Experimental Design for Practical Network Diagnosis

Yin Zhang
University of Texas at Austin
yzhang@cs.utexas.edu

Joint work with Han Hee Song and Lili Qiu

MSR EdgeNet Summit
June 2, 2006

Practical Network Diagnosis

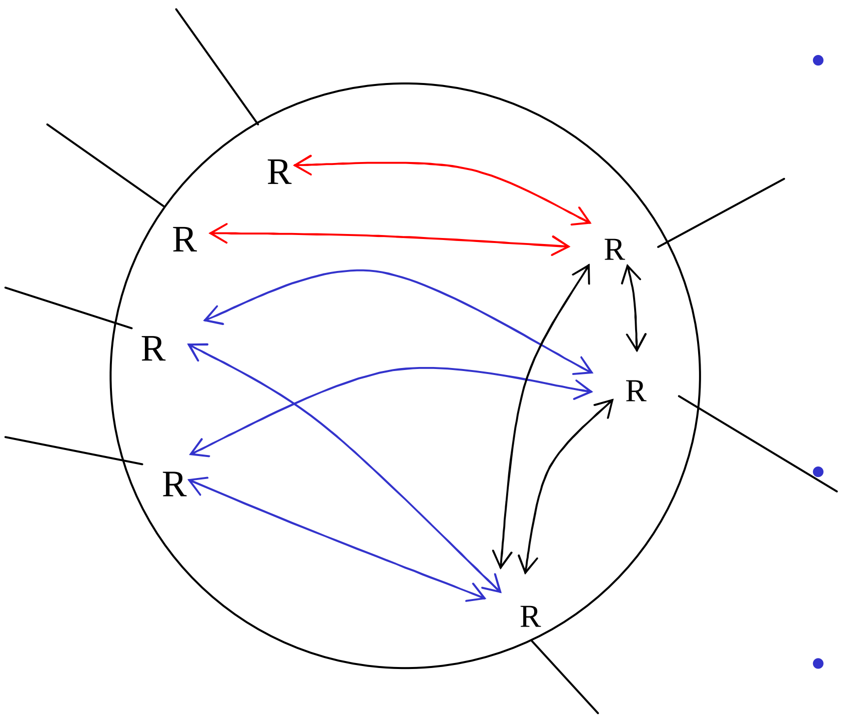
- Ideal
  - Every network element is self-monitoring, self-reporting, self-…, there is no silent failures …
  - Oracle walks through the haystack of data, accurately pinpoints root causes, and suggests response actions
- Reality
  - Finite resources (CPU, BW, human cycles, …)
    → cannot afford to instrument/monitor every element
  - Decentralized, autonomous nature of the Internet
    → infeasible to instrument/monitor every organization
  - Protocol layering minimizes information exposure
    → difficult to obtain complete information at every layer

Practical network diagnosis: Maximize diagnosis accuracy under given resource constraint and information availability

Design of Diagnosis Experiments

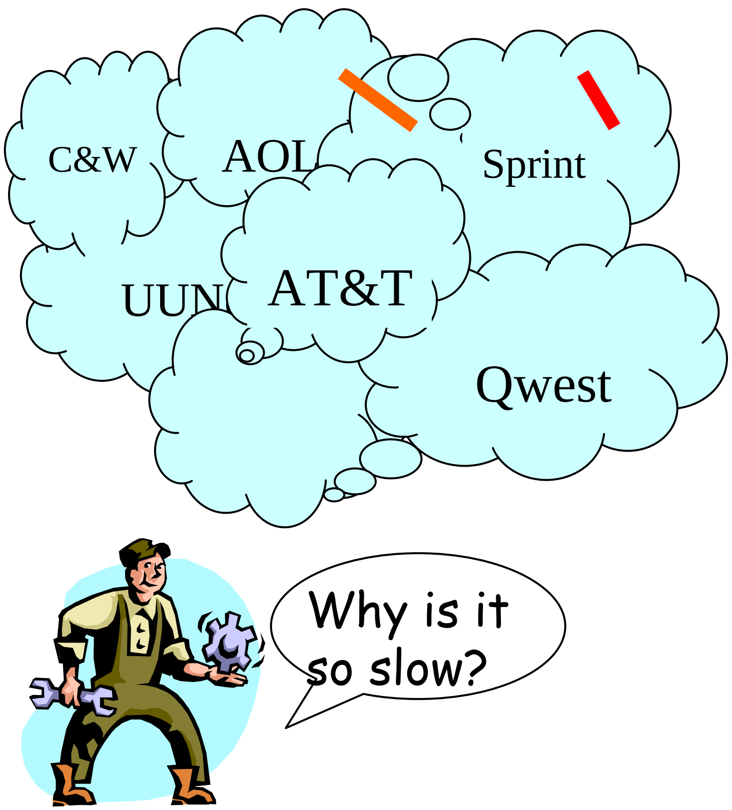
- Input
  - A candidate set of diagnosis experiments
    - Reflects infrastructure constraints
  - Information availability
    - Existing information already available
    - Information provided by each new experiment
  - Resource constraint
    - E.g., number of experiments to conduct (per hour), number of monitors available
- Output: A diagnosis experimental plan
  - A subset of experiments to conduct
  - Configuration of various control parameters
    - E.g., frequency, duration, sampling ratio, …

Example: Network Benchmarking



- 1000s of virtual networks over the same physical network
- Wants to summarize the performance of each virtual net
  - E.g. traffic-weighted average of individual virtual path performance (loss, delay, jitter, …)
  - Similar problem exists for monitoring per-application/customer performance
- Challenge: Cannot afford to monitor all individual virtual paths
  - N^2 explosion times 1000s of virtual nets
- Solution: monitor a subset of virtual paths and infer the rest
- Q: which subset of virtual paths to monitor?

Example: Client-based Diagnosis



- Clients probe each other
- Use tomography/inference to localize trouble spot
  - E.g. links/regions with high loss rate, delay jitter, etc.
- Challenge: Pair-wise probing too expensive due to N^2 explosion
- Solution: monitor a subset of paths and infer the link performance
- Q: which subset of paths to probe?

More Examples

- Wireless sniffer placement
  - Input:
    - A set of locations to place wireless sniffers
      - Not all locations possible – some people hate to be surrounded by sniffers
    - Monitoring quality at each candidate location
      - E.g. probabilities for capturing packets from different APs
    - Expected workload of different APs
    - Locations of existing sniffers
  - Output:
    - K additional locations for placing sniffers
- Cross-layer diagnosis
  - Infer layer-2 properties based on layer-3 performance
  - Which subset of layer-3 paths to probe?

Beyond Networking

- Software debugging
  - Select a given number of tests to maximize the coverage of corner cases
- Car crash test
  - Crash a given number of cars to find a maximal number of defects
- Medicine design
  - Conducting a given number of tests to maximize the chance of finding an effective ingredient
- Many more …

Need Common Solution Framework

- Can we have a framework that solves them all?
  - As opposed to ad hoc solutions for individual problems

- Key requirements:
  - Scalable: work for large networks (e.g. 10000 nodes)
  - Flexible: accommodate different applications
    - Differentiated design
      - Different quantities have different importance, e.g., a subset of paths belong to a major customer
    - Augmented design
      - Conduct additional experiments given existing observations, e.g., after measurement failures
    - Multi-user design
      - Multiple users interested in different parts of network or have different objective functions

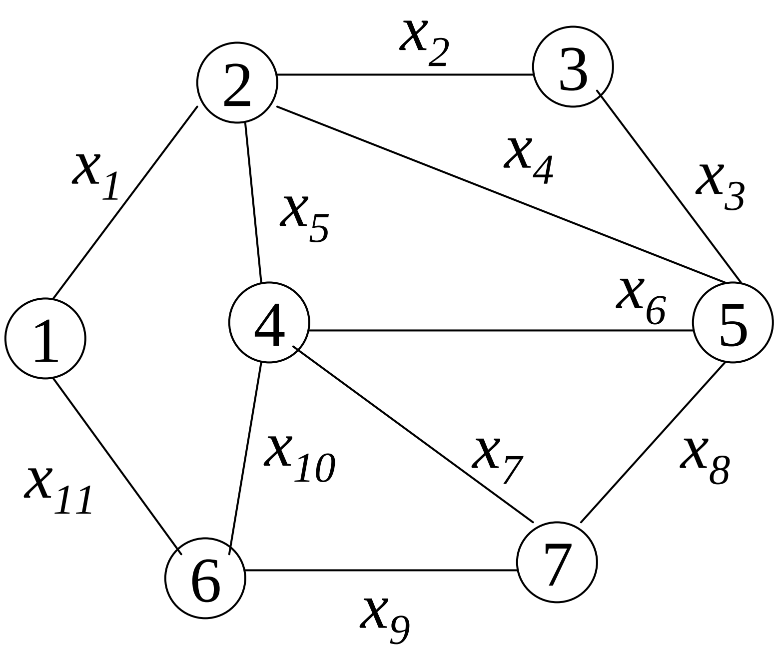
NetQuest

- A baby step towards such a framework
  - "NetQuest: A flexible framework for large-scale network measurement", Han Hee Song, Lili Qiu and Yin Zhang. ACM SIGMETRICS 2006.
- Achieves scalability and flexibility by combining
  - Bayesian experimental design
  - Statistical inference
- Developed in the context of e2e performance monitoring
- Can extend to other network monitoring/ diagnosis problems

What We Want

A function f(x) of link performance x

- We use a linear function f(x)=F*x in this talk



Ex. 1: average link delay
f(x) = (x1+…+x11)/11

Ex. 2: end-to-end delays

$$f(x) = \begin{bmatrix} 1 & 0 & .... & 0 \\ 1 & 1 & 0... & 0 \\ . & . & . & . \\ 0 & ... & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ : \\ x_{11} \end{bmatrix}$$

Apply to any additive metric,
eg. Log (1 – loss rate)

Problem Formulation

What we can measure: e2e performance

Network performance estimation

- Goal: e2e performance on some paths → f(x)
- Design of experiments
  - Select a subset of paths S to probe such that we can estimate f(x) based on the observed performance y_S, A_S, and $y_S=A_Sx$
- Network inference
  - Given e2e performance, infer link performance
  - Infer x based on y=F*x, y, and F

Design of Experiments

- State of the art
  - Probe every path (e.g., RON)
    - Not scalable since # paths grow quadratically with #nodes
  - Rank-based approach [sigcomm04]
    - Let *A* denote routing matrix
    - Monitor rank(*A*) paths that are linearly independent to exactly reconstruct end-to-end path properties
    - Still very expensive

- Select a "best" subset of paths to probe so that we can accurately infer f(x)

- How to quantify goodness of a subset of paths?

Bayesian Experimental Design

- A good design maximizes the expected utility under the optimal inference algorithm
- Different utility functions yield different design criteria
  - Let $D(\eta) = (A_S^T A_s + R)^{-1}$, where $\sigma^2 R^{-1}$ is covariance matrix of x
  - Bayesian A-optimality
    - Goal: minimize the squared error $\| Fx - Fx_S \|_2^2$

$$\phi_A(\eta) = \mathrm{trace}\{FD(\eta)F^T\}$$

  - Bayesian D-optimality
    - Goal: maximize the expected gain in Shannon information

$$\phi_D(\eta) = \det\{FD(\eta)F^T\}$$

Search Algorithm

- Given a design criterion $\phi(\eta)$, next step is to find s rows of *A* to optimize $\phi(\eta)$
  - This problem is NP-hard
  - We use a sequential search algorithm to greedily select the row that results in the largest improvement in $\phi(\eta)$
  - Better search algorithms?

Flexibility

Differentiated design

- Give higher weights to the important rows of matrix F

Augmented design

- Ensure the newly selected paths in conjunction with previously monitored paths maximize the utility

Multi-user design

- New design criteria: a linear combination of different users' design criteria

Network Inference

Goal: find x s.t. Y=Ax

Main challenge: under-constrained problem

L2-norm minimization

$$\min \lambda^2 \| x-\mu \|_2 + \| y-Ax \|_2^2$$

L1-norm minimization

$$\min \lambda \| x-\mu \|_1 + \| y-Ax \|_1$$

Maximum entropy estimation

$$\min \sum_i x_i \log_2 \frac{x_i}{\mu_i} + \| y-Ax \|_2^2$$

Evaluation Methodology

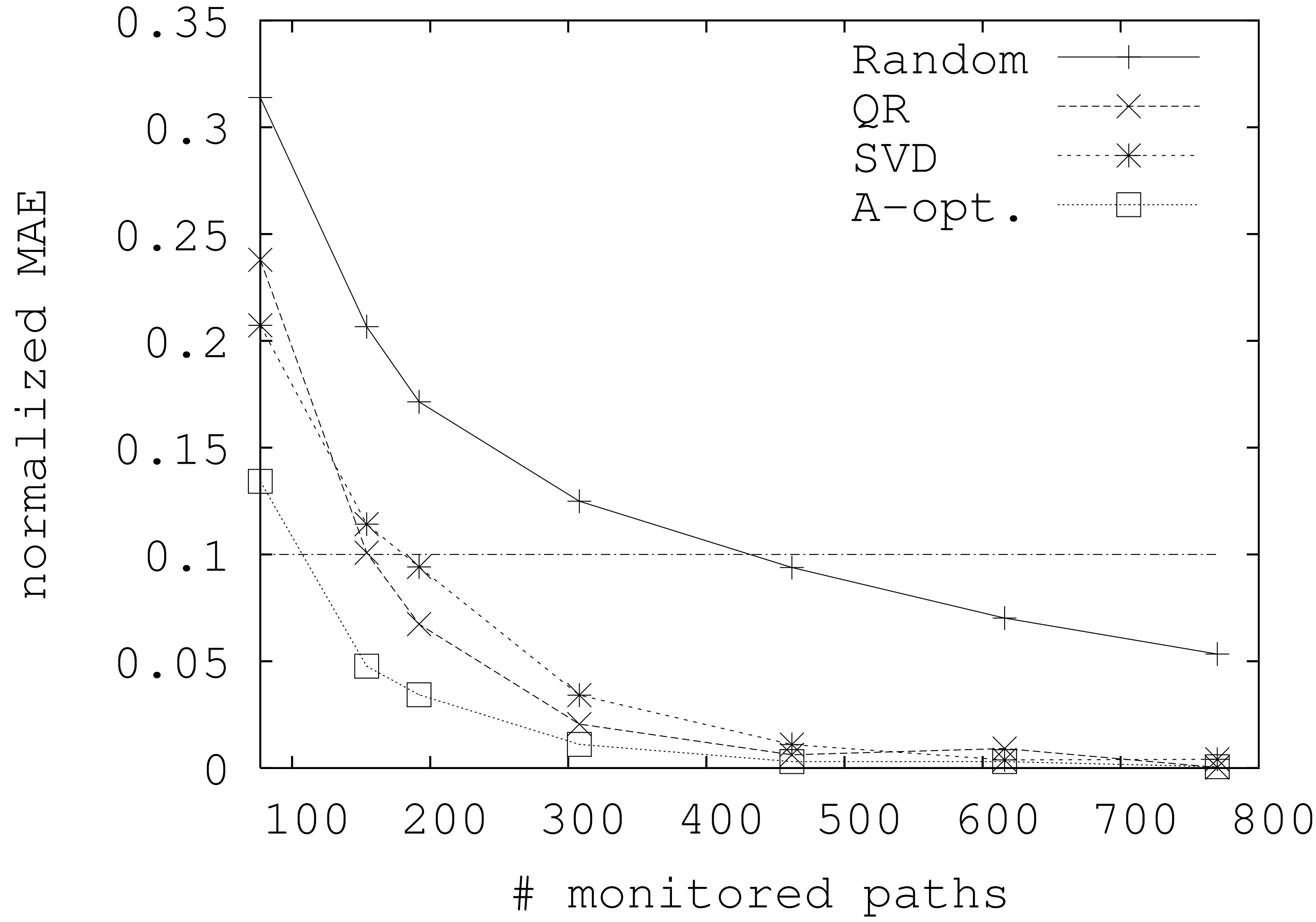
Data sets

	# nodes	# overlay nodes	# paths	# links	Rank
PlanetLab-RTT	2514	61	3657	5467	769
Planetlab-loss	1795	60	3270	4628	690
Brite-n1000-o200	1000	200	39800	2883	2051
Brite-n5000-o600	5000	600	359400	14698	9729

Accuracy metric

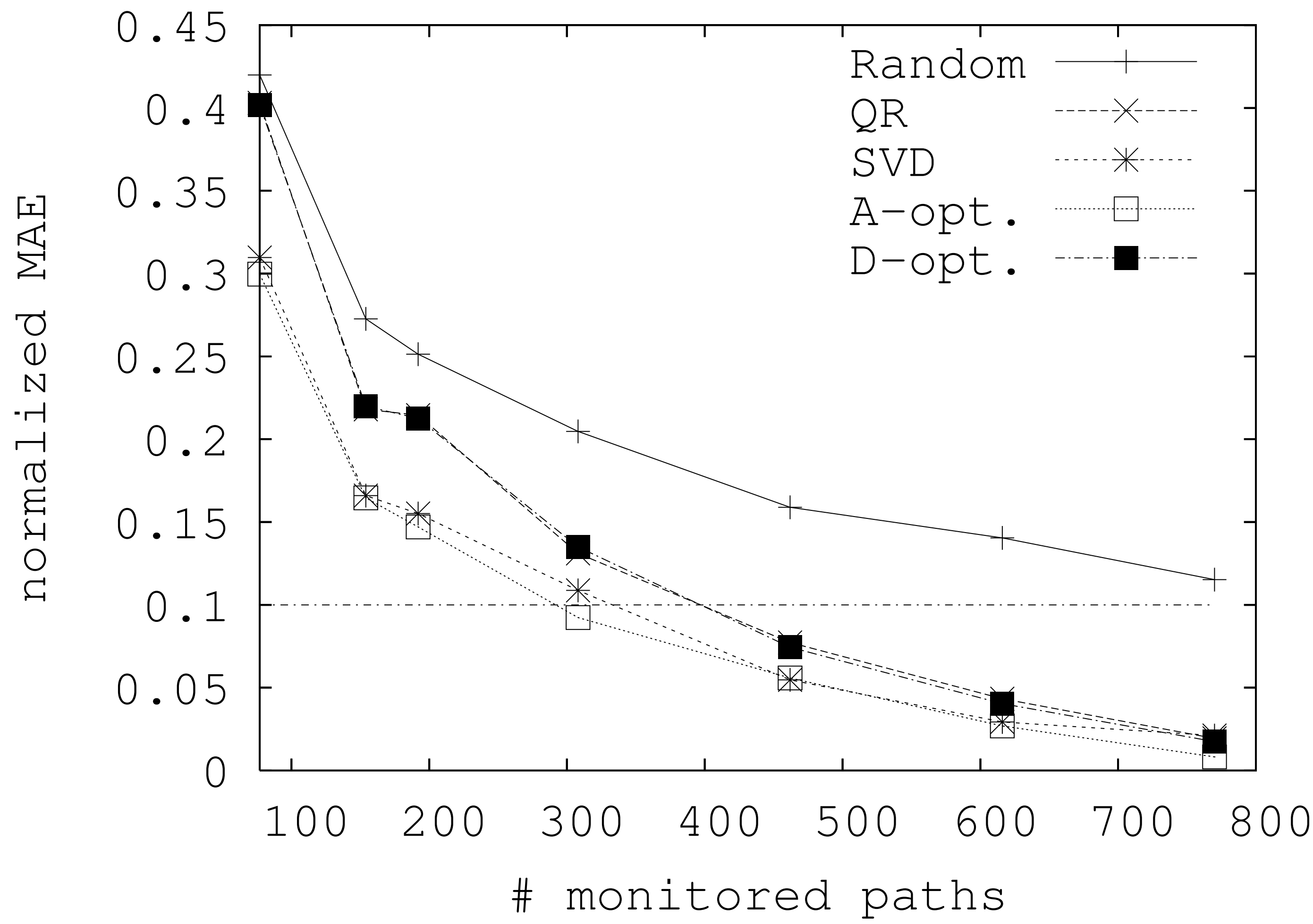
$$normalized\ MAE = \frac{\sum_i |\mathrm{infer}_i - actual_i|}{\sum_i actual_i}$$

Comparison of DOE Algorithms: Estimating Network-Wide Mean RTT



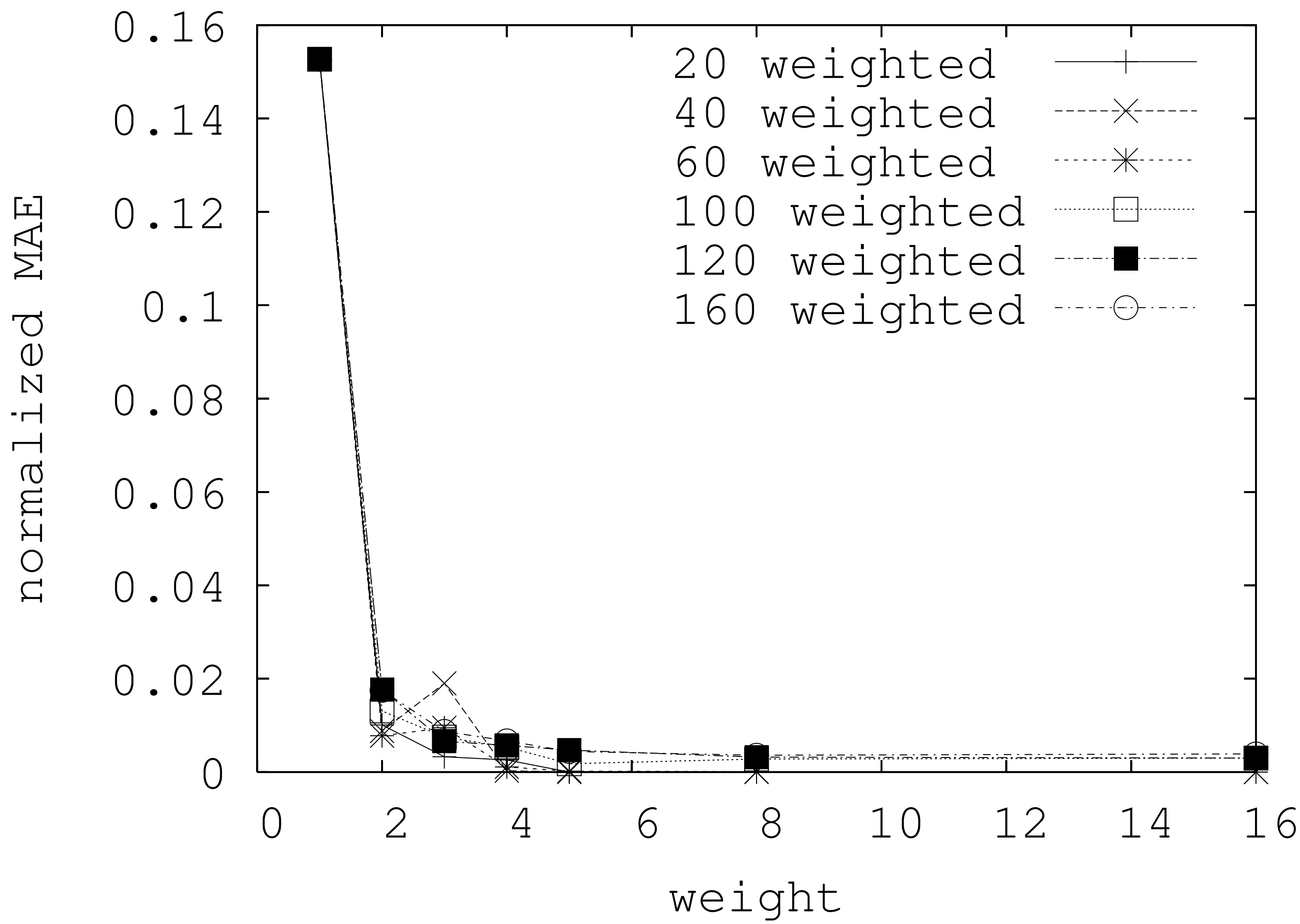
A-optimal yields the lowest error.

Comparison of DOE Algorithms: Estimating Per-Path RTT



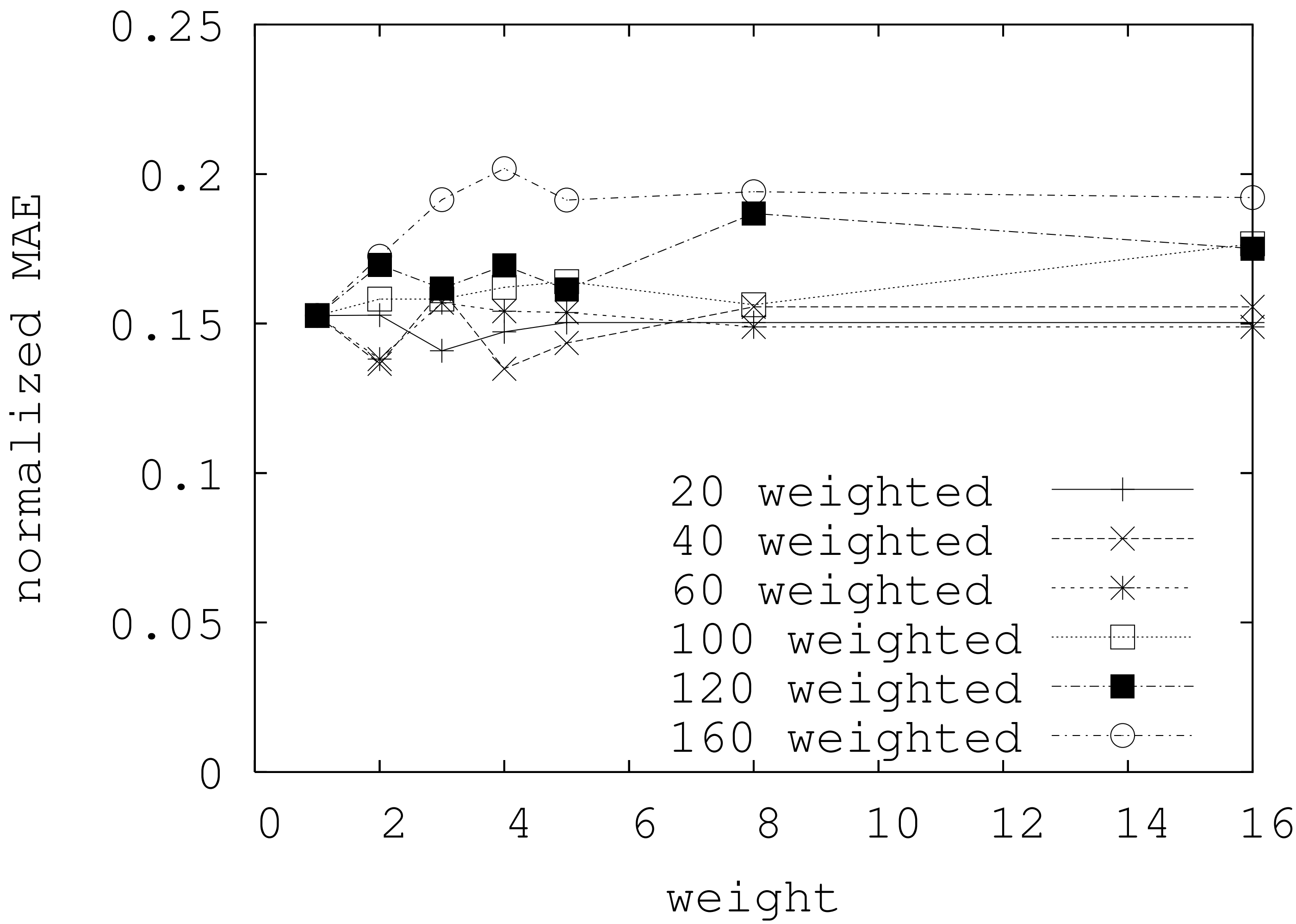
A-optimal yields the lowest error.

Differentiated Design: Inference Error on Preferred Paths



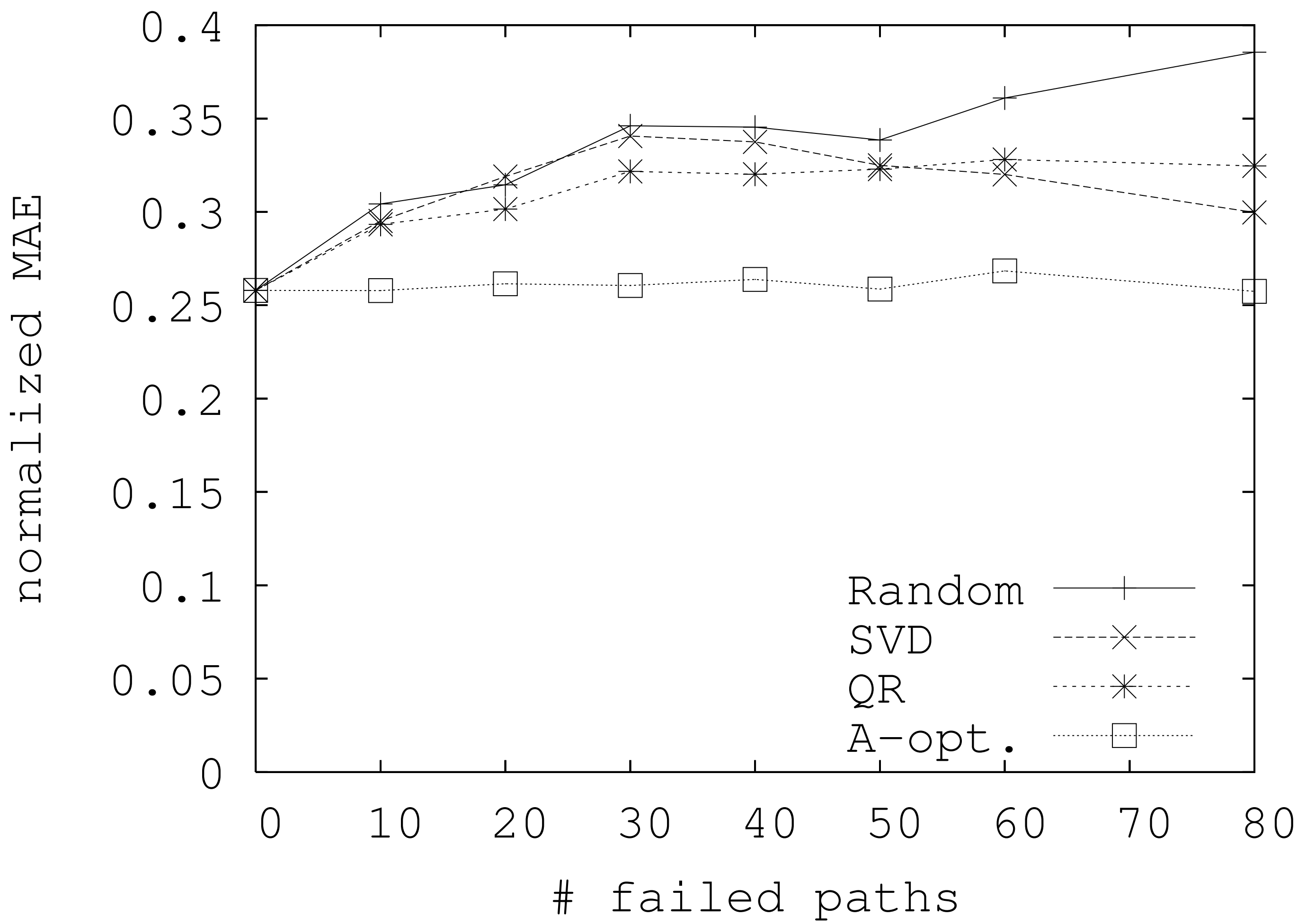
Lower error on the paths with higher weights.

Differentiated Design: Inference Error on the Remaining Paths



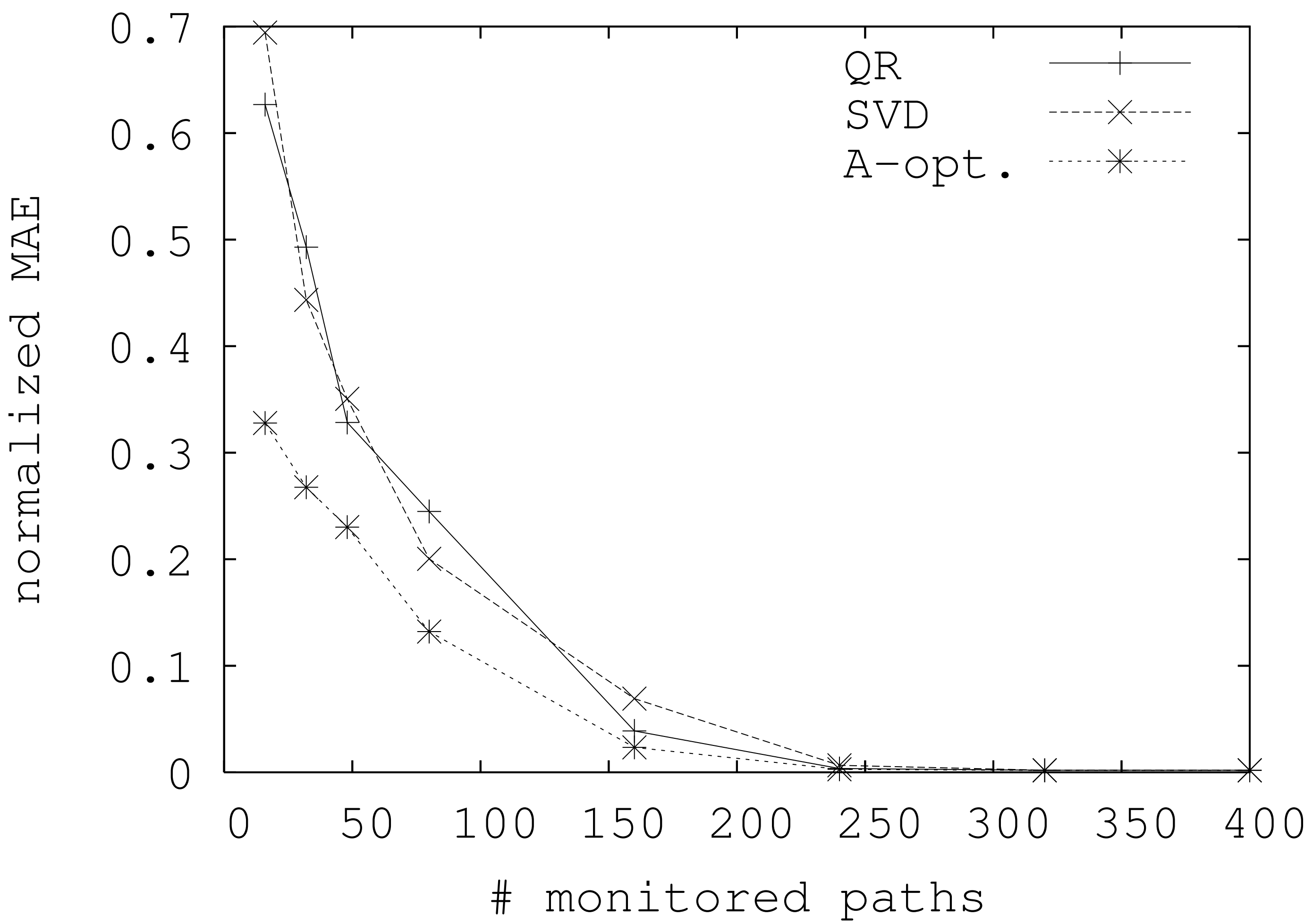
Error on the remaining paths increases slightly.

Augmented Design



A-optimal is most effective in augmenting an existing design.

Multi-user Design



A-optimal yields the lowest error.

Summary

Our contributions

- Bring Bayesian experimental design to network measurement and diagnosis
- Develop a flexible framework to accommodate different design requirements
- Experimentally show its effectiveness

Future work

- Making measurement design fault tolerant
- Applying our technique to other diagnosis problems
- Extend our framework to incorporate additional design constraints

Thank you!