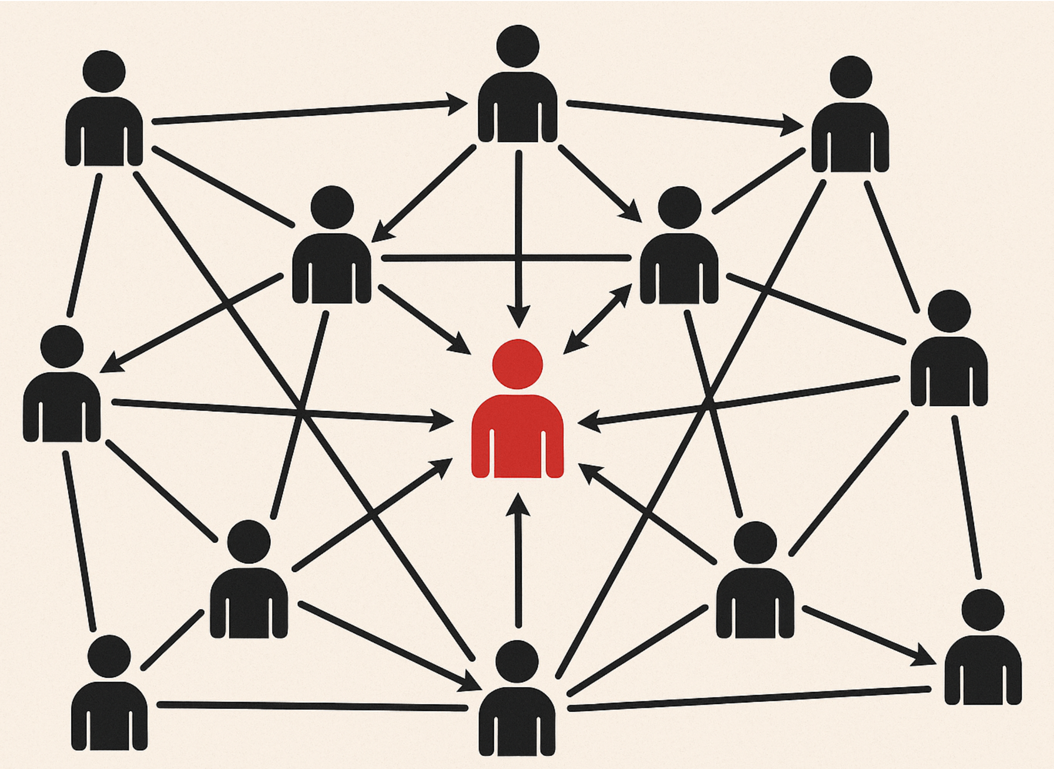
Fast Percolation Centrality Approximation with Importance Sampling

Antonio Cruciani

Leonardo Pellegrina

Aalto University




FAKE NEWS
SHOCKING DISCOVERY!
SECRET PLAN REVEALED
GOVERNMENT COVER-UP

Percolation Centrality

Given a graph $G = (V, E)$ and a percolation states vector $x \in [0,1]^n$ for all nodes v:

$$p(v) = \sum_{s \neq t} \frac{\sigma_{st}(v)}{\sigma_{st}} \cdot \frac{R(x_s - x_t)}{\sum_{u \neq v \neq w} R(x_u - x_w)} \in [0,1]$$

- $\sigma_{st}(v)$ is the number of shortest paths between s and t passing through v
- σ_{st} overall number of shortest paths between s and t
- $R(x) = \max(0, x)$

Percolation Centrality

Given a graph $G = (V, E)$ and a percolation states vector $x \in [0, 1]^n$ for all nodes v:

$$p(v) = \sum_{s \neq t} \frac{\sigma_{st}(v)}{\sigma_{st}} \cdot \kappa(s, t, v) \in [0, 1]$$

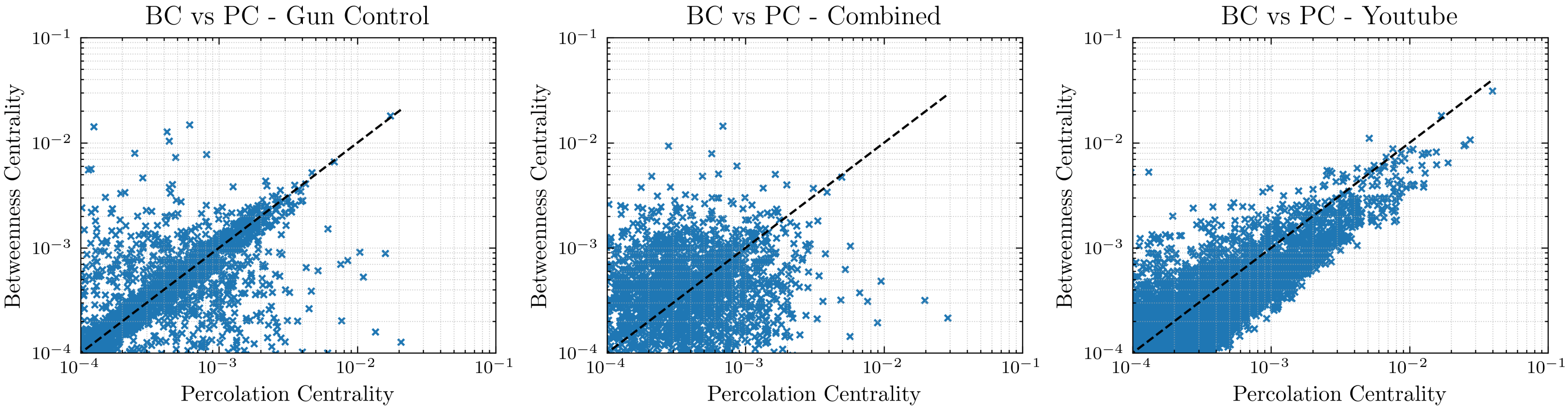
- $\sigma_{st}(v)$ is the number of shortest paths between s and t passing through v

- σ_{st} overall number of shortest paths between s and t

- $R(x) = \max(0, x)$

Why Percolation Centrality ?

Graph	$\|V\|$	$\|E\|$	$\mathcal{L}_{avg}$	$\mathcal{L}$	ρ	Type
Guns	632659	5741968	0.347	$\{0, 1\}$	2.859	U
Combined	677753	6134836	0.246	$\{0, 1\}$	3.053	U
Youtube	152582	6268398	0.310	$[0, 1]$	2.563	D



	Jaccard Similarity Top-K		
Graph	**10**	**50**	**100**
Guns	0.053	0.087	0.117
Combined	0.0	0.031	0.015
Youtube	0.429	0.369	0.504

Efficient Computation

Problem: The exact computation of the Percolation Centrality requires $\mathcal{O}(n \cdot m)$ time!

Efficient Computation

Problem: The exact computation of the Percolation Centrality requires $\mathcal{O}(n \cdot m)$ time!

Idea: Let's compute a high-quality approximation using random sampling. ☺

Goal: Given the accuracy parameter $\varepsilon \in (0, 1]$ we want :

$$|\tilde{p}(v) - p(v)| \leq \varepsilon, \quad \forall v \in V$$

Previous Works

- [de Lima et al,KDD'20] Estimating the Percolation Centrality of Large Networks through Pseudo-dimension Theory.

General Idea:

1) Pick two random nodes $s \neq t$ **uniformly at random**

2) Sample a shortest path between s and t uniformly at random

3) Update the score of each internal node v by $\kappa(s,t,v)$

Their results in a nutshell

They use uniform sampling **(UNIF)** to approximate

$$p^\star(v) = \frac{1}{n(n-1)} \sum_{s \neq t} \frac{\sigma_{st}(v)}{\sigma_{st}} \cdot \kappa(s,t,v)$$

Sample size of

$$\ell = \tfrac{0.5}{\varepsilon^2} \left(\lfloor \log(D) - 2 \rfloor + 1 - \ln \delta \right)$$

To achieve ε-approximation with probability $\geq 1 - \delta$

Their results in a nutshell

They use uniform sampling **(UNIF)** to approximate

$$\boxed{p^\star(v) = \frac{1}{n(n-1)} \sum_{s \neq t} \frac{\sigma_{st}(v)}{\sigma_{st}} \cdot \kappa(s,t,v)}$$

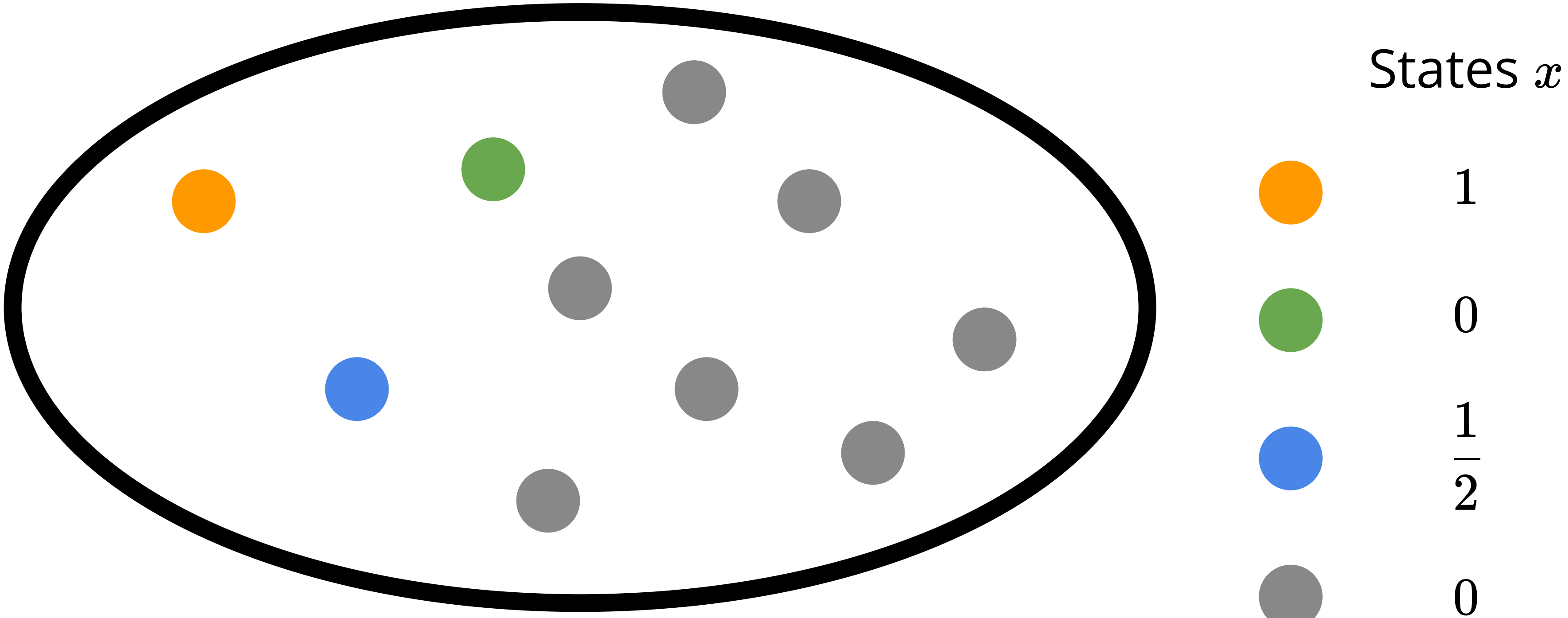
$\varepsilon \geq \frac{1}{n(n-1)}$ $\longrightarrow$ Is uninformative!

Better to directly set $\tilde{p}^\star(v) = 0,\ \forall v$

$\varepsilon < \frac{1}{n(n-1)}$ $\longrightarrow$ We need $\ell \in \Omega(n^4)$ samples!

Some Problems with UNIF

If $x_s \leq x_t$ then $\kappa(s,t,v) = 0$



Importance Sampling in a Nutshell

We want to approximate an expectation

$$\mu = \mathbb{E}_p\left[f(X)\right] = \sum_x f(x)p(x)$$

Problem: Sampling from p may be inefficient

Idea: Sample from a proposal distribution q which emphasizes "important" regions.

$$\mathbb{E}_p\left[f(X)\right] = \mathbb{E}_q\left[f(X)\frac{p(X)}{q(X)}\right]$$

The (maximum) Likelihood ratio

$$\hat{d} = \max_{x:q(x)>0} \frac{p(x)}{q(x)}$$

Small ($\approx$ 1): balanced weights $\longrightarrow$ Low variance, stable estimator

Large: extreme weights $\longrightarrow$ High variance, unstable estimator

Our Importance Sampling Distribution (PercIS)

$$\tilde{\kappa} : V \times V \to [0, 1] \qquad \tilde{\kappa}(s,t) = \frac{R(x_s - x_t)}{\sum_{u \neq w} R(x_u - x_w)}$$

$\tilde{\kappa}$ is a valid distribution over all couple of nodes

For any shortest path τ_{st} between s and t

$$q(\tau_{st}) = \frac{\tilde{\kappa}(s,t)}{\sigma_{st}}$$

The ImportanceSampler

Idea: Once we sample s we need to sample a t such that $x_s > x_t$

1) Sample s with marginal

$$\Pr(s) = \sum_u \tilde{\kappa}(s,u)$$

2) Sample t with

$$\Pr(t \mid s) = \frac{\tilde{\kappa}(s,t)}{\sum_u \tilde{\kappa}(s,u)}$$

> $\mathcal{O}(\log n)$ time per sample with a $\mathcal{O}(n \log n)$ preprocessing

Sample Complexity Analysis

Sample size of

$$\ell \approx \frac{\hat{d}^2 \left(2\hat{v} + \frac{2}{3}\frac{\varepsilon}{\hat{d}}\right)}{\varepsilon^2} \left(\ln\left(2\hat{\rho}/\hat{v}\right) + \ln\left(1/\delta\right)\right)$$

To achieve ε-approximation with probability $\geq 1 - \delta$

- $\hat{v}$ is an upper bound on the maximum variance of the PC
- $\mathrm{avg\_path\_length} \leq \hat{\rho} \leq \hat{d} \cdot \mathrm{avg\_path\_length}$

PercIS: an ε-approximation algorithm

PercIS in three lines

- Quickly observes the graph
- Estimates $\hat{v}$, $\hat{\rho}$ and computes the upper bound on the sample size ℓ
- Draws ℓ random samples and computes the approximation

$$\boxed{\tilde{p}(v) = \frac{1}{\ell} \sum_{i=1}^{\ell} \frac{\kappa(s,t,v)}{\tilde{\kappa}(s,t)} 1[v \in \mathtt{Int}(\tau_{st}^{i})]}$$

PercIS vs UNIF

$$\Delta = \min_{v} \max_{s \neq v \neq t} (x_s - x_t)$$

On all the tested instances it holds $\Delta \approx 1.0$

When $\Delta \in \Omega(1)$, the likelihood ratio $\hat{d}$ of the importance sampling distribution q is $\hat{d} \in \mathcal{O}(1)$

PercIS vs UNIF

There exists instances with $\Delta \in \Omega(1)$ where the likelihood ratio of the uniform distribution is $\Omega(n)$

There exists instances with $\Delta \in \Omega(1)$ where at least $\Omega(n^2)$ random samples are needed by UNIF, while $\mathcal{O}(n)$ random samples are sufficient for PercIS

Experimental Setting - Datasets

Graph	$\|V\|$	$\|E\|$	D	ρ	Type
P2P-Gnutella31	62586	147892	31	7.199	D
Cit-HepPh	34546	421534	49	5.901	D
Soc-Epinions	75879	508837	16	2.755	D
Soc-Slashdot	82168	870161	13	2.135	D
Web-Notredame	325729	1469679	93	9.265	D
Web-Google	875713	5105039	51	9.713	D
Musae-Facebook	22470	170823	15	2.974	U
Email-Enron	36692	183831	13	2.025	U
CA-AstroPH	18771	198050	14	2.194	U

All algorithms implemented in C++ (using OpenMP), and confidence parameter $\delta = 0.05$

Experimental Setting - Percolation States

RS - Random Seeds

- Pick a fixed number of random nodes with state = 1, all others = 0.
- Models early infection / first spreaders

RSS - Random Seeds Spread

- Pick a $\log n$ seeds with state = 1, simulate diffusion
- Models infection spread

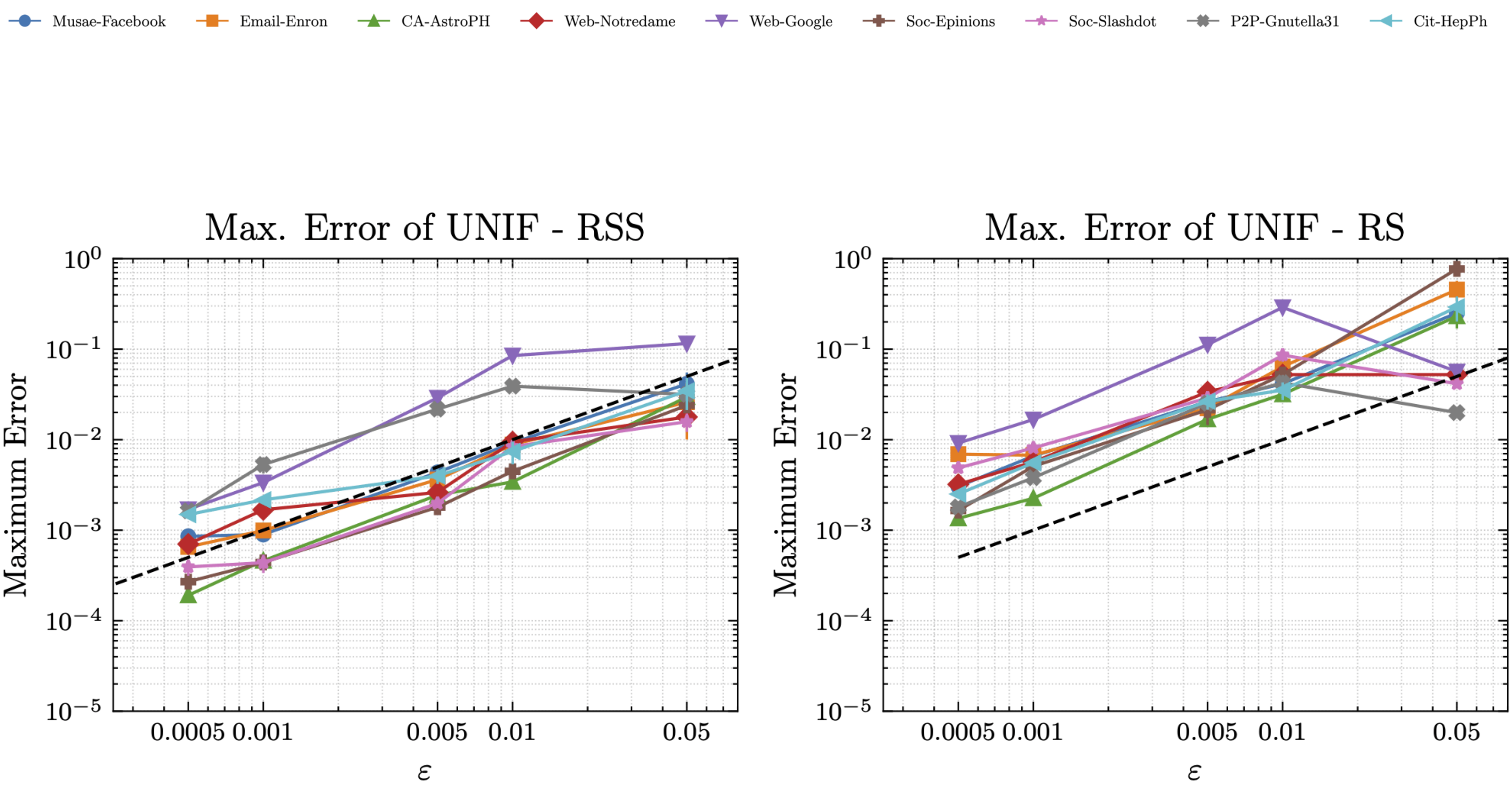
IC - Isolated Component

- Add a small component with mixed states (half 1, half 0).
- Stress test: isolated outbreaks $\rightarrow$ where UNIF usually fails.

UN - Uniform States

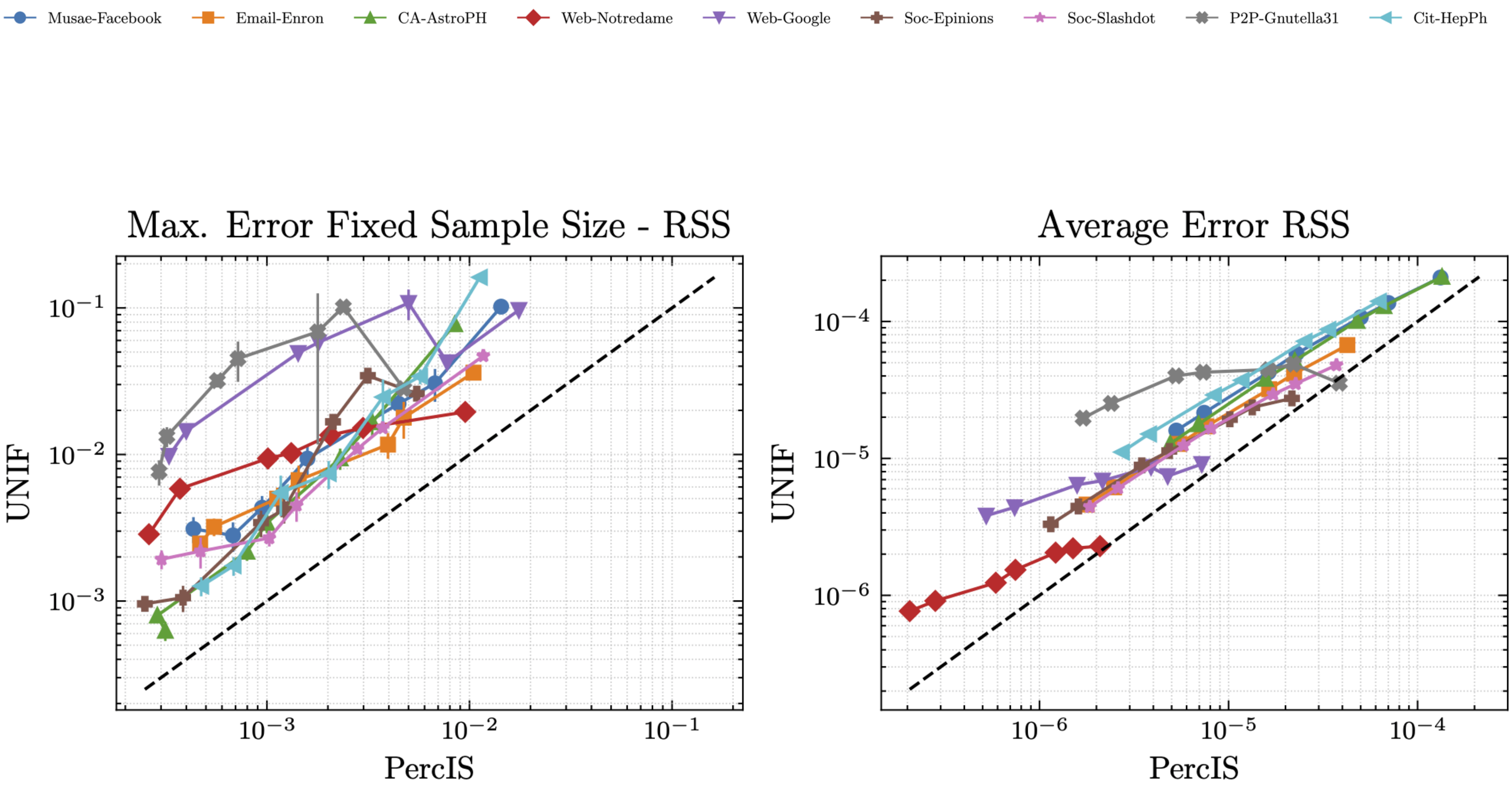
- Assign each node a random value in [0,1].
- Baseline comparison with prior work.

Maximum Error of UNIF using the PD-Bound



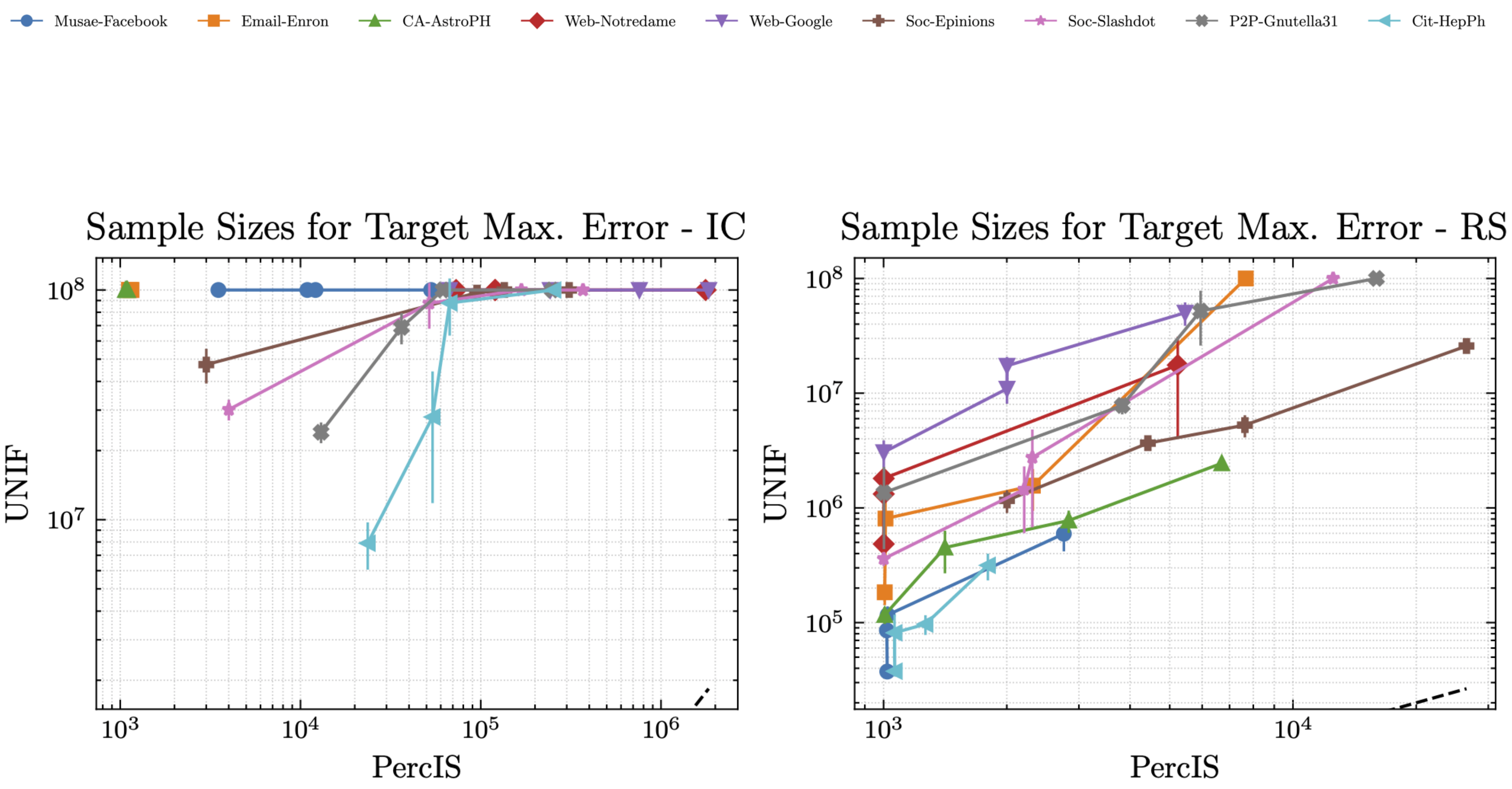
Sample size $\ell \in \mathcal{O}(\ln(D/\delta)/\varepsilon^2)$ for $\varepsilon \in [0.0005, 0.05]$

Maximum Error and Average Error



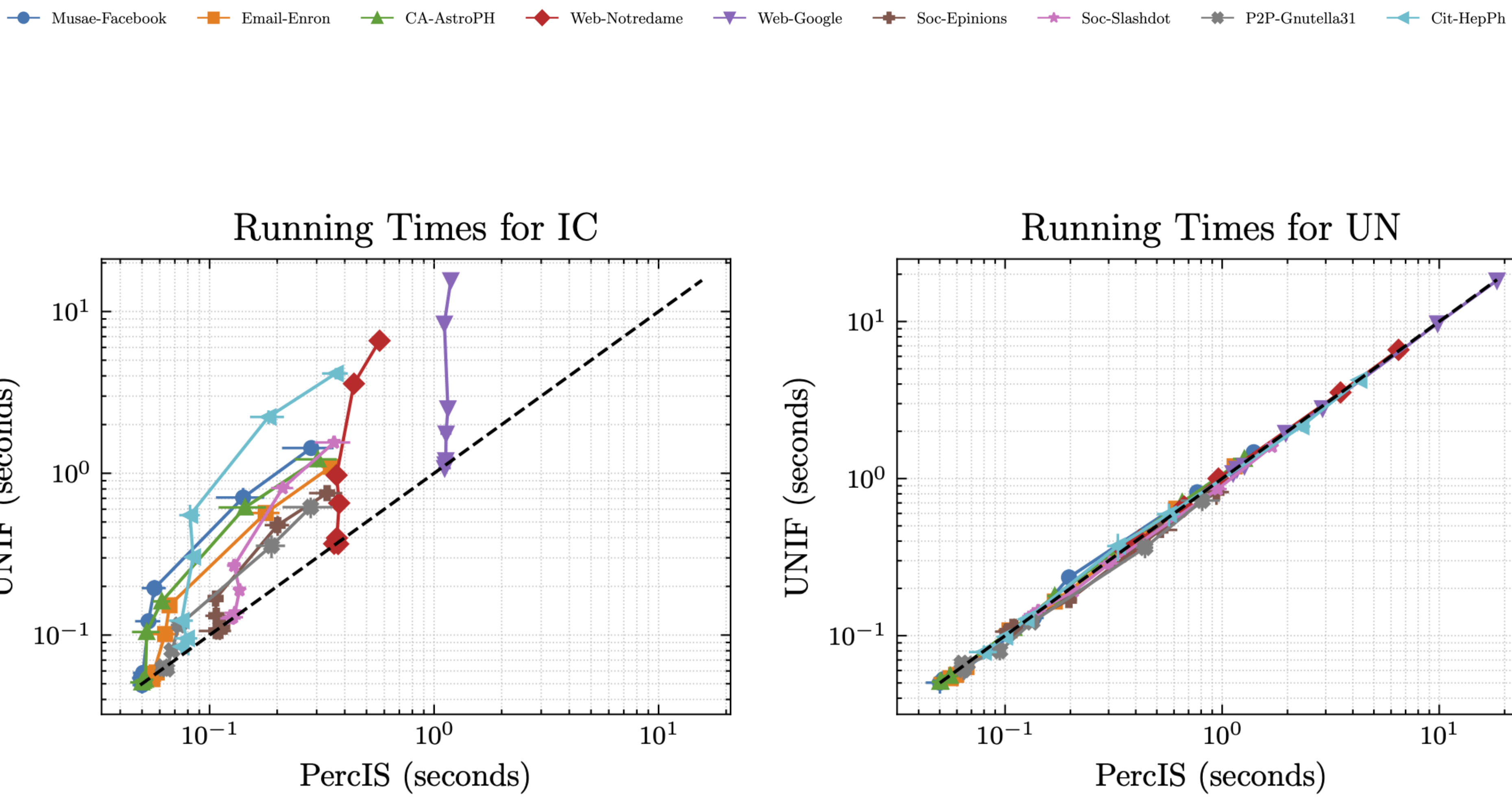
Sample size $\ell \in \{10^3, 5 \cdot 10^3, 10^4, 5 \cdot 10^4, 10^5, 5 \cdot 10^5, 10^6\}$

Sample Size with target ε



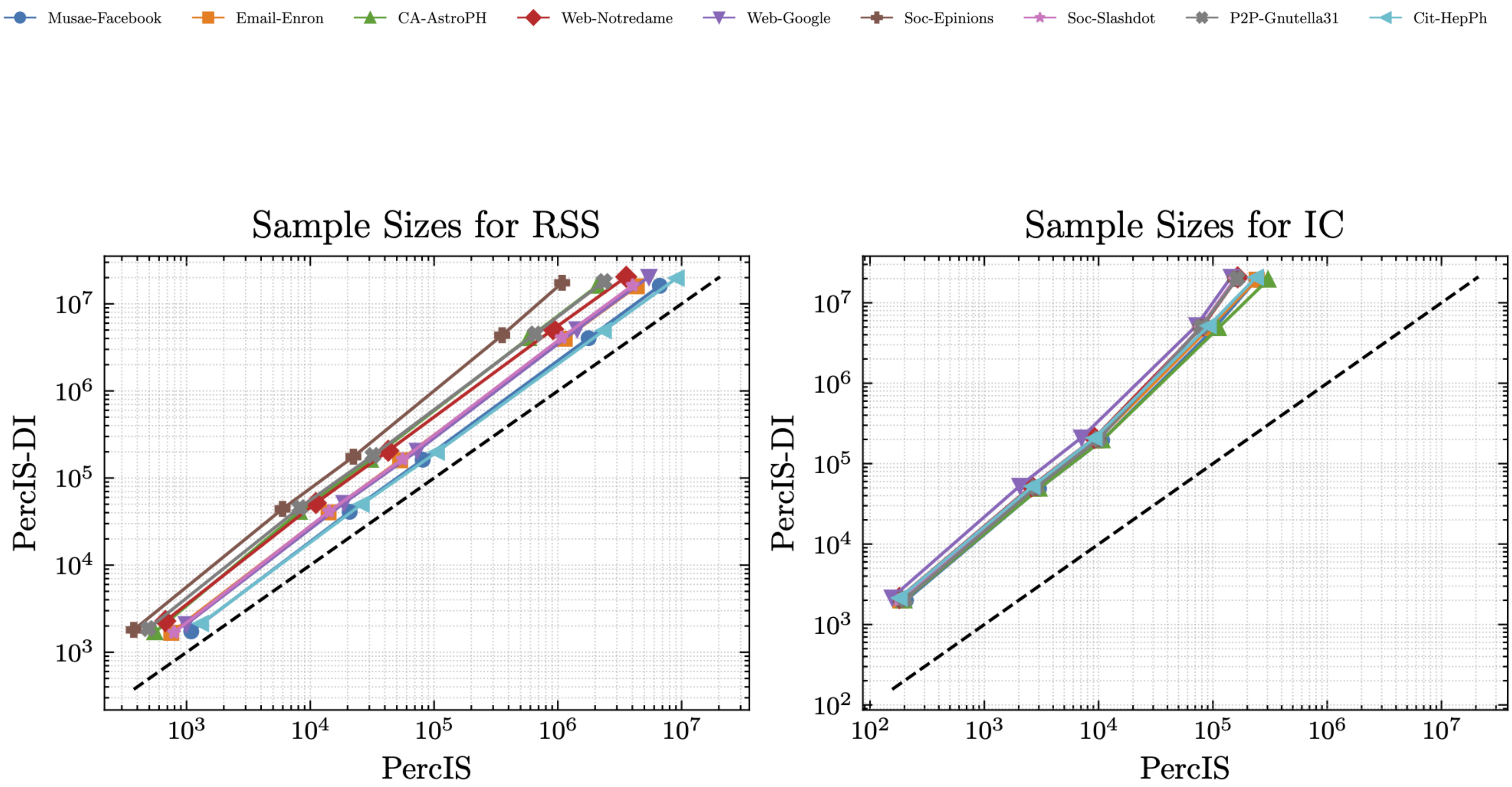
Target ε is set to $(1/k) \cdot \max_{v \in V} p(v)$, for $k \in \{2, 4, 5, 10\}$

Running Times



Sample size $\ell \in \{10^3, 5 \cdot 10^3, 10^4, 5 \cdot 10^4, 10^5, 5 \cdot 10^5, 10^6\}$

Our new Data Dependent bound



$$\varepsilon \in [0.0005, 0.05]$$

Conclusions

- We provide the first practical importance-sampling algorithm for PC.
- New sample complexity analysis for the problem
- PercIS achieves up to 100× fewer samples, orders of magnitude faster than exact, robust across diverse settings
- Uniform sampling fails; PercIS makes percolation centrality scalable and practical.

Thank You

Upper bound on $\hat{\rho}$

Let $\mathcal{S} = \{\tau_1, \ldots, \tau_\ell\}$ be a sample of ℓ shortest paths drawn from q

Define the empirical variance as

$$\Lambda(\mathcal{S}) = \frac{1}{\ell(\ell-1)} \sum_{1 \le i < j \le \ell} \left(|\mathtt{Int}(\tau_i)| - |\mathtt{Int}(\tau_j)| \right)^2$$

Then (via Empirical Bernstein Bound)

$$\boxed{\hat{\rho} = \sum_{v \in V} \tilde{p}(v) + \sqrt{\frac{2\hat{d}\Lambda(\mathcal{S}) \log(1/\delta)}{\ell}} + \frac{7\hat{d}D \log(1/\delta)}{3\ell}}$$

With probability $\ge 1 - \delta$ it holds $\sum_{v \in V} p(v) \le \hat{\rho}$

Upper bound on $\hat{v}$

Let $\mathcal{S} = \{\tau_1, \ldots, \tau_\ell\}$ be a sample of ℓ shortest paths drawn from q

Then (using self bounding functions)

$$\boxed{\hat{v} = \hat{\tilde{d}}^2 \max_{v \in V} \left\{ \tilde{p}(v) + \sqrt{\frac{2\tilde{p}(v)\log(1/\delta)}{\ell}} + \frac{\log(1/\delta)}{3\ell} \right\}}$$

With probability $\geq 1 - \delta$ it holds $\max_v \mathrm{Var}_q\left[\tilde{p}(v)\right] \leq \hat{v}$

The ImportanceSampler + Estimator

For $i = 1, \ldots\ldots, \ell$ DO

1) Sample $s \neq t$ as showed

2) Perform Balanced Bidirectional BFS from s to t

3) Sample a shortest path τ_{st}^{i} and put it in $\mathcal{S}$

4) $\tilde{p}(v) = \dfrac{1}{\ell} \displaystyle\sum_{i=1}^{\ell} \frac{\kappa(s,t,v)}{\tilde{\kappa}(s,t)} 1[v \in \mathtt{Int}(\tau_{st}^{i})]$

PercIS

Algorithm 1: PERCIS

Input: Graph $G = (V, E)$, percolation states $x_1, x_2, \ldots, x_n$, $\ell_1 \geq 1$, $\varepsilon, \delta \in (0, 1)$.

Output: ε-approximation of $\{p(v), v \in V\}$ with probability $\geq 1 - \delta$

$D \leftarrow$ VERTEXDIAMUB(G);

$\mathcal{S} \leftarrow$ IMPORTANCESAMPLER$(G, \{x_v\}, \ell_1)$;

forall $v \in V$ **do** $\tilde{p}(v) \leftarrow \frac{1}{\ell} \sum_{i=1}^{\ell} \frac{\kappa(s,t,v)}{\tilde{\kappa}(s,t)} \mathbb{1}\left[v \in \tau_{st}^{i}\right]$

$\hat{\rho} \leftarrow \sum_{v \in V} \tilde{p}(v) + \sqrt{\frac{2\hat{\tilde{d}}^2 \Lambda(\mathcal{S}) \log(4/\delta)}{\ell_1}} + \frac{7\hat{\tilde{d}} D \log(4/\delta)}{3\ell_1}$;

$\hat{v} \leftarrow \hat{\tilde{d}}^2 \max_{v \in V} \left\{ \tilde{p}(v) + \sqrt{\frac{2\tilde{p}(v) \log(4/\delta)}{\ell_1}} + \frac{\log(4/\delta)}{3\ell_1} \right\}$;

$\hat{x} \leftarrow 1/2 - \sqrt{1/4 - \min\{1/4, \hat{v}\}}$;

$\ell \leftarrow \sup_{x \in (0, \hat{x})} \left\{ \frac{\ln\left(\frac{4\hat{\rho}}{x\delta}\right)}{g(x) h\left(\frac{\varepsilon}{g(x)\tilde{d}}\right)} \right\}$;

$\mathcal{S} \leftarrow$ IMPORTANCESAMPLER$(G, \{x_v\}, \ell)$;

forall $v \in V$ **do** $\tilde{p}(v) \leftarrow \frac{1}{\ell} \sum_{i=1}^{\ell} \frac{\kappa(s,t,v)}{\tilde{\kappa}(s,t)} \mathbb{1}\left[v \in \tau_{st}^{i}\right]$

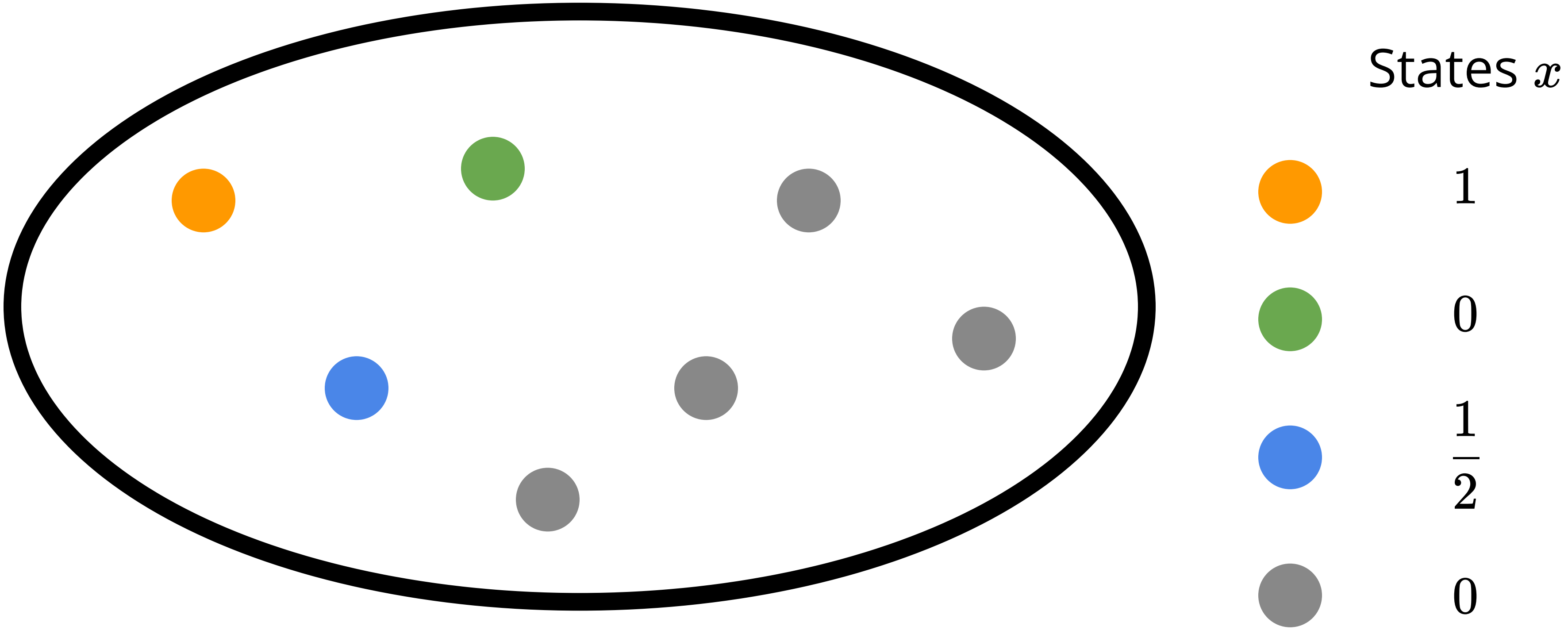
return $\{\tilde{p}(v), v \in V\}$

Observes the graph

Computes APX.

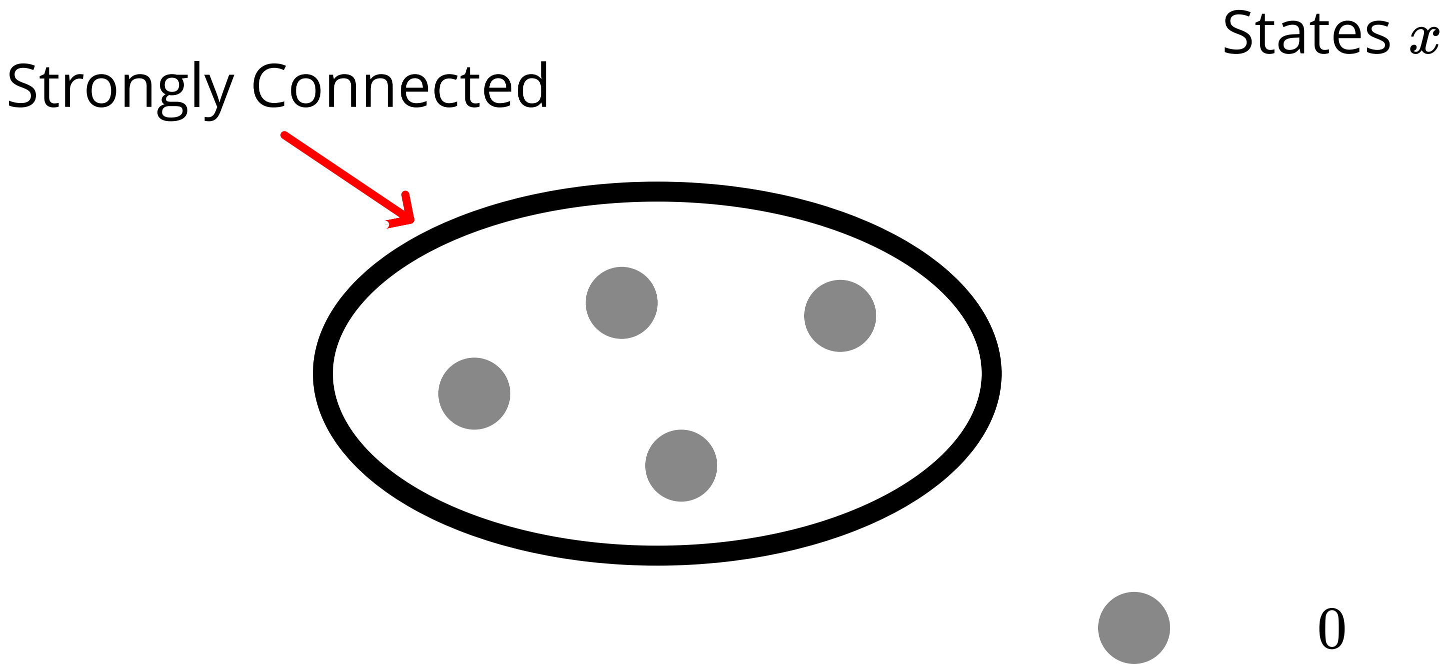
PercIS vs UNIF

There exists instances with $\Delta \in \Omega(1)$ where the likelihood ratio of the uniform distribution is $\Omega(n)$



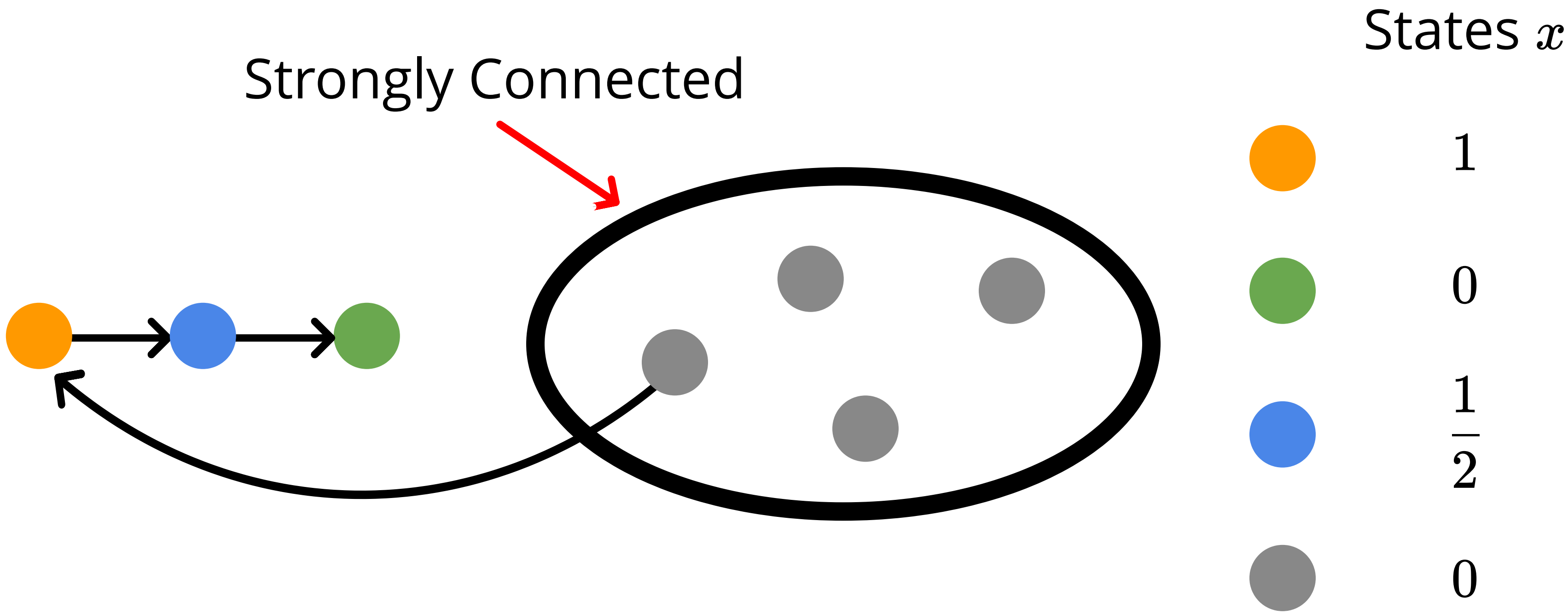
PercIS vs UNIF

There exists instances with $\Delta \in \Omega(1)$ where at least $\Omega(n^2)$ random samples are needed by UNIF, while $\mathcal{O}(n)$ random samples are sufficient for PercIS

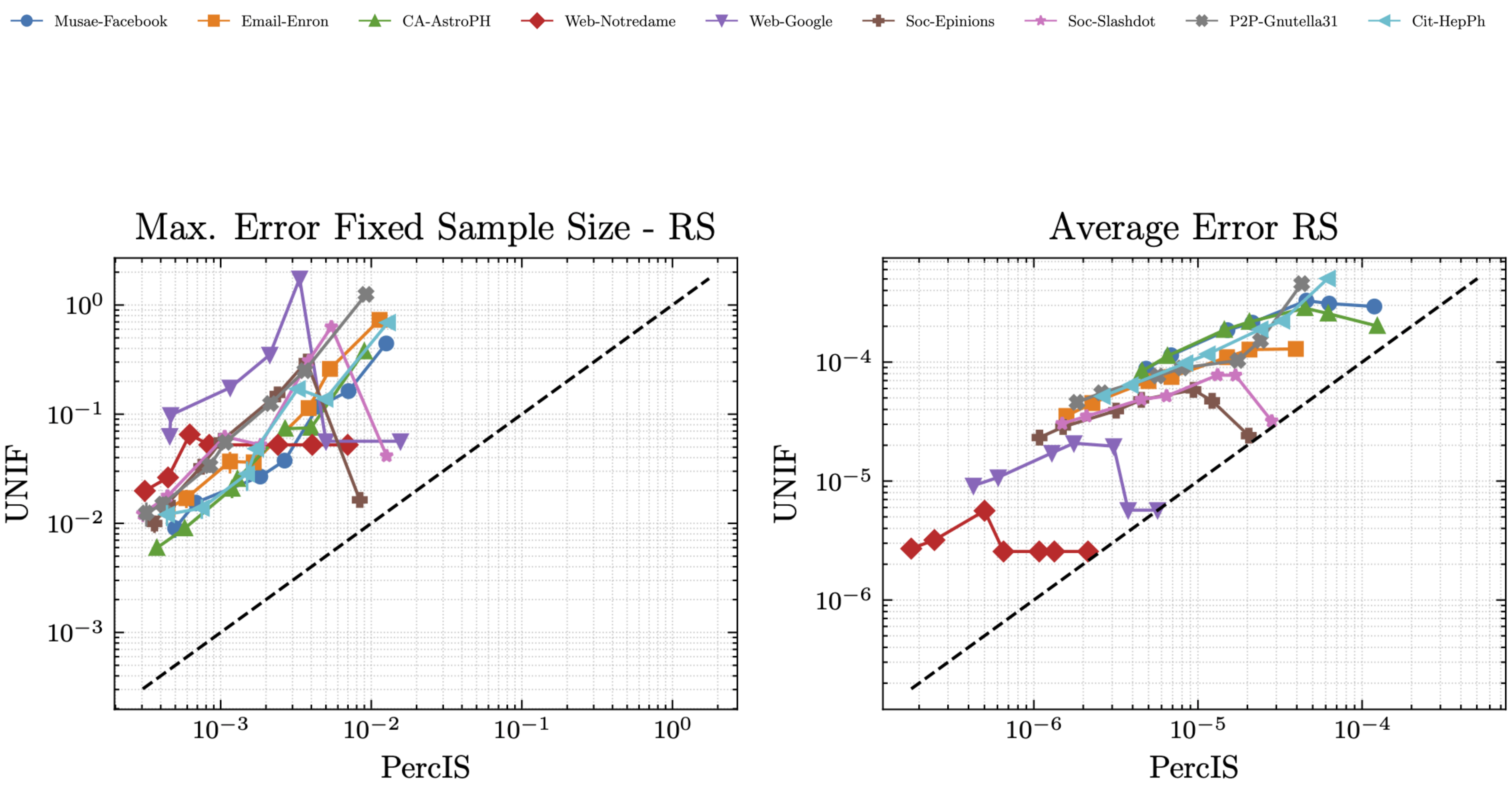


PercIS vs UNIF

There exists instances with $\Delta \in \Omega(1)$ where at least $\Omega(n^2)$ random samples are needed by UNIF, while $\mathcal{O}(n)$ random samples are sufficient for PercIS

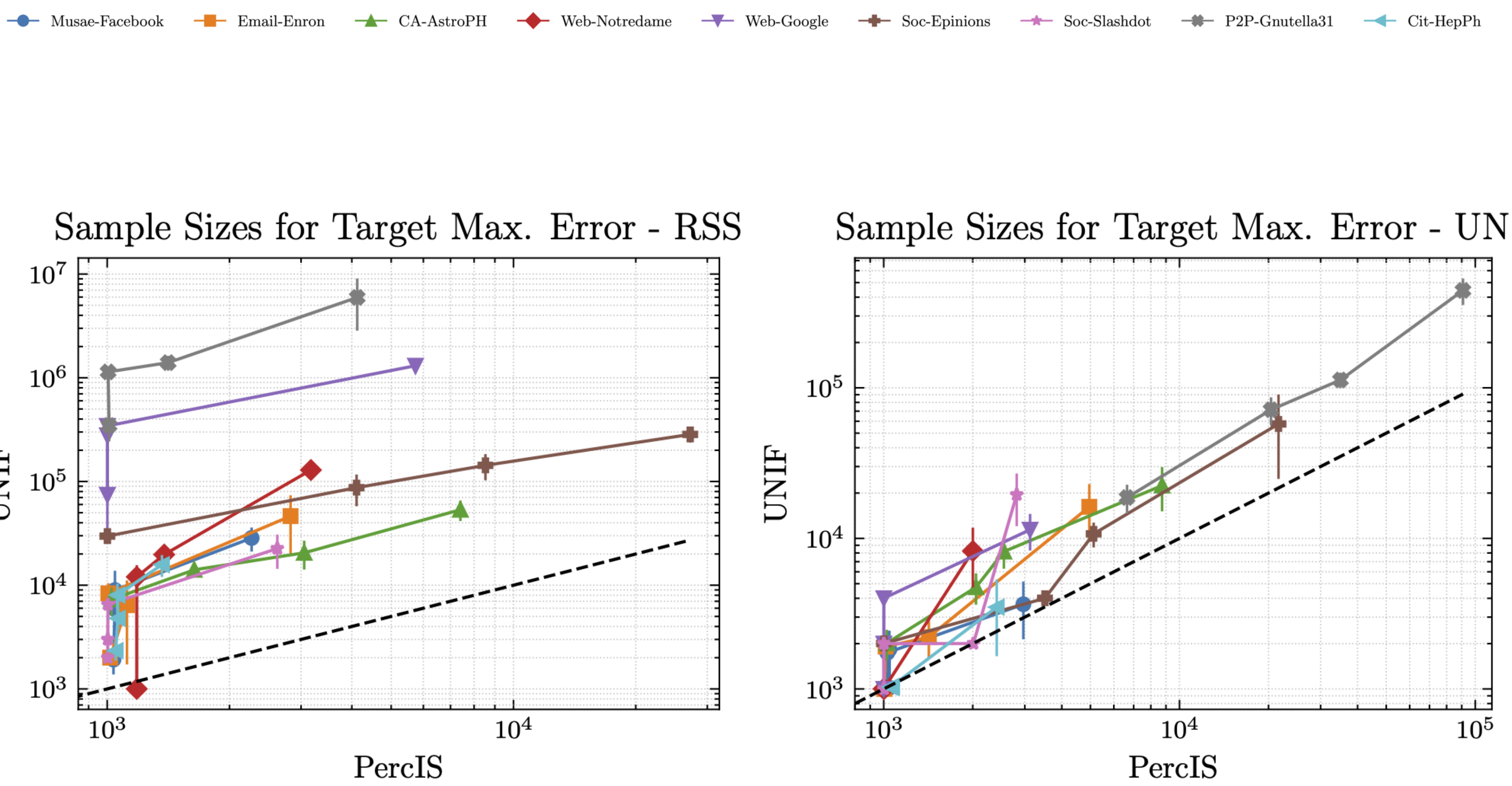


Maximum Error and Average Error



Sample size $\ell \in \{10^3, 5 \cdot 10^3, 10^4, 5 \cdot 10^4, 10^5, 5 \cdot 10^5, 10^6\}$

Sample Size with target ε



Target ε is set to $(1/k) \cdot \max_{v \in V} p(v)$, for $k \in \{2, 4, 5, 10\}$

Running Times