Network Tomography

Don Towsley

Umass-Amherst

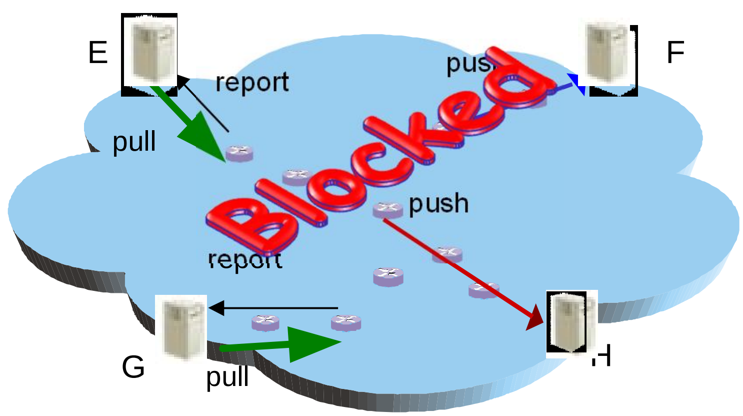
Collaborators: T. He, K. Leung, C. Liu,
L. Ma, A. Swami

Motivation

Knowledge of link behavior crucial for network operations:

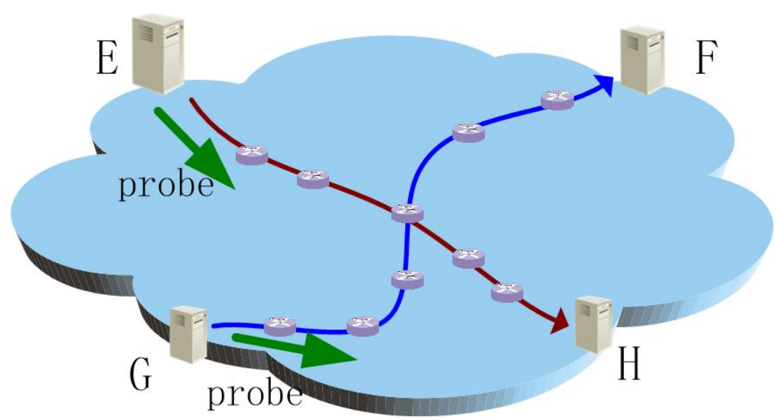
Direct measurements

- requires administrative privileges



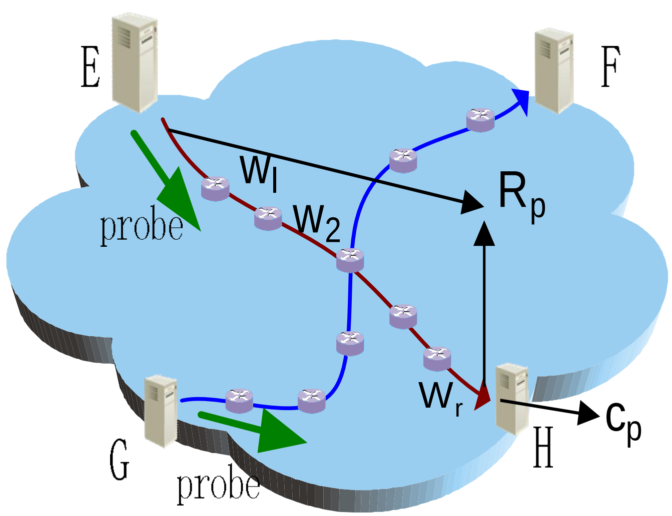
Network tomography

- infer link metrics from end-to-end measurements
- no administrative privileges



Introduction

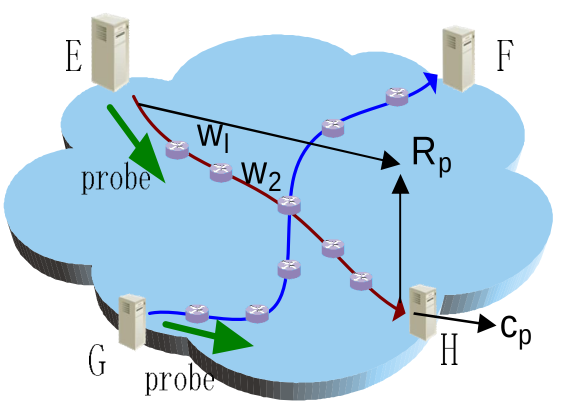
Additive link metrics (e.g., delay, log of delivery probability)

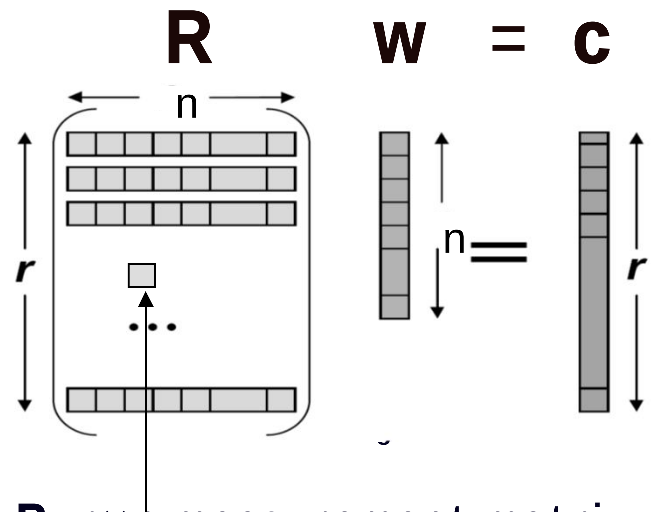


$$C_p = W_1 + W_2 + \cdots + W_r$$

Introduction

Additive link metrics (e.g., delay, log of delivery probability)





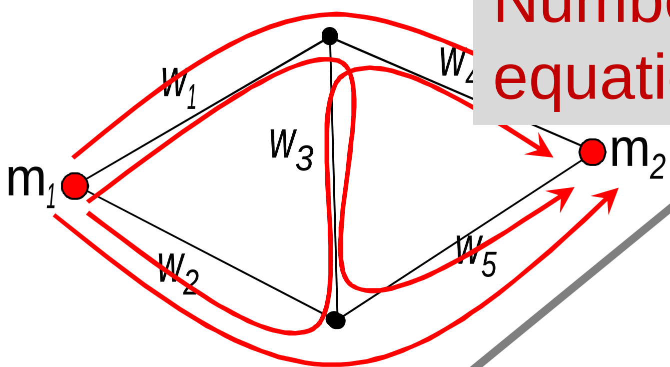
n: #links;
r: #measurement paths
w: link metrics (unknown)
c: path metrics (known)

R: r×n measurement matrix

Goal: "invert" linear system (**w** =?):

- if invertible, then w = $R^{-1}c$
- not always invertible! (linearly dependent paths)

Challenges



Number (linearly independent) equations = 4

= **c**

$$w_2+w_5 = c_2$$

$$w_1+w_3+w_5 = c_3$$

$$w_2+w_3+w_4 = c_4$$

$$\Rightarrow \begin{pmatrix} \\ 0\ 1\ 0\ 0\ 1 \\ 1\ 0\ 1\ 0\ 1 \\ 0\ 1\ 1\ 1\ 0 \end{pmatrix} \begin{pmatrix} \\ w_2 \\ w_3 \\ w_4 \\ w_5 \end{pmatrix} = \begin{pmatrix} c_1 \\ c_2 \\ c_3 \\ c_4 \end{pmatrix}$$

rank(**R**) = 4, but 5 variables: **R** not invertible ⟹ **w** not uniquely determined

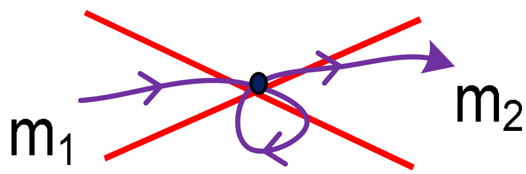
- topological conditions for identifiability?
- how to place monitors, construct paths?

Outline

- motivation
- problem formulation
- identifiability
- monitor placement
- partial identifiability
- summary

Problem formulation

- link metrics additive, constant
- known undirected topology, G = (V, L)
  - V: set of nodes, L: set of links, n = |L|
  - links ij, ji have same metrics
- monitor set $M \subset V$, start/report measurements
- measurement paths: controllable arbitrary cycle-free (simple) paths



Objectives:

- topological conditions for identifiability
- monitor placement, path selection
- partial identifiability

Conditions for identifiability

Related work

- emphasis on inferring path segment metrics from available path measurements
- unique identification of underlying link metrics usually ignored

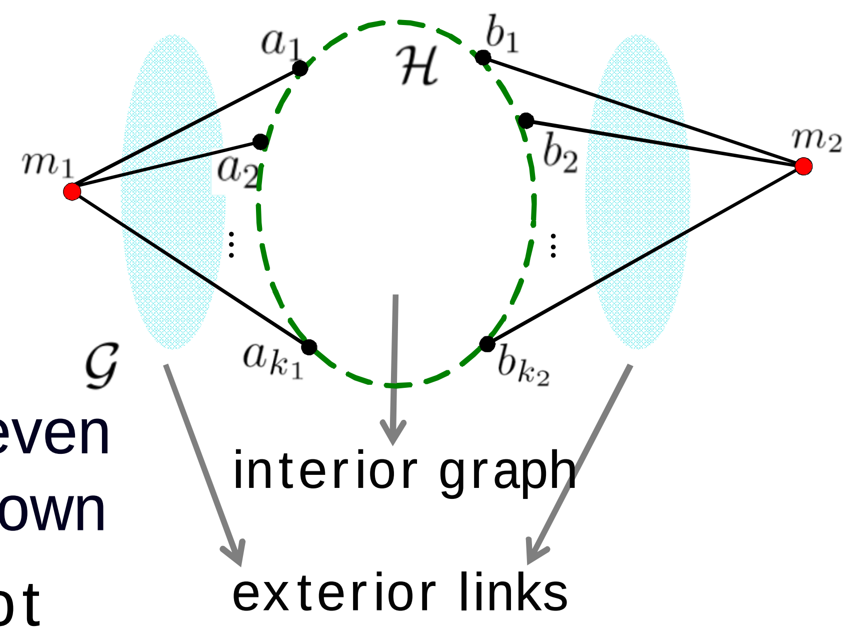
Q: Fundamental topological conditions ensuring unique identifiability of link metrics?

- related work applies to special cases:
  - binary link metrics (0, 1) (Ahuja et al '08)
  - measurement paths with cycles in (Gopalan et al '11)

Two monitors

Thm. *G* always unidentifiable using two monitors

Proof. Network partitions
into interior graph $\mathcal{H}$,
exterior links



Exterior links unidentifiable even if all interior link metrics known

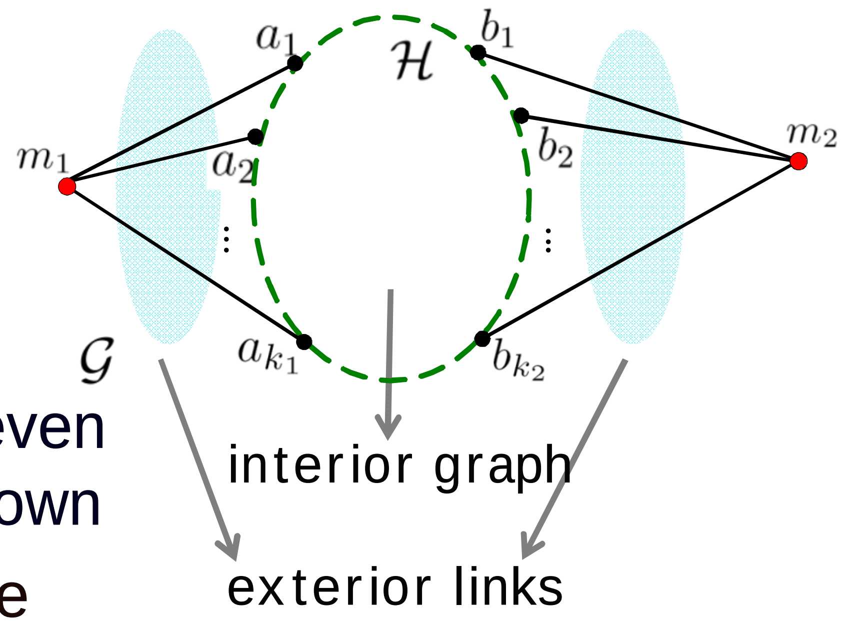
Can infer $w_{m_1 a_i} + w_{a_j m_2}$ but not

$w_{m_1 a_i}$, $w_{a_j m_2}$

Two monitors

Thm. G always unidentifiable using two monitors

Proof. Network partitions into interior graph $\mathcal{H}$, exterior links



Exterior links unidentifiable even if all interior link metrics known

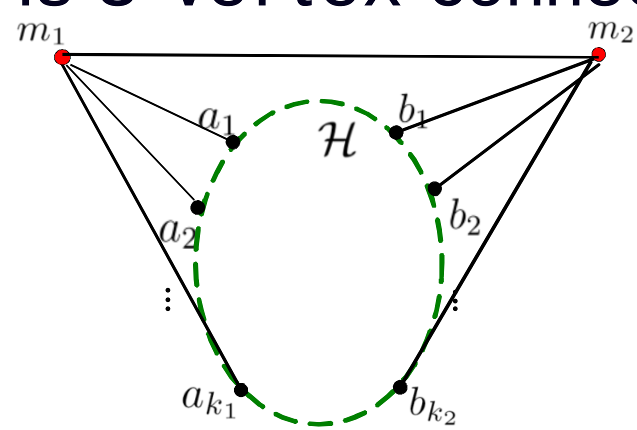
$\Longrightarrow$ no exterior link identifiable using two monitors

Possible to identify interior graph?

Two monitors

Thm. Interior graph H of G identifiable iff:

1. $G - l$ is 2-edge connected for each interior link $l \in H$
2. $G + m_1 m_2$ is 3-vertex connected



*2-edge-connected: delete 1 edge ⟶ still connected

*3-vertex-connected: delete 2 nodes ⟶ still connected

Two monitors

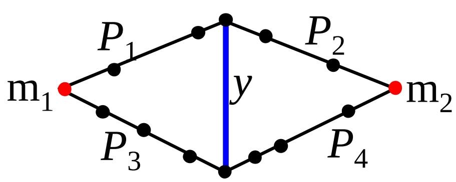
Thm. Interior graph H of G identifiable iff:

1. $G - l$ is 2-edge connected for each interior link $l \in H$
2. $G + m_1 m_2$ is 3-vertex connected

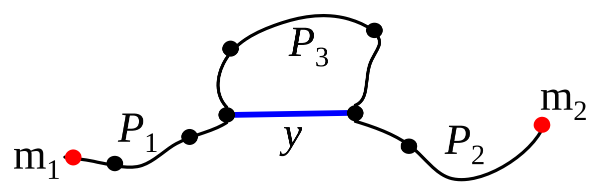
Proof.

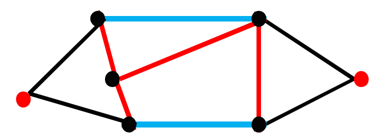
- necessary part: contradiction
- sufficient part: interior links either

cross link



short cut





$$\begin{cases} P_A = P_1 \cup P_2 \\ P_B = P_3 \cup P_4 \end{cases}, \quad \begin{cases} P_C = P_1 \cup y \cup P_4 \\ P_D = P_3 \cup y \cup P_2 \end{cases},$$

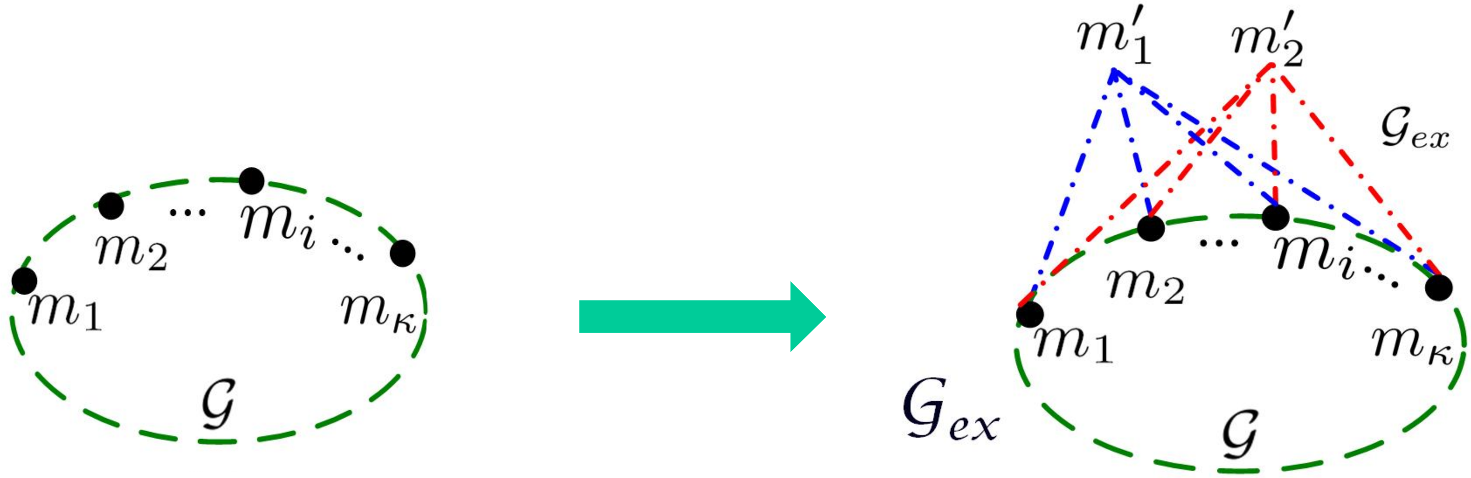
$$w_y = \frac{1}{2}(w_{P_C} + w_{P_D} - w_{P_A} - w_{P_B})$$

$$\begin{cases} P_A = P_1 \cup y \cup P_2 \\ P_B = P_1 \cup P_3 \cup P_2 \end{cases}$$

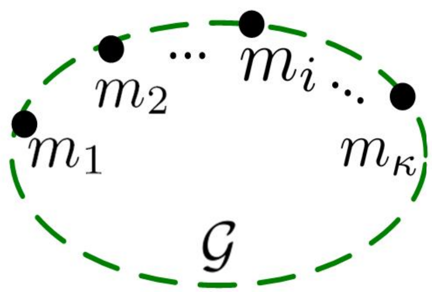
$$w_y = w_{P_A} - w_{P_B} + w_{P_3}$$

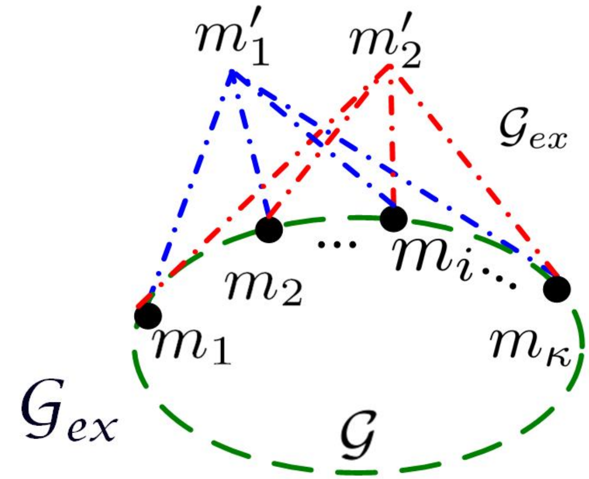
Three or more monitors

- convert $\kappa \geq 3$ monitor case to 2-monitor case
- construct extended graph G_{ex}
- add 2 virtual nodes m'_1, m'_2, 2κ virtual links



Three or more monitors





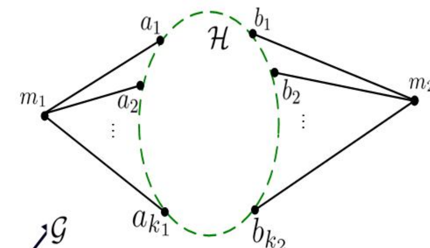
G_{ex}-l 2-edge-connected for each link l $\in G$ $\Leftrightarrow$ G_{ex} 3-edge-connected

G_{ex}+m'$_1$m'$_2$ 3-vertex-connected $\Leftrightarrow$ G_{ex} 3-vertex-connected (also 3-edge-connected)

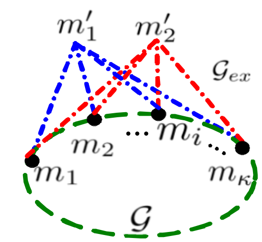
Thm. With $\kappa \geq 3$ monitors, G identifiable iff G_{ex} is 3-vertex-connected

Summary of identifiability conditions

Path type	#monitors	Condition for identifiability
Non-simple paths Gopalan'11	1	3-edge-connected
	≥ 3	Each component in G-l_1-l_2 has monitor
Cycle-free paths	2	Entire G: impossible
		Interior: G- l 2-edge-connected; $G+m_1m_2$ 3-vertex-connected
	≥ 3	G_{ex} 3-vertex-connected



Interior graph H



Extended graph G_{ex}

Testing algorithm:

- interior identifiability, 2 monitors: O(|L|(|V|+|L|))
- complete identifiability, $\kappa \geq 3$ monitors: O(|V|+|L|)
- 2-edge-connected in O(|V|+|L|) (Tarjan '74)
- 3-vertex-connected in O(|V|+|L|) (Hopcroft, Tarjan '73)

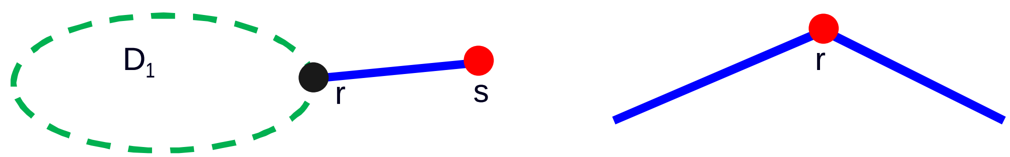
Minimum monitor placement

Minimum monitor placement (MMP)

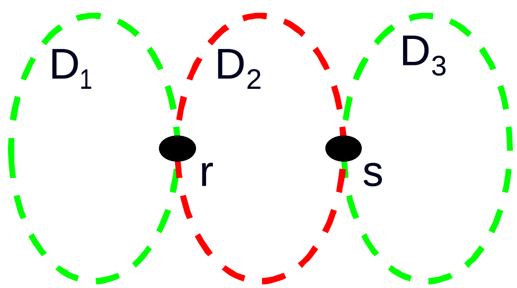
Q: minimum # monitors, placement to identify given G?

General rules: in addition to total #monitors ≥ 3

1. nodes with degree 1, 2 must be monitors



2. 3-vertex-connected subgraph identifiable iff has 3 "monitors"
   - "monitor" includes
     i. actual monitors
     ii. connecting points w/identified subgraphs



If D_1, D_3 identified, then r, s are "monitors" for D_2

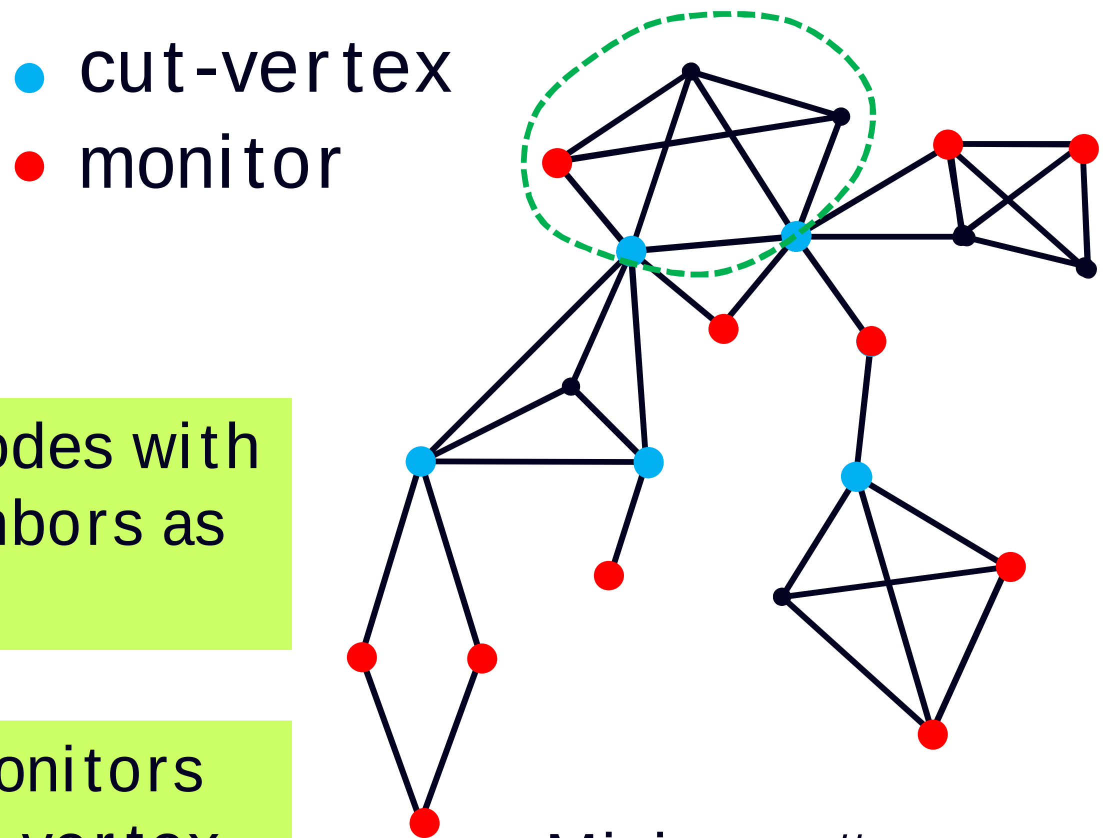
Minimum monitor placement

Algorithm MMP

a) select degree 1, 2 nodes as monitors
b) decompose *G* into 3-vertex connected components, select necessary monitors in each component (3 effective monitors)

- places necessary monitors in *G*
  - sufficient to identify all links in the network
  - MMP places **minimum** # monitors to identify all links in *G*
- linear time complexity: O(|V|+|L|)
- under uncontrollable routing (Kumar, etal '06): minimum monitor placement NP-hard

MMP: Example



Select nodes with 1, 2 neighbors as monitors

Select monitors in each 3-vertex-connected component

Minimum # monitors - 11

Evaluation – benchmark, setting

Benchmark

- random monitor placement (RMP):
- randomly select κ monitors, test network identifiability

Topologies

- random: Erdos-Renyi (ER), Random Geometric (RG), Barabasi-Albert (BA), and Random Power Law (PL) graphs
- ISP: Rocketfuel, CAIDA

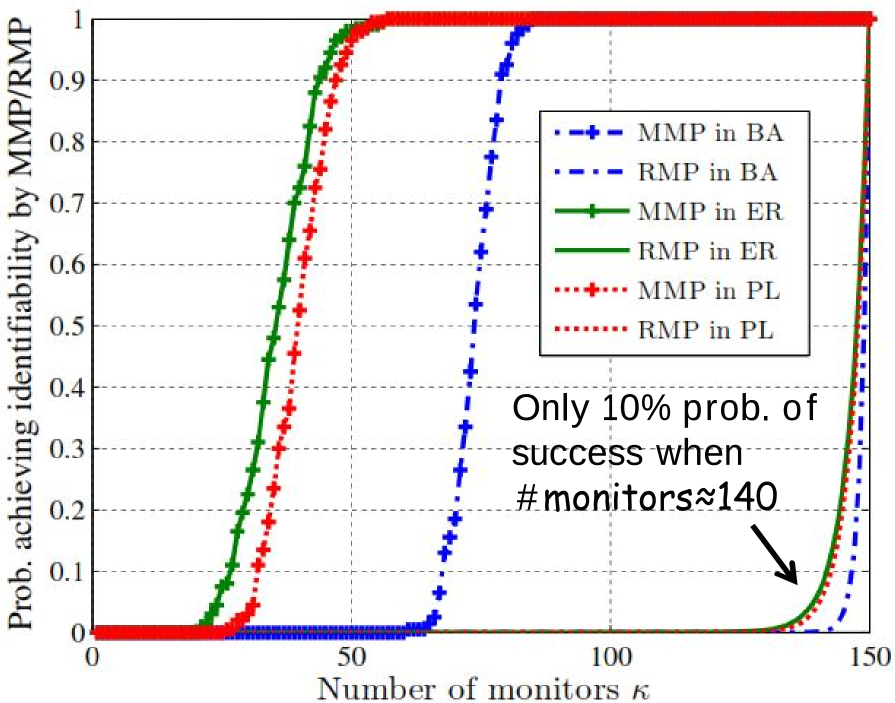
Performance metric

- fraction achieving network identifiability over multiple simulations

Evaluation – random topologies

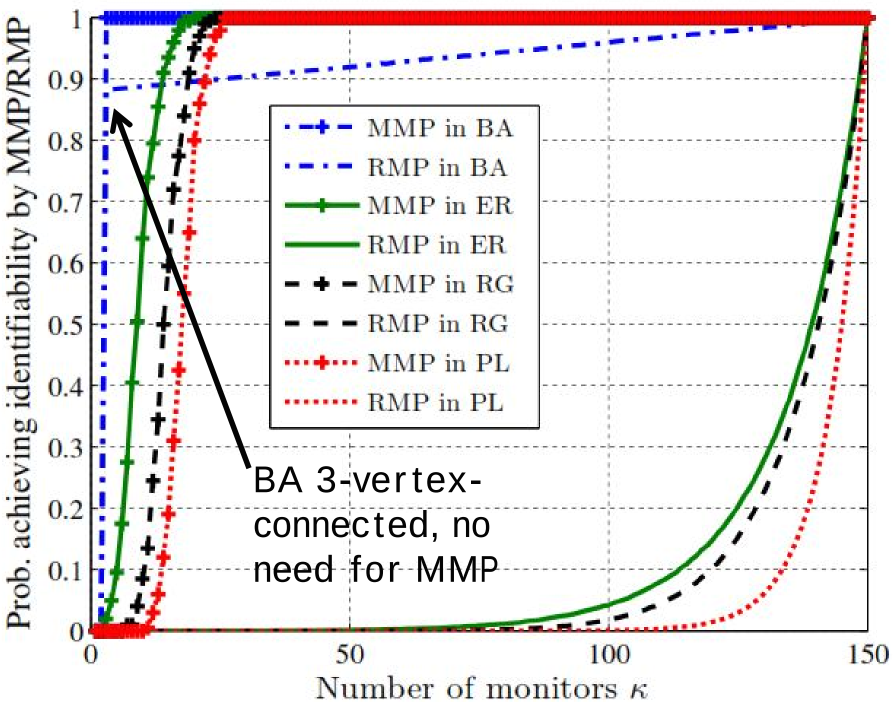
- 100 graphs, 2000 simulations for each graph
- |V| = 150

Sparsely-connected, |L| ≈ 295

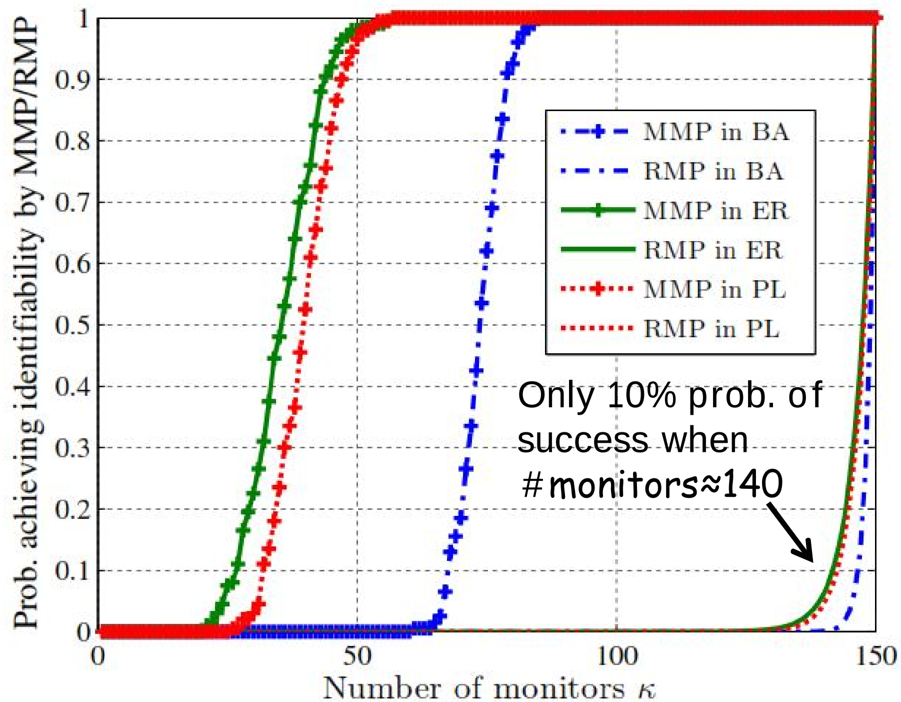


Evaluation – random topologies

- 100 graphs, 2000 simulations for each graph
- |V| = 150



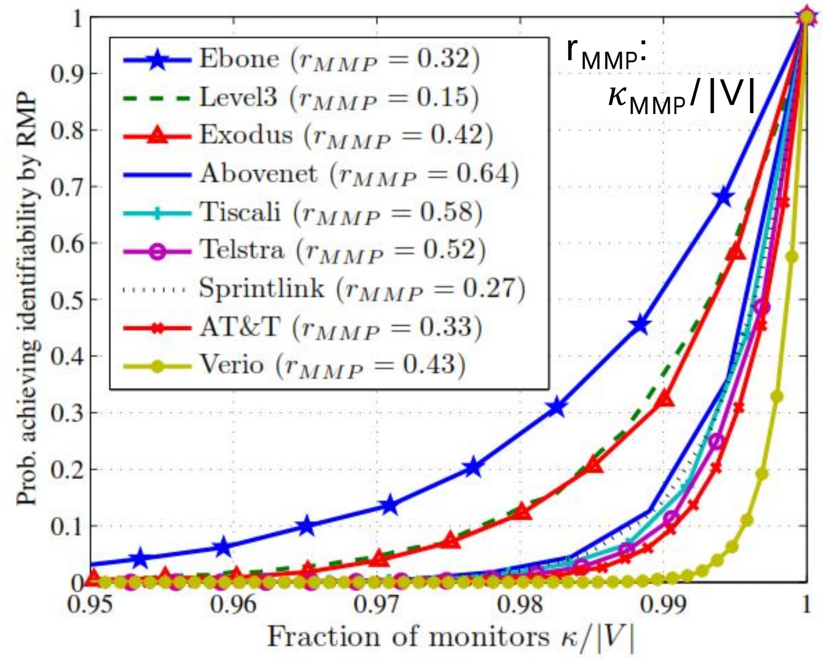
Densely-connected, |L| ≈ 450



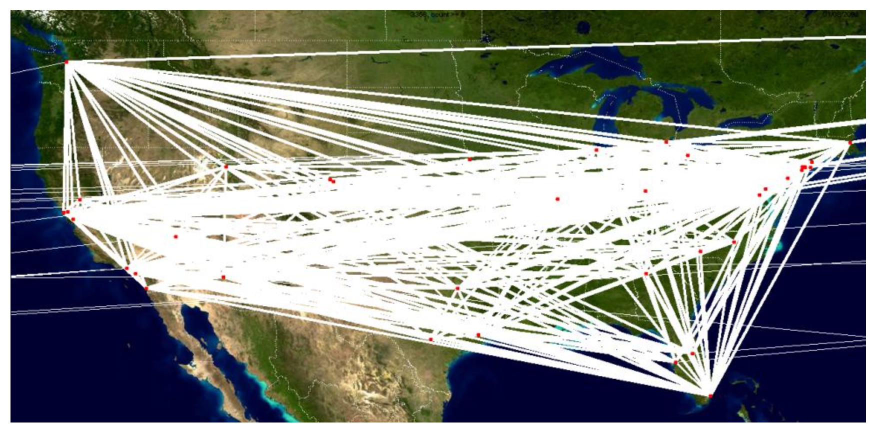
Sparsely-connected, |L| ≈ 295

Evaluation – Rocketfuel topologies

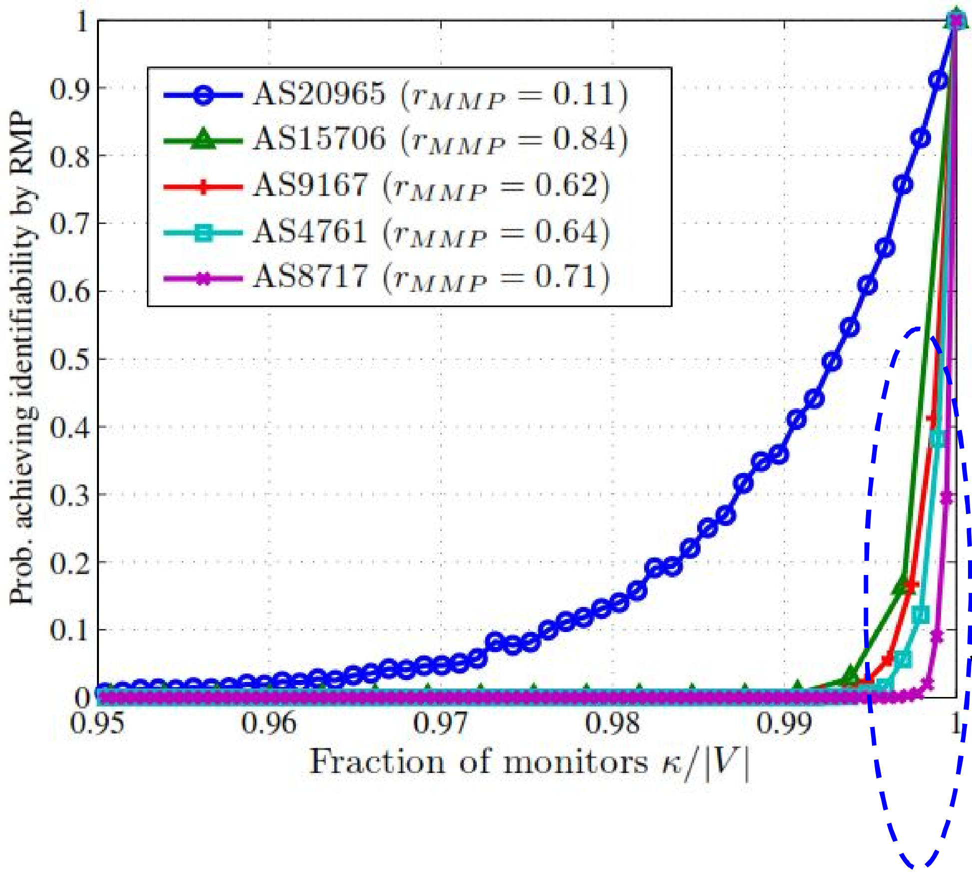
AS	ISP Name	$\lvert L \rvert$	$\lvert V \rvert$	κ_{MMP}
6461	Abovenet (US)	294	182	117
1755	Ebone (Europe)	381	172	55
3257	Tiscali (Europe)	404	240	138
3967	Exodus (US)	434	201	85
1221	Telstra (Australia)	758	318	164
7018	AT&T (US)	2078	631	208
1239	Sprintlink (US)	2268	604	163
2914	Verio (US)	2821	960	408
3356	Level3 (US)	5298	624	94



r_{MMP}: $\kappa_{\mathrm{MMP}}/|V|$



Evaluation – CAIDA topologies



AS	$\|L\|$	$\|V\|$	κ_{MMP}	r_{MMP}
15706	874	325	276	0.84
9167	1590	769	483	0.62
8717	3755	1778	1266	0.71
4761	3760	969	624	0.64
20965	8283	968	110	0.11

K_{MMP}: minimum #monitors computed by MMP

r_{MMP}: $K_{MMP}/|V|$

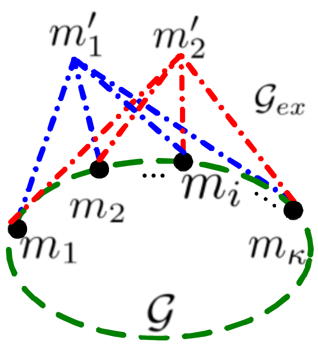
a) In most cases, RMP fails > 60% of time even if **all but one** node are monitors
b) MMP efficient in reducing #monitors

Path selection

Given minimum # monitors: how to select paths?

Construct extended graph G_{ex}

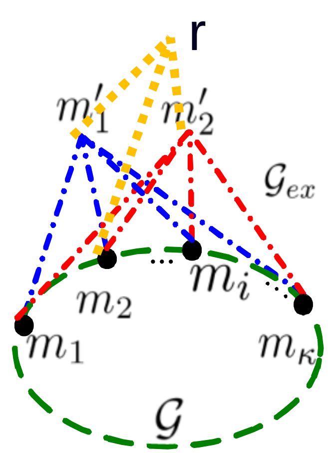
Construct r-extended graph G^*_{ex}



Extended graph G_{ex}

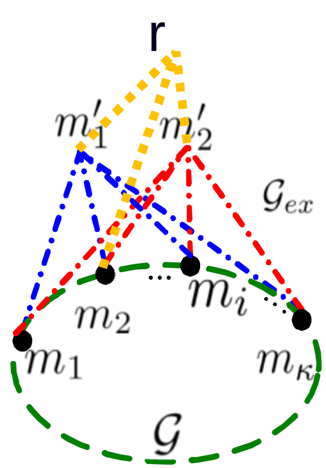
G_{ex} 3-vertex connected,

then G^*_{ex} also 3-vertex-connected



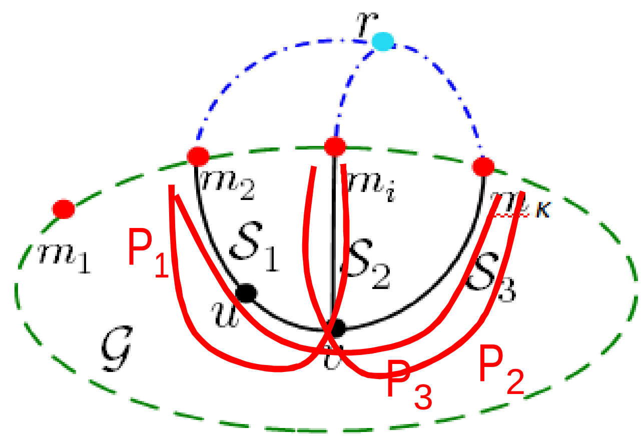
r-extended graph G^*_{ex}

Path selection (Cont'd)



r-extended graph G^*_{ex}

There exist 3 independent spanning trees wrt r (Cheriyan '88)



Each node in G^*_{ex} has 3 internally vertex disjoint paths to r, each along a spanning tree

Measurement paths:

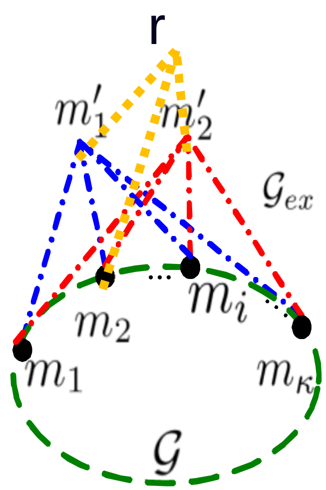
$P_1 = S_1 + S_2$

⇒

Intermediate results:

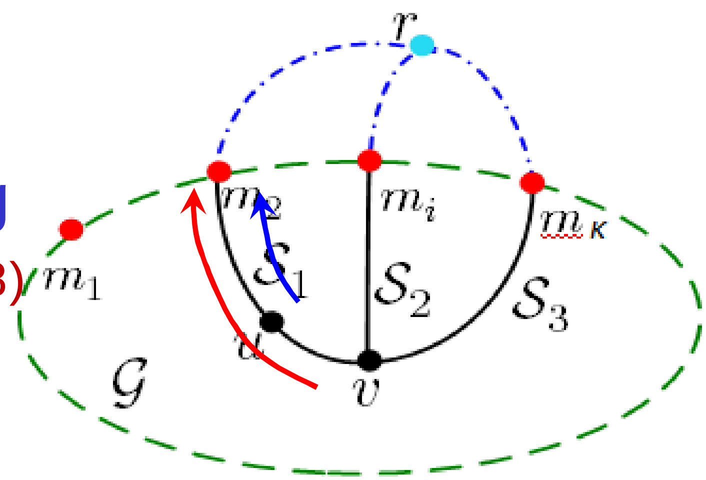
Metrics of S_1, S_2, S_3

Path selection (Cont'd)



r-extended graph G^*_{ex}

There exist 3 independent spanning trees wrt r (Cheriyan '88)



Each node in G^*_{ex} has 3 internally vertex disjoint paths to r, each along a spanning tree

Measurement paths:

$P_1 = S_1 + S_2$
$P_2 = S_2 + S_3$
$P_3 = S_3 + S_1$

⇒ Intermediate results:

Metrics of S_1, S_2, S_3

⇒ Link metrics:

$W_{uv} = W_{v \to m2} - W_{u \to m2}$

Can add paths to infer non-tree links

Evaluation – settings

Topologies

- random
  - Erdos-Renyi (ER), Random Geometric (RG), Barabasi-Albert (BA) graphs
- ISP topologies from Rocketfuel Project

Comparison of Spanning Tree Path Computation (SPTC) to Random Walk PC

(1) success rate of RWPC (simulation)

(2) average running time

(3) average path length

Platform

- Matlab R2010a on Laptop with Intel Core i5-2540M CPU @ 2.60GHz, 4GB memory, 64bit Win7 OS

Evaluation – ISP topologies

ISP	n	m	κ	r_{succ}	Υ	t_{STPC} (s)	t_{RWPC} (s)	t_{STLI} (ms)	t_{MILI} (ms)	h_{STPC}	h_{RWPC}
Abovenet	294	182	117	80.00%	99.61%	10.12	58.20	2.46	5.08	5.68	4.03
Ebone	381	172	55	75.00%	99.69%	13.65	139.37	3.78	11.06	9.61	7.00
Tiscali	404	240	138	70.00%	99.67%	28.07	171.58	3.81	10.71	7.05	4.89
Exodus	434	201	85	67.00%	99.76%	21.13	226.15	4.13	14.49	8.26	6.13
Telstra	758	318	164	24.00%	99.76%	80.38	2999.96	6.70	118.17	7.86	6.22
AT&T	2078	631	208	NA	NA	685.46	131.1 hrs	19.50	1302.85	23.48	11.33
Sprint	2268	604	163	NA	NA	608.18	46.8 hrs	20.52	1560.55	15.03	11.06
Verio	2821	960	408	NA	NA	697.86	170.3 hrs	29.15	3366.79	13.22	8.97

n: # edges
m: # nodes
κ: minimum #monitors to achieve identifiability

6x (Abovenet, Tiscali) to 879x (Verio) speedup for path construction

2x (Abovenet) to 115x (Verio) speedup for link identification

Need to probe longer paths

If impossible to identify all links?

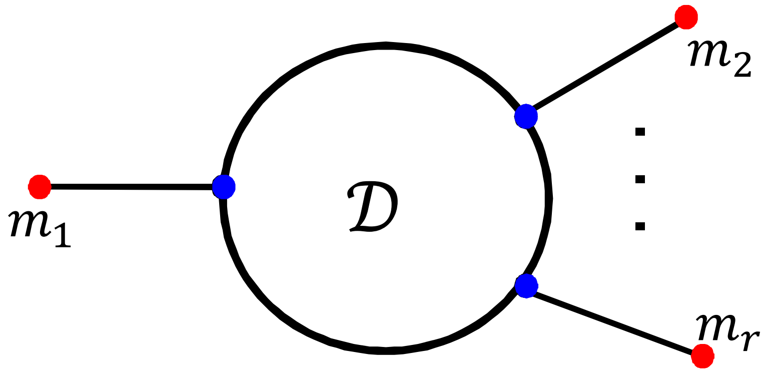
- determine maximum set of identifiable links
- for given monitor budget, maximize number of identified links

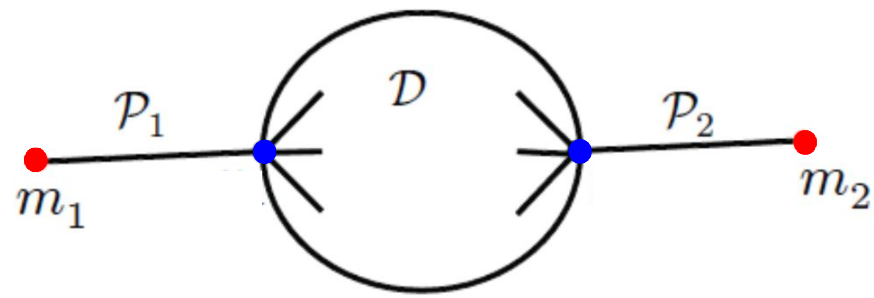
If impossible to identify all links?

- determine maximum set of identifiable links
- for given monitor budget, maximize number of identified links

Idea: use topological conditions for identifiability

- decompose graph into 3-vertex-connected subgraphs*



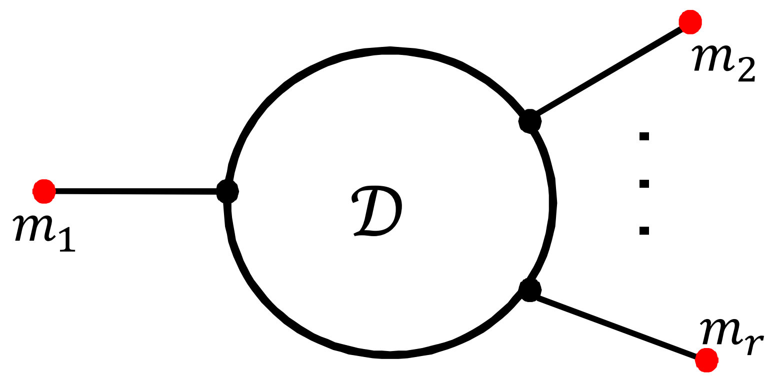


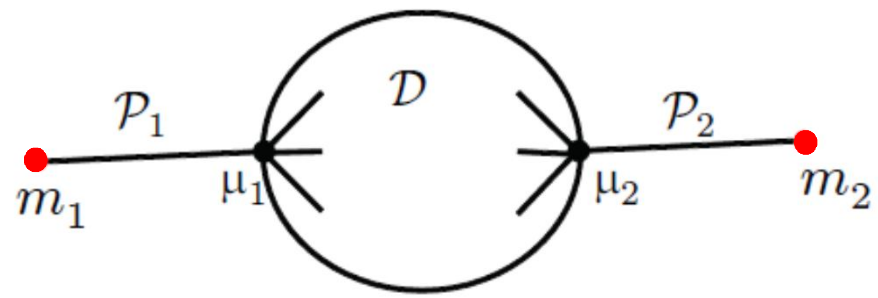
- "effective monitor"

* Hopcroft, Tarjan 73 - O(|V| + |L|)

Idea: use topological conditions for identifiability

- decompose graph into 3-vertex-connected subgraphs



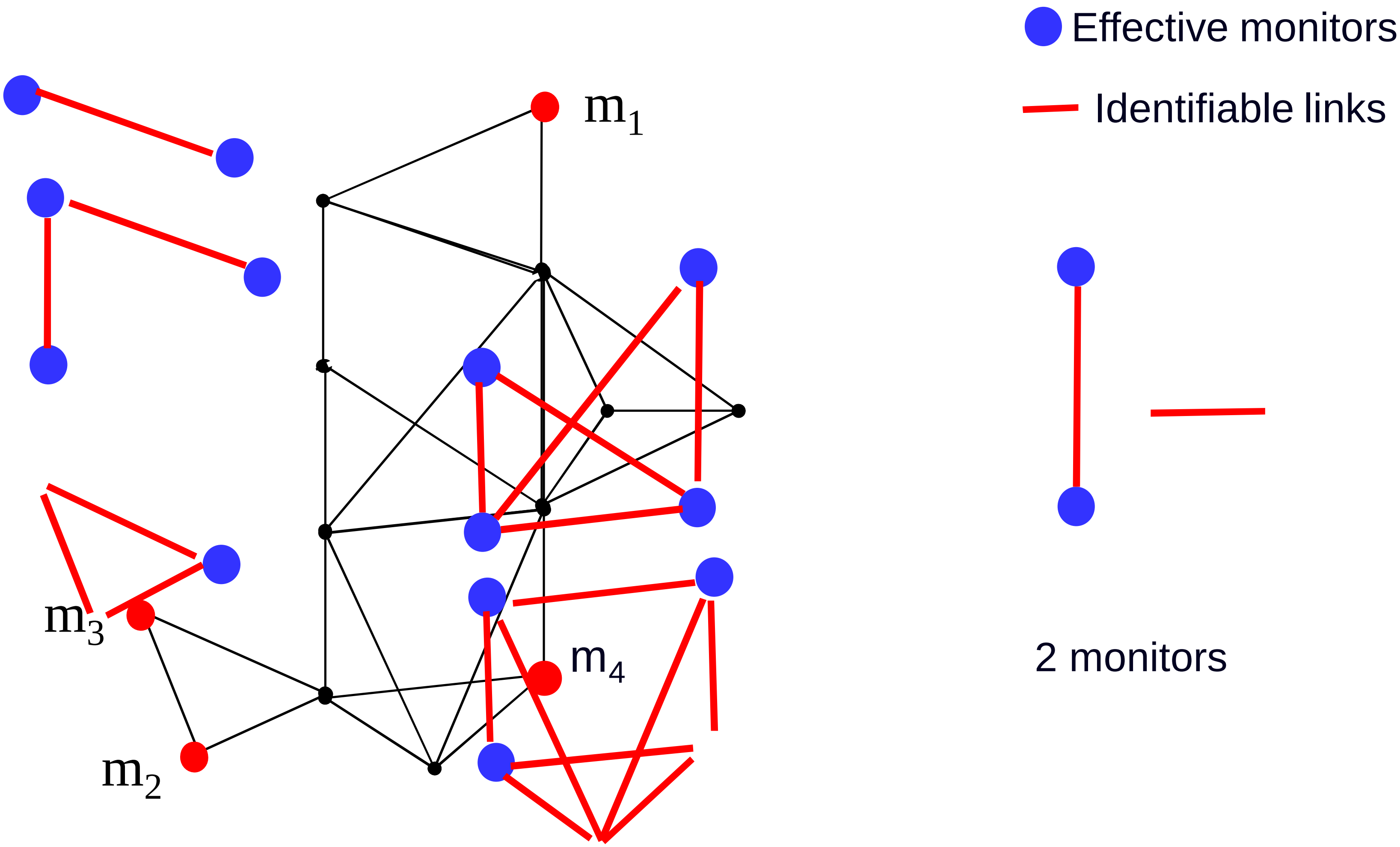


can identify all links

can identify all but those adjacent to μ_1, μ_2

Subgraph with one monitor cannot be identified

Determining identifiable links: Example



3 monitors

4+ monitors

Identification ratio:
18/24

Summary

- efficient algorithms to check identifiability
- efficient algorithm O(|V|+|L|) to place monitors optimally (MMP)
  - first known optimal placement algorithm
  - significantly reduces #monitors over random placement
- efficient algorithm for path construction
- efficient algorithm to check maximum partial identifiability

Current and future Work

- maximum link identifiability given monitor constraint (Infocom2014)
- fault localization
  - topological conditions for localizing single/k faults (IMC2014)
- account for noise in measurements → optimal experimental design
- mobile networks

Thanks!

Slides to be posted at:

http://www-net.cs.umass.edu/towsley/UFRJ-14.pdf