation to find the mapping from ε to μ




  - For the given μ, using fast binary search to find an ε
- Eigen-Error ε → Filtering parameters δ's

$$2\sqrt{\frac{\bar{\lambda}}{m}\cdot\sum_{i=1}^{n}\frac{\delta_i^2}{3}}+\sqrt{\left(\frac{1}{m}+\frac{1}{n}\right)\sum_{i=1}^{n}\frac{\delta_i^4}{9}}=\epsilon$$

## Evaluation

- Given user-specified false alarm rate, evaluate the actual detection accuracy and communication overhead
- Experiment setup
  - Abilene backbone network data
  - Traffic matrices of size 1008 X 41
  - Set uniform slack $\delta_i = \delta$ for all monitors

## Performance

| $\mu$ | Missed Detections | | False Alarms | | Data Reduction | |
|---|---|---|---|---|---|---|
| | Week 1 | Week 2 | Week 1 | Week 2 | Week 1 | Week 2 |
| 0.01 | 0 | 0 | 0 | 0 | 75% | 70% |
| 0.03 | 0 | 1 | 1 | 0 | 82% | 76% |
| 0.06 | 0 | 1 | 0 | 0 | 90% | 79% |

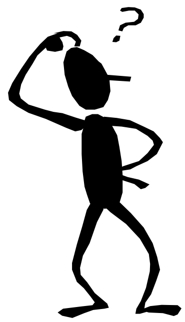
error tolerance = upper bound on error

Data Used: Abilene traffic matrix, 2 weeks, 41 links.

## Summary

- A communication-efficient framework that
  - detects anomalies at desired accuracy level
  - with minimal communication cost
- A distributed protocol for data processing
  - local monitors decide when to update data to coordinator
  - coordinator makes global decision and feedback to monitors
- An algorithmic framework to guide the tradeoff between communication overhead and detection accuracy

## Questions



### *Reference*


[Lakhina'04] *Diagnosing Network-Wide Traffic Anomalies.* A. Lakhina, M. Crovella and C. Diot. In SIGCOMM '04.

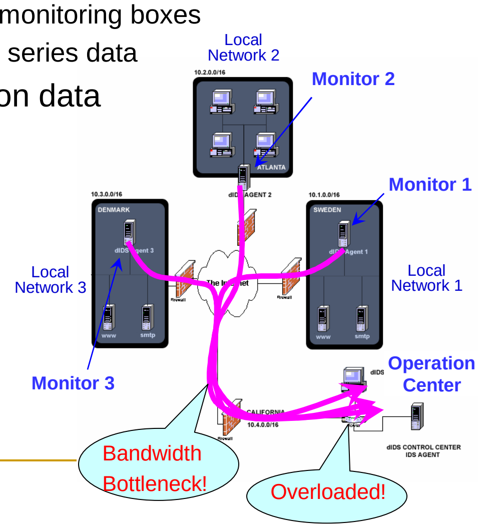
[Huang'06] *In-Network PCA and Anomaly Detection.* L. Huang, X. Nguyen, M. Garofalakis, M. Jordan, A. Joseph and N. Taft. In NIPS 19, 2006.

[Huang'07] *Communication-Efficient Online Detection of Network-Wide Anomalies.* L. Huang, X. Nguyen, M. Garofalakis, J. Hellerstein, M. Jordan, A. Joseph and N. Taft. To appear in INFOCOM'07.


# Backup Slides

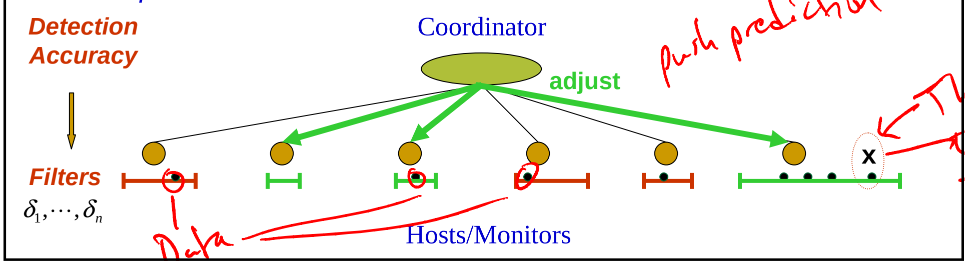
# Traditional Distributed Monitoring

- Large-scale network monitoring and detection systems
  - Distributed and collaborative monitoring boxes
  - Continuously generating time series data
- Existing research focuses on data streaming
  - *Centrally* collect, store and aggregate network state
  - Well suited to answering approximate queries and continuously recording system state
  - Incur high overhead!

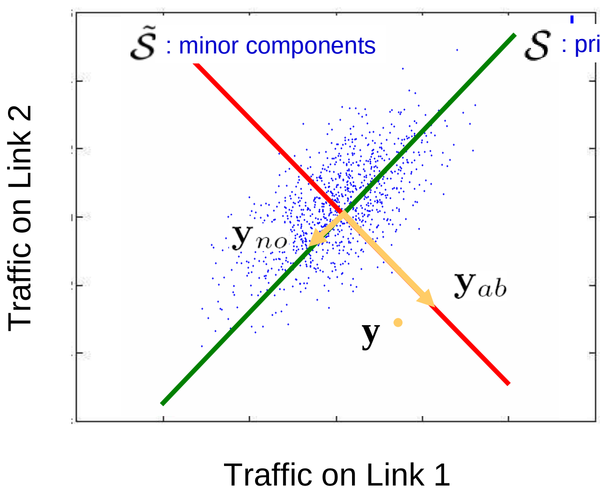



# Our Distributed Processing Approach

- A coordinator
  - Is aggregation, correlation and detection center
- A set of distributed monitors
  - Each produces a time series signals
  - Processes data locally, only sends needed info. to coordinator
  - No communication among monitors
  - *Coordinator tells monitors the level of accuracy for signal updates*



# Principal Component Analysis (PCA)




Principal components are top eigenvectors of covariance matrix. They form the subspace projection matrices $C_{no}$ and $C_{ab}$

$$\mathbf{y}_{no} = \mathbf{C}_{no}\mathbf{y}$$

$$\mathbf{y}_{ab} = \mathbf{C}_{ab}\mathbf{y}$$

Anomalous traffic usually results in a large value of $\mathbf{y}_{ab}$