DIMES

Approximation Algorithms for the Management Optimization of a Large Scale Distributed Measurement System

Yuval Shavitt
School of Electrical Engineering
TEL AVIV UNIVERSITY אוניברסיטת תל-אביב
shavitt@eng.tau.ac.il
http://www.netDIMES.org

Who

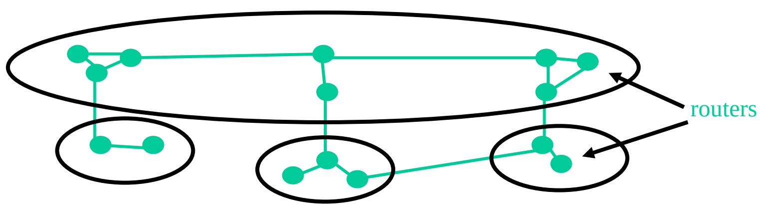
- PI: **Yuval Shavitt**
- Ph.D. students: **Eran Shir**, Tomer Tankel, Miriam Allalouf, **Udi Weinsberg**
- Master's students: Dima Feldman, Elad Kaplan, Anat Halpern, Yaron Singer, Udi Weinsberg, Yaron Schwartz, Nir Chen
- Programmers: Anat Halpern, Ohad Serfati, Yoav Freund, Ela Malinovski, **Ido Blutman, Galina Ryvchin, Boaz Harel**
- Undergrads: *long list*
- Collaborators: HUJI, Collegium Budapest, BIU

Approximation result:
Joint work with **Mira Gonen**

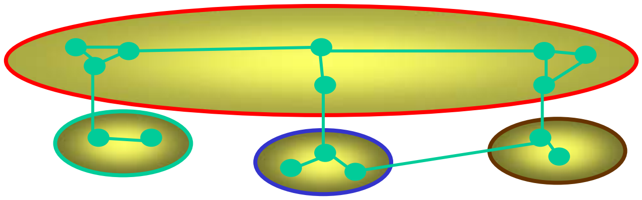
Why DIMES?

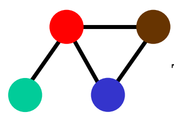
- The Internet is an engineered system, so someone must know how it is built, no?
- NO! It is an uncoordinated interconnection of Autonomous Systems (ASes=networks).
- No central database about Internet structure.
- Several projects attempted to reveal the structure: Skitter, RouteViews, …

The Internet Structure



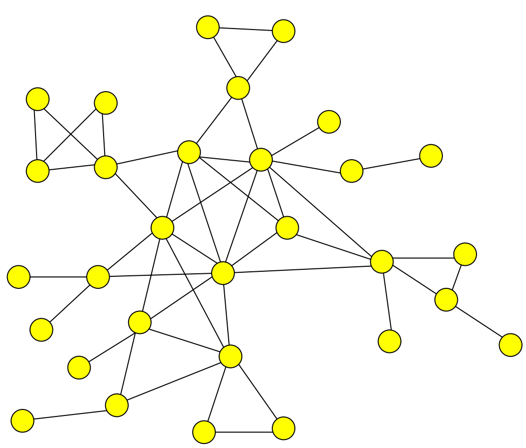
The Internet Structure



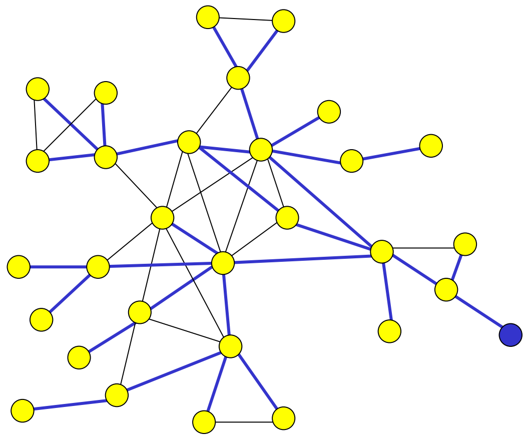


The **AS** graph

Revealing the Internet Structure

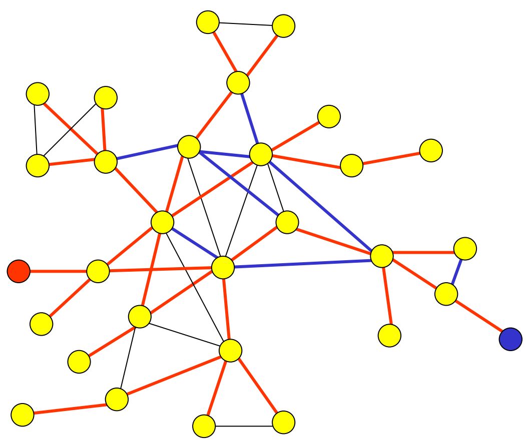


Revealing the Internet Structure



7 Yuval Shavitt - Tel Aviv University www.netDIMES.org

Revealing the Internet Structure



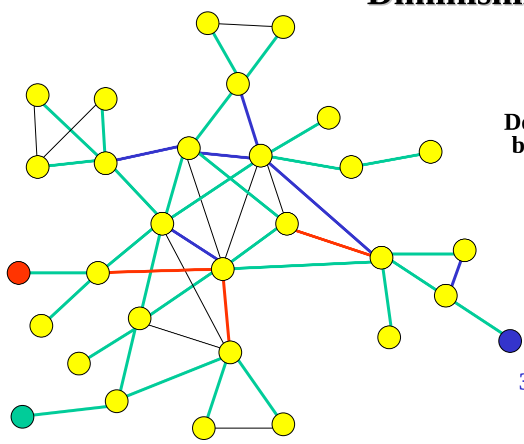
8 Yuval Shavitt - Tel Aviv University www.netDIMES.org

Revealing the Internet Structure

Diminishing return!

⇓

Deploying more boxes does not pay-off



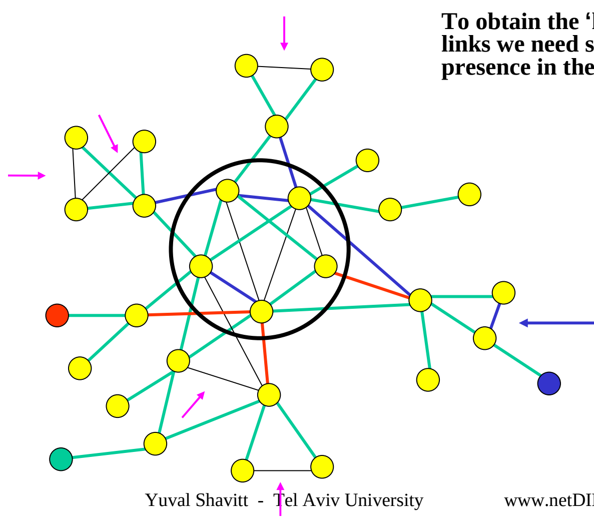
7 new links

30 new links

NO new links

Revealing the Internet Structure

To obtain the 'horizontal' links we need strong presence in the edge

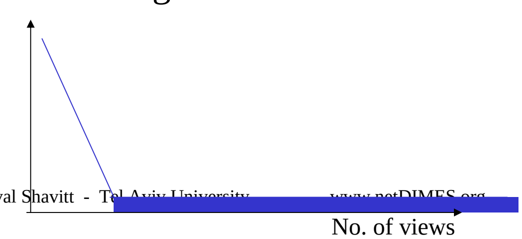


Diminishing Return?

- [Chen et al 02], [Bradford et al 01]: when you combine more and more points of view the return diminishes very fast
- What have they missed?

Diminishing Return?

- [Chen et al 02], [Bradford et al 01]: when you combine more and more points of view the return diminishes very fast
- What have they missed?
  - The mass of the tail is significant

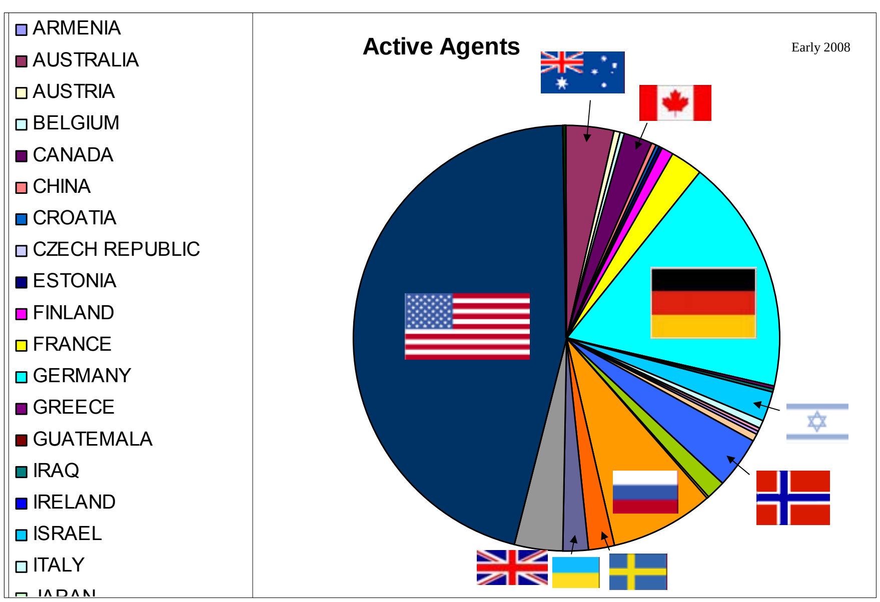


DIMES: Why and What

DIMES

- Diminishing return?
  - Replace instrumentation boxes with software agents
  - Ask for volunteers do help with the measurement

    ⇓
  - The cost of the first agent is very high
  - each additional agent costs almost zero
- Advantages
  - Large scale distribution: view the Internet from everywhere
  - Remove the "academic bias", measure the commercial Internet
- Capabilities
  - Anything you can write in Java!
  - Obtaining Internet maps at all granularity level with annotations
    - connectivity, delay, loss, bandwidth, jitter, ….
  - Tracking the Internet evolution in time
  - Monitoring the Internet in real time



The Validation Problem

- To make periodic maps we need to make sure previously seen edges still exist.
- We maintain for each agent the list of all traceroute it ever perfomed
  – Traceroute: a group of edges
- An agent can perform 2-4 traceroute a minute
- The goal: to validate all edge existence in minimum time.

Gonen & Shavitt, IPL, to appear

The Validation Problem (2)

- Set Cover: cover the graph with the minimal set of traceroute
  – One agent may need to perform many traceroutes ⇒ the validation will be slow
- We want minimize the maximum number of traceroute per agent

The Set Cover Problem

- Problem Definition: given a universe $U = \{u_1,\ldots,u_n\}$ and $\mathcal{S}$ a family of its subsets $\mathcal{S} = \{S_1,\ldots,S_k\} \subseteq P(U)$ s.t. $\cup_{S_j \in \mathcal{S}} S_j = U$, set cover is the problem of finding a minimal sub-family $\hat{S}$ of $\mathcal{S}$ that covers the whole universe, $\cup_{S_j \in \hat{S}} S_j = U$.
- Set Cover can be approximated to within $\ln n - \ln\ln n + \theta(1)$ [Slavik 95, Srinivasan 99].
- Set Cover cannot be efficiently approximated to within $(1 - o(1)) \ln n$ [feige 98].

Validation Set Cover – VSC

Problem Definition: given
- a universe $U = \{u_1,\ldots,u_n\}$ (edges)
- $\mathcal{S}$ a family of its subsets $\mathcal{S} = \{S_1,\ldots,S_k\} \subseteq P(U)$ s.t. $\cup_{S_j \in \mathcal{S}} S_j = U$, (traceroutes)
- a partition of $\mathcal{S}$ $\pi = \{A_1,\ldots,A_m\}$ where $A_i \subseteq \mathcal{S}$,
- and a weight function $\omega: \pi \to N$, (agent speed)

find a sub-family $\hat{S}$ of $\mathcal{S}$ that covers the whole universe U s.t.
$\max_{1 \le i \le m} \lceil |A_i \cap \hat{S}| / \omega(A_i) \rceil$ is minimum.

For m = 1 the VSC problem is exactly the set cover problem.
$\Rightarrow$The VSC problem is also $\mathcal{NP}$-hard.

A O(log n) Approximation Algorithm for VSC

1. ℓ:=0
2. C:=0
3. While $C \neq U$
   1. ℓ:=ℓ+1
   2. For $1 \leq i \leq m$
      1. Repeat $\omega(A_i)$ times
         1. Find a set S_j s.t. $S_j \in A_i$ and $S_j \cap (U \setminus C)$ is maximum
         2. Pick S_j
         3. $C := C \cup S_j$
4. Output ℓ

S_j covers maximum uncovered elements

Choose the subsets for A_i

A O(log n) Approx. Alg. for VSC

Main Theorem:

The above algorithm gives an approximation ratio of O(log n).

Ideas and Techniques

Define the ℓ- residual VSC problem:

- Input: the input to the VSC problem after ℓ steps of the algorithm, with the same objective.
- n_ℓ = be the number of elements in U that remain after ℓ steps of the algorithm.
- For $\ell = 0$: $n_\ell = n$.
- C_ℓ = the set of elements in U that are covered until step ℓ.

The ℓ- residual VSC Problem

Formal Definition

- For all $1 \le j \le k = |\mathcal{S}|$
  - $S_j^\ell = S_j \setminus C_\ell$.
  - For all $1 \le i \le m$
    $A_i^\ell = A_i \setminus \{S_j \in A \mid S_j$ has been picked until step $\ell\}$
  - $\mathcal{S}^\ell = \{S_j^\ell \mid S_j^\ell \neq \phi\}$
  - For all $1 \le i \le m$ $\omega(A_i^\ell) = \omega(A_i)$.
- Opt_ℓ = the optimal solution of the residual input after ℓ steps.

→ $Opt_\ell = \min_{\hat{S}^\ell} \max_{1 \le i \le m} \lceil |A_i^\ell \cap \hat{S}^\ell| / \omega(A_i^\ell) \rceil$
where $\hat{S}^\ell$ is a subcollection of $\mathcal{S}^\ell$ that covers all elements of $U \setminus C_\ell$.

Ideas and Techniques

Claim: at step $\ell \geq 1$ of the algorithm at least $n_{\ell-1}/opt_{\ell-1}$ elements in U are covered.

Proof: main observation - any optimal algorithm covers all the elements in opt stages.

→ there exists a stage in which at least n/opt elements are covered. Since the order of the stages does not change opt, assume w.l.o.g that at stage 1 any optimal algorithm covers at least n/opt elements.

Claim's Proof

→ (since our algorithm is greedy):

For ℓ=1: at least n/opt = $n_{\ell-1}/opt_{\ell-1}$ elements are covered at step ℓ.

For ℓ>1: an optimal algorithm covers all the $n_{\ell-1}$ remaining elements of $U \setminus C_{\ell-1}$ in $opt_{\ell-1}$ steps.

→ (since our algorithm is greedy):

At least $n_{\ell-1}/opt_{\ell-1}$ elements are covered at step ℓ. □

Ideas and Techniques

Lemma: $n_\ell \leq n(1-1/opt)^\ell$.

Proof: by induction on ℓ.

- Claim
  $\rightarrow$ at step $\ell \geq 1$ of the algorithm at least $n_{\ell-1}/opt_{\ell-1}$ elements in U are covered.
- For all ℓ: $opt_\ell \leq opt$.

  - $n_1 \leq n - n/opt = n(1-1/opt)$.
  - Assume that for all $i < \ell$ it holds that $n_i \leq n(1-1/opt)^i$

  $\rightarrow n_\ell \leq n_{\ell-1} - n_{\ell-1}/opt_{\ell-1} \leq n(1-1/opt)^\ell$. □

Proof of the main Theorem

- In the worst case the algorithm stops after $\ell + 1$ steps for the minimal ℓ such that $n_\ell \leq 1$.
- Lemma $\rightarrow n_\ell \leq n(1-1/opt)^\ell$

$\rightarrow$ For that ℓ with $n(1-1/opt)^\ell \leq 1$ it holds that $n_\ell \leq 1$.

$\rightarrow n(1-1/opt)^\ell \leq 1 \leftrightarrow \ell \leq \log n / \log(1+1/(opt-1))$

- $\log n / \log(1+1/(opt-1)) \leq \log n \cdot opt$

Using Taylor series

$\rightarrow$ the number of steps used by the algorithm is at most $1 + \log n \cdot opt$.

□

Lower Bound for the VSC problem

- Unless $P = NP$ the approximation ratio of the set cover problem is $\Omega(\log n)$ [Feige].
- For m = 1 and for $1 \leq i \leq m$ $\omega(Ai) = 1$.

→ The VSC problem is exactly the set cover problem.

→ Unless $P = NP$ the approximation ratio of the VSC problem is $\Omega(\log n)$.

→ Unless $P = NP$ the approximation ratio of the VSC problem is $\theta(\log n)$.

Other Related Problem

Concurrent Monitoring Experiments:

- An important use of highly distributed Internet measurement systems is to monitor the network.
- Multiple experiments can be run concurrently, each has a certain set of links it wishes to monitor.
- For an experiment to be meaningful one must be able to monitor all the targeted links in the experiment.
- A database of all the different measurements which can be performed by the server is maintained.

Concurrent Monitoring Experiments

- Each measurement can monitor a set of Internet links at the same time.
- Given the limited measurement capabilities of a server, not all the experiments can be performed simultaneously.

→ Our goal:

Find a maximum set of experiments that can be performed at the same time, using a fixed number of measurements.