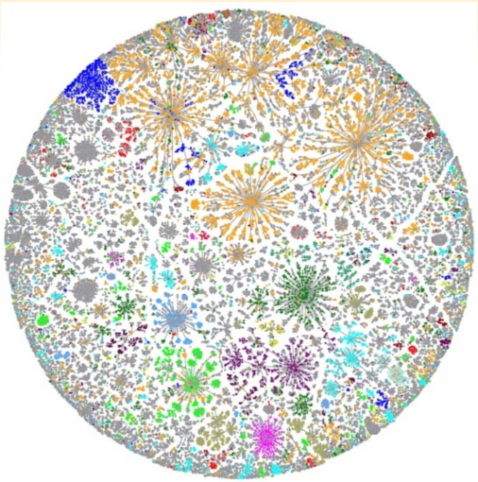
Optimal Connectivity on Big Graphs: Measures, Algorithms and Applications

Hanghang Tong

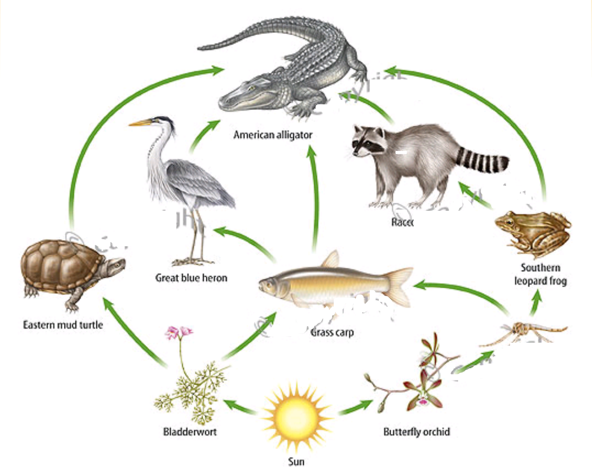
hanghang.tong@asu.edu

http://tonghanghang.org

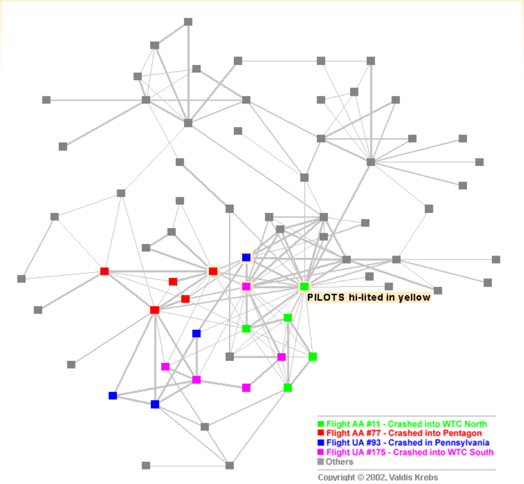
Observation: Graphs are everywhere!



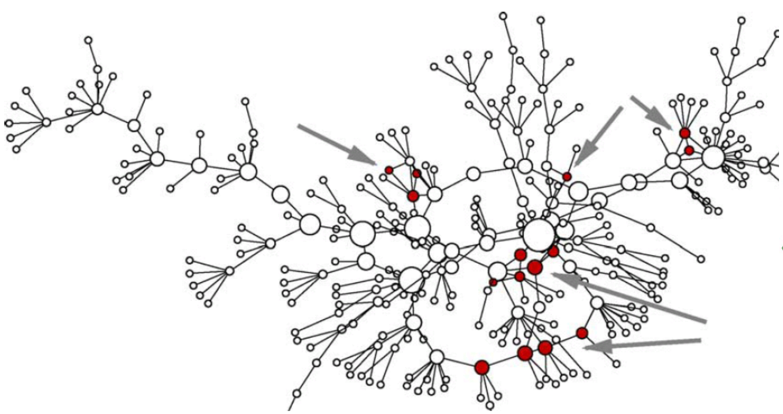
Internet Map [Koren 2009]

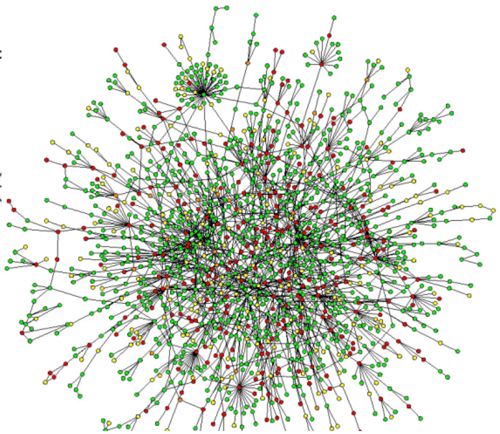


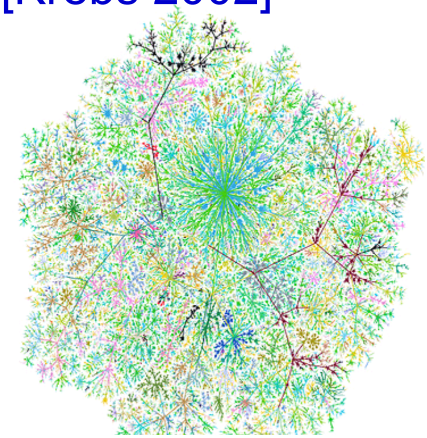
Food Web [2007]



Terrorist Network [Krebs 2002]



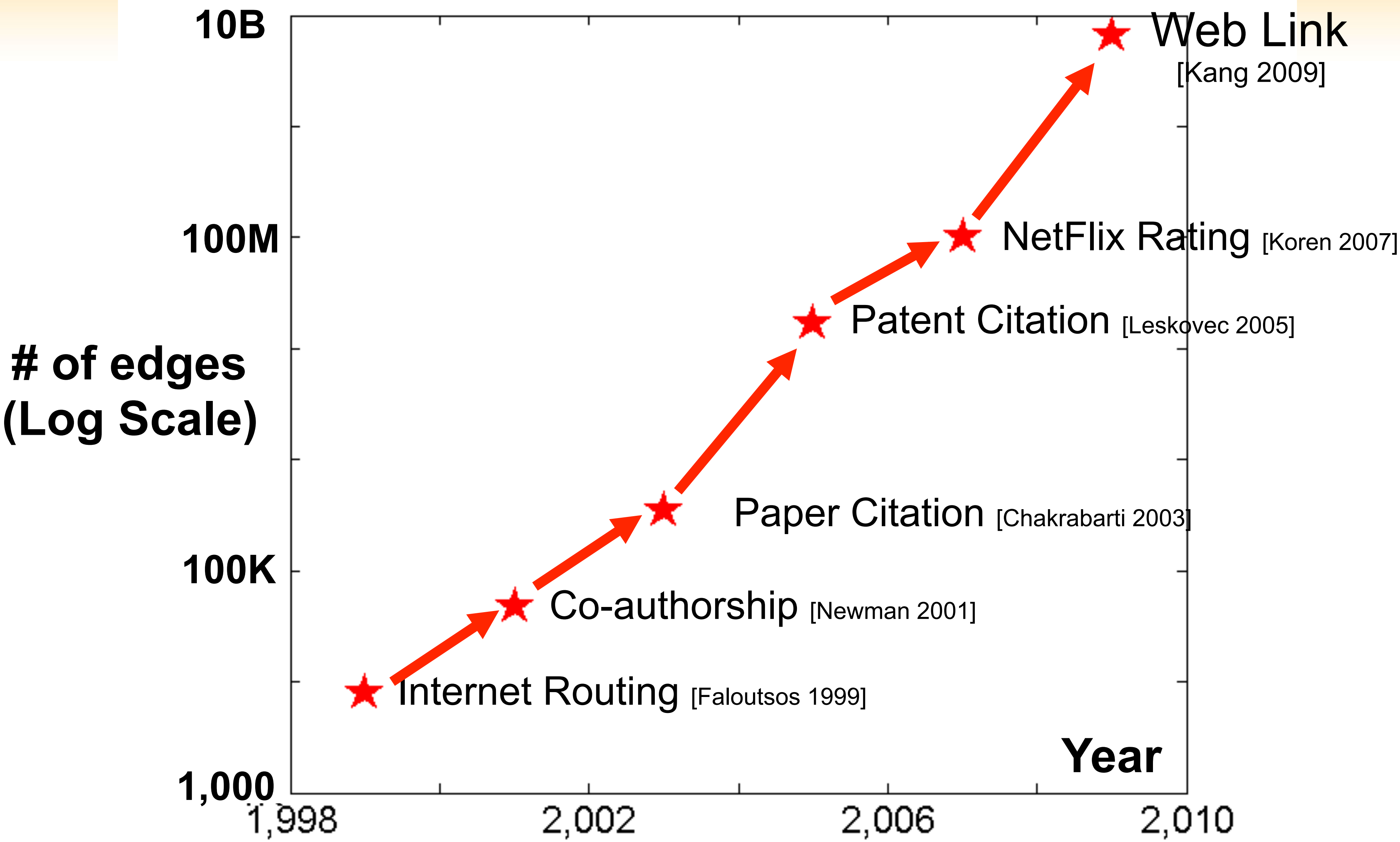




Goal: ***understand*** and ***utilize*** graph data

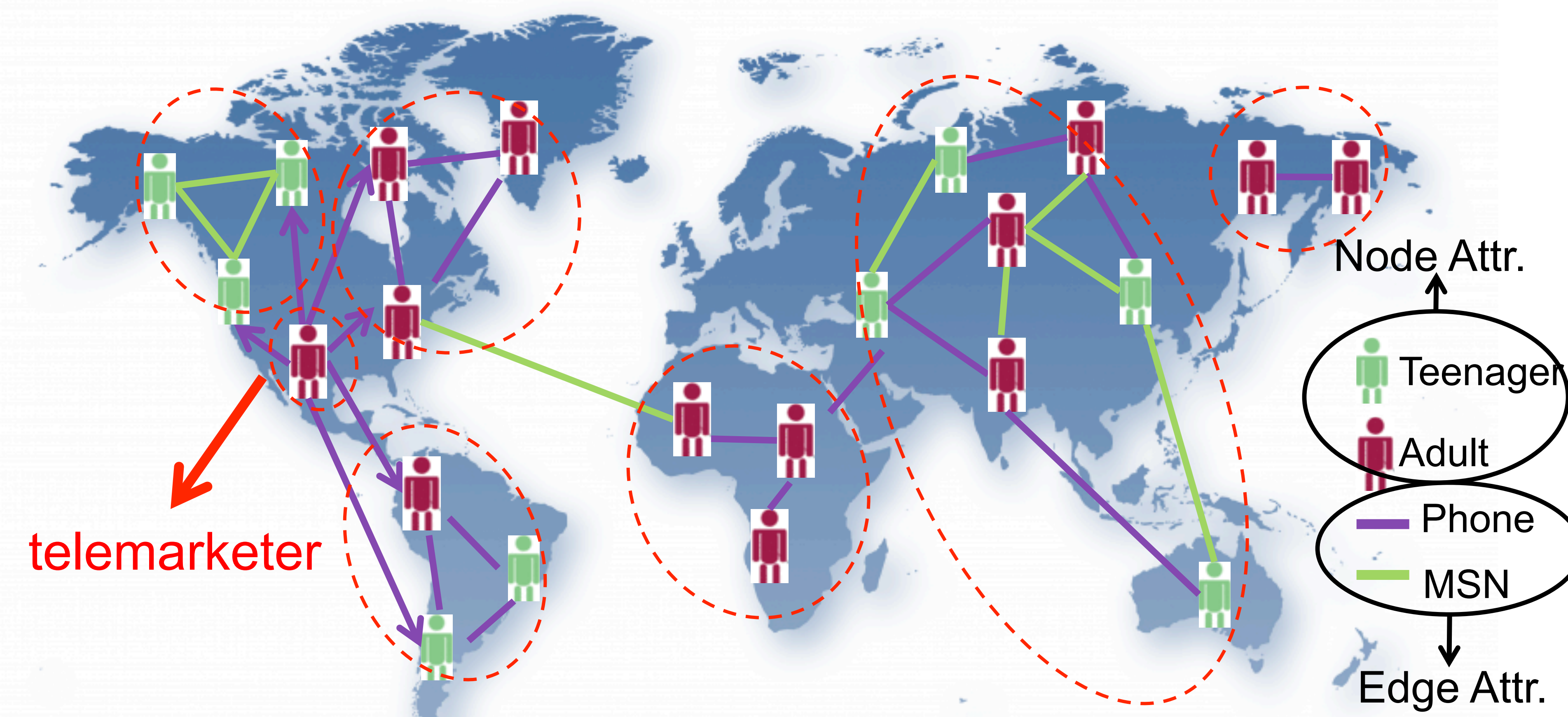
Challenges: real graphs are often BIG!

BIG Graphs #1: The Size of Graphs is Growing!



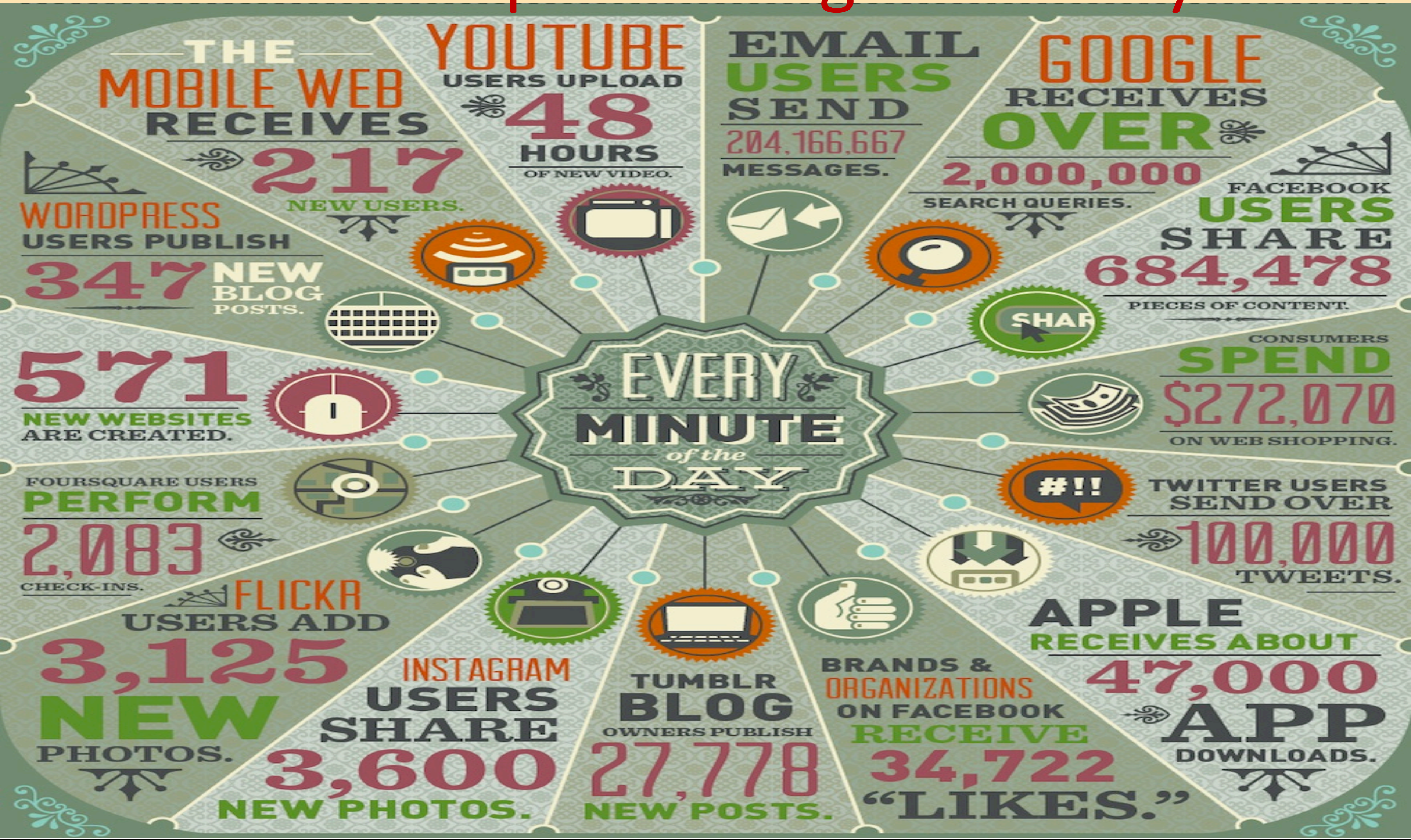
Q: How to Speed-up & Scale-up?

BIG Graphs #2: Data Complexity
(Rich graphs, e.g., geo-coded, attributed)



Q: What is difference between North America and Asia?;
How to find patterns? (e.g., anomalies, communities, etc)

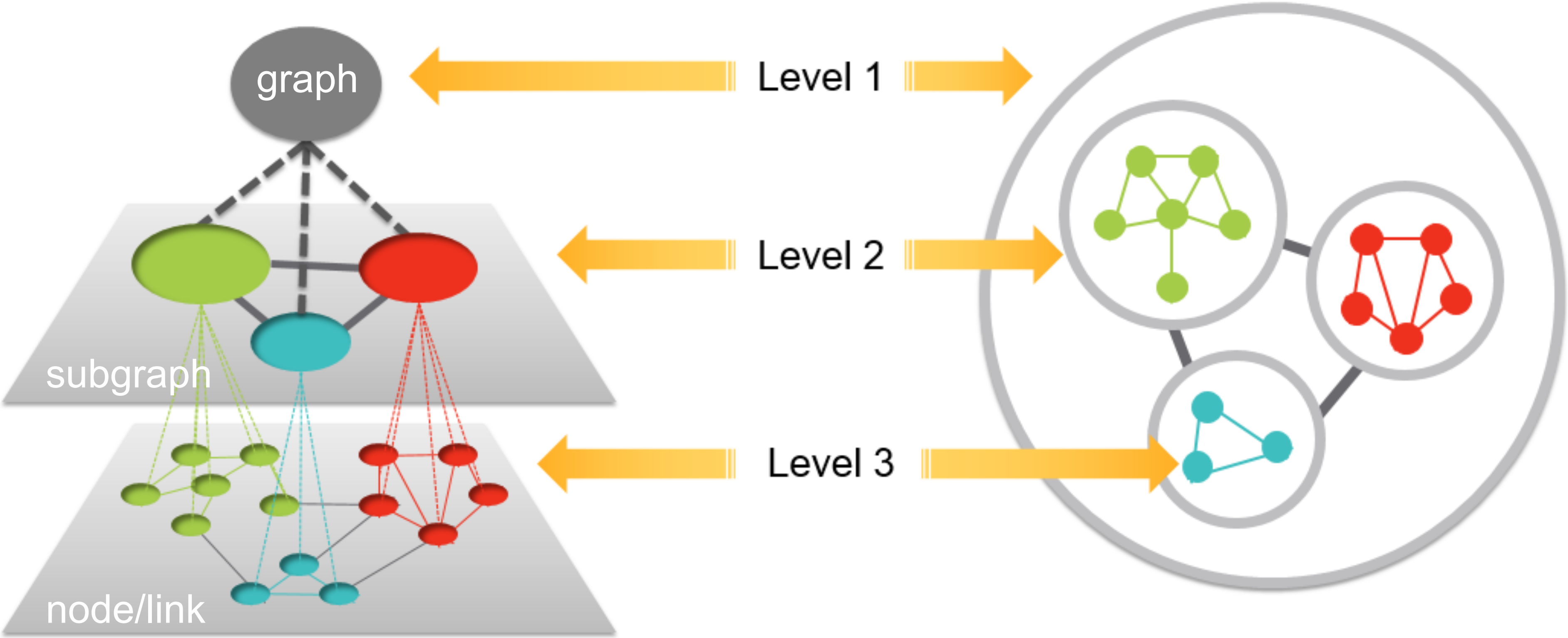
BIG Graphs #3: High Volatility



The amount of the data that is created every one minute

Q: How to respond in real-time or near-real time?

Graph Mining: An Overview



Q: Where does the graph come from?

A Typical Graph Mining Paradigm

Graphs → Patterns

Graph Connectivity Optimization (GCO) - This Tutorial

Given:
(1) an initial graph
(2) a graph operation
(3) a mining task

→

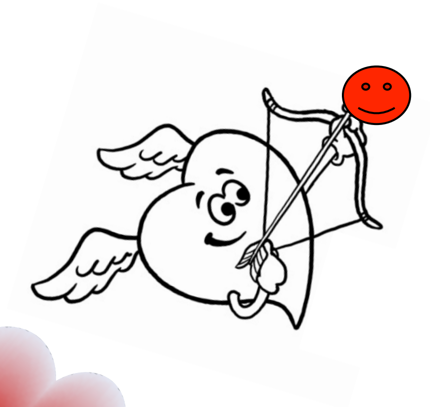
Find:
an 'optimal' graph

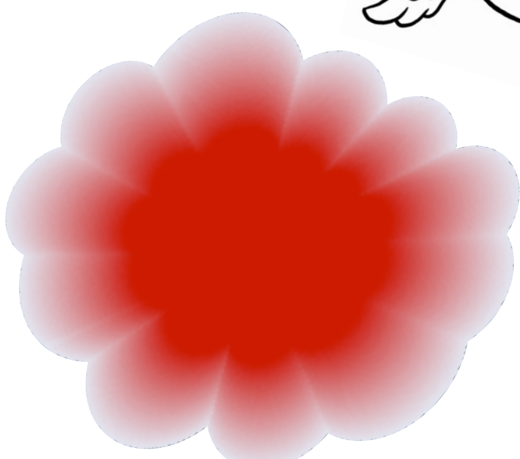
Graph operation: deleting 10 nodes; adding 5 links; etc.
Mining tasks: contain the virus; maximize the traffic flow

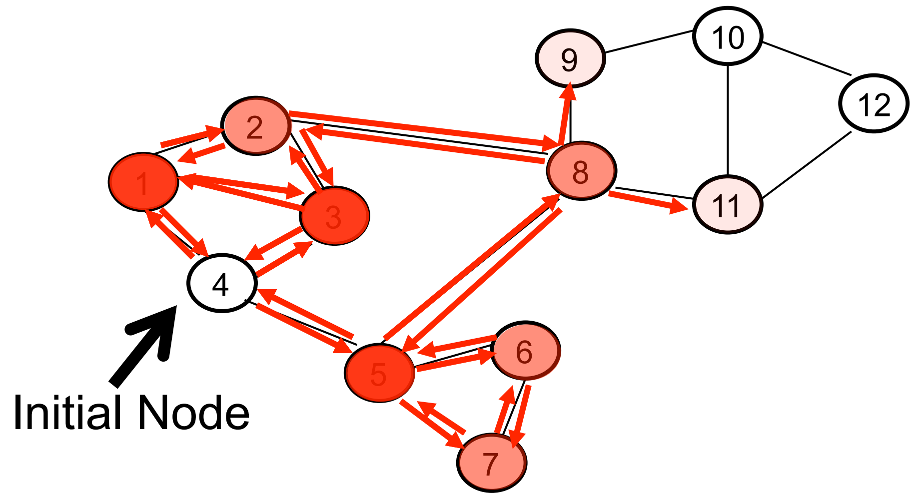
Dissemination: Think of it as Wine Spill

1. Spill a drop of wine on cloth
2. Spread/disseminate to the neighborhood

Dissemination: Wine Spill on a Graph



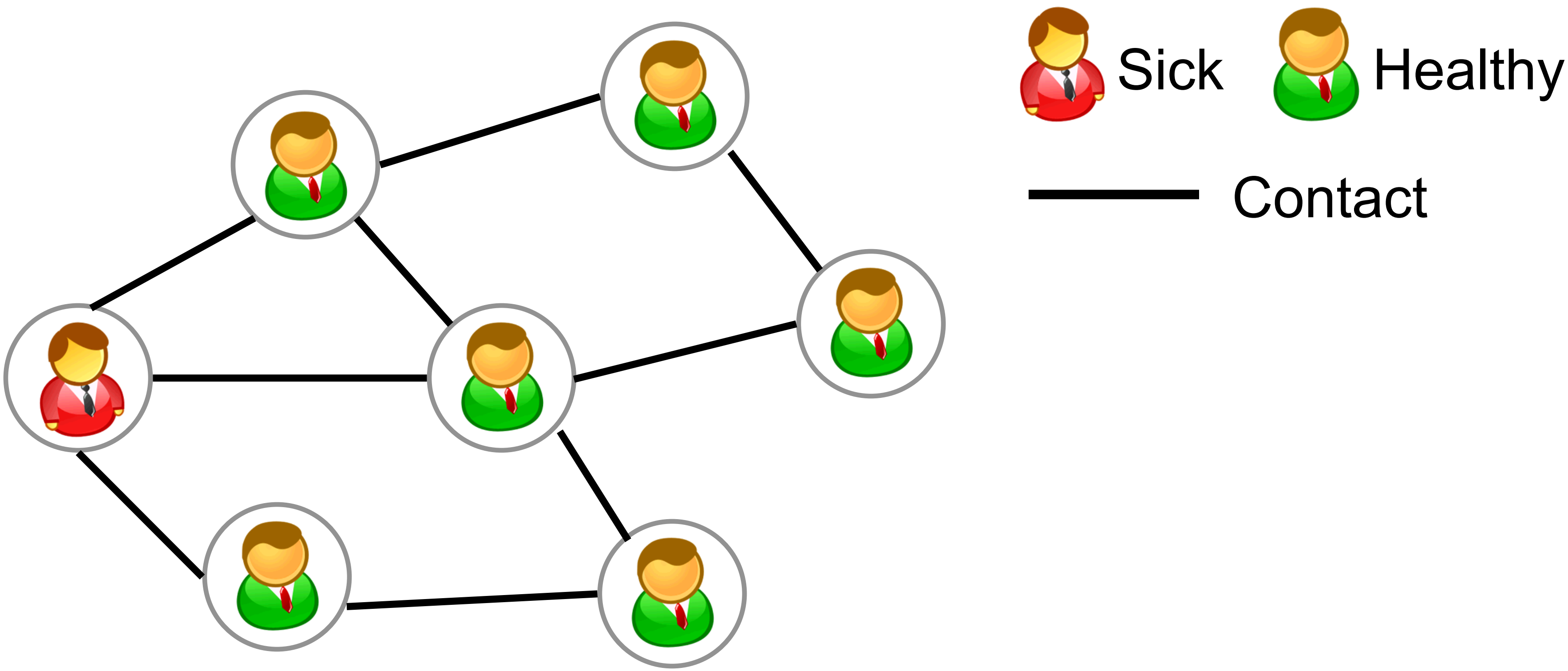




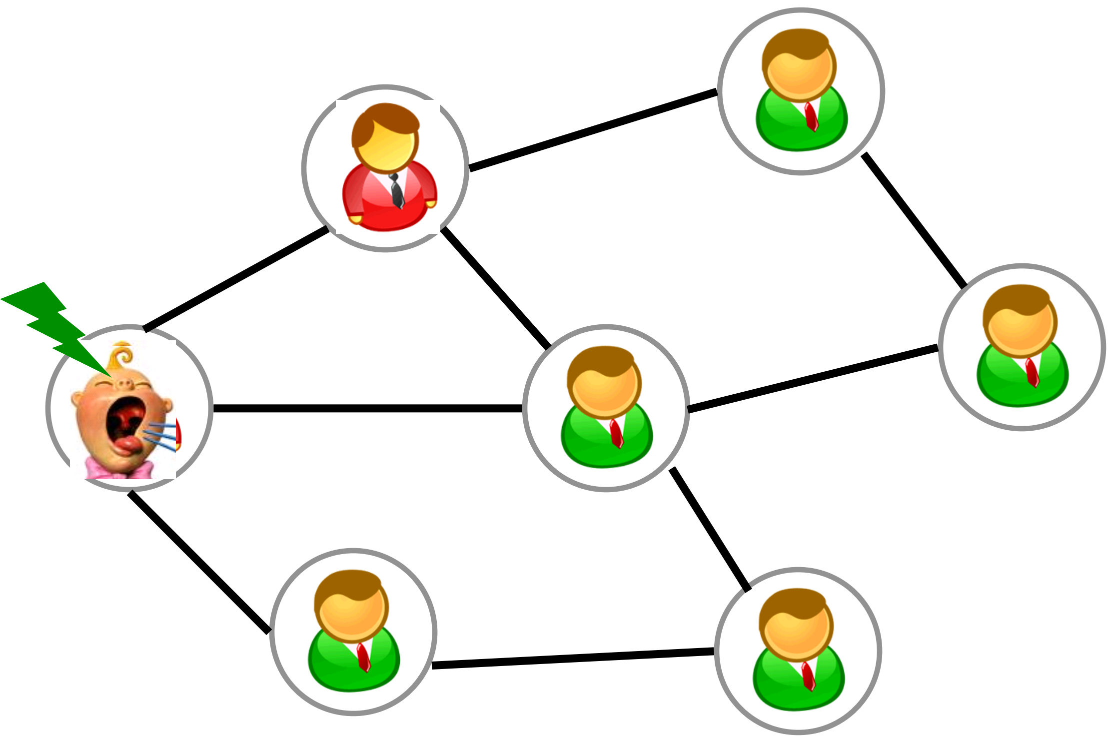
wine spill on cloth Dissemination on a graph

Same Diffusion Eq.

An Example: Virus Propagation/Dissemination



An Example: Virus Propagation/Dissemination



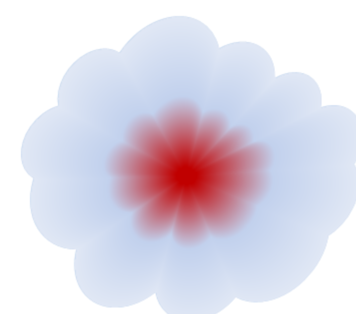
Sick Healthy

Contact

1: Sneeze to neighbors

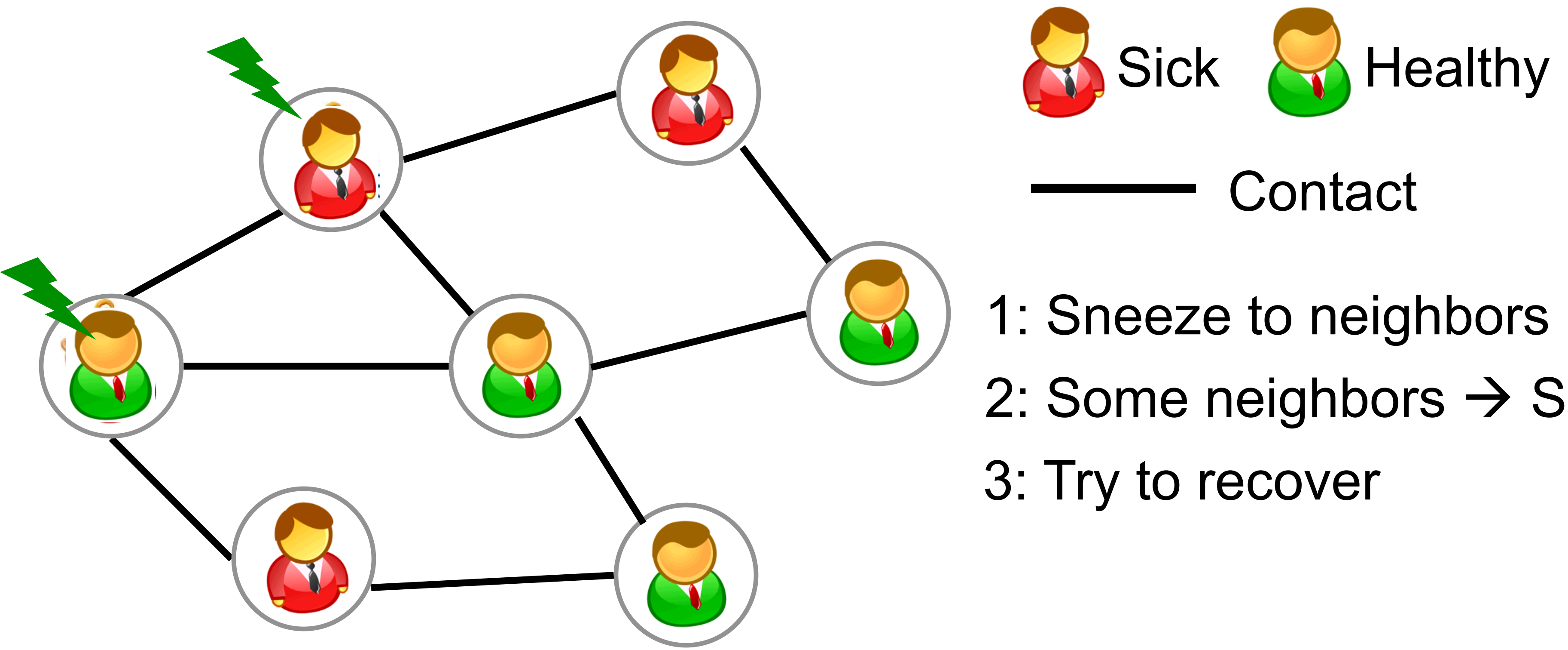
2: Some neighbors → Sick

3: Try to recover



Similar Diffusion Eq.

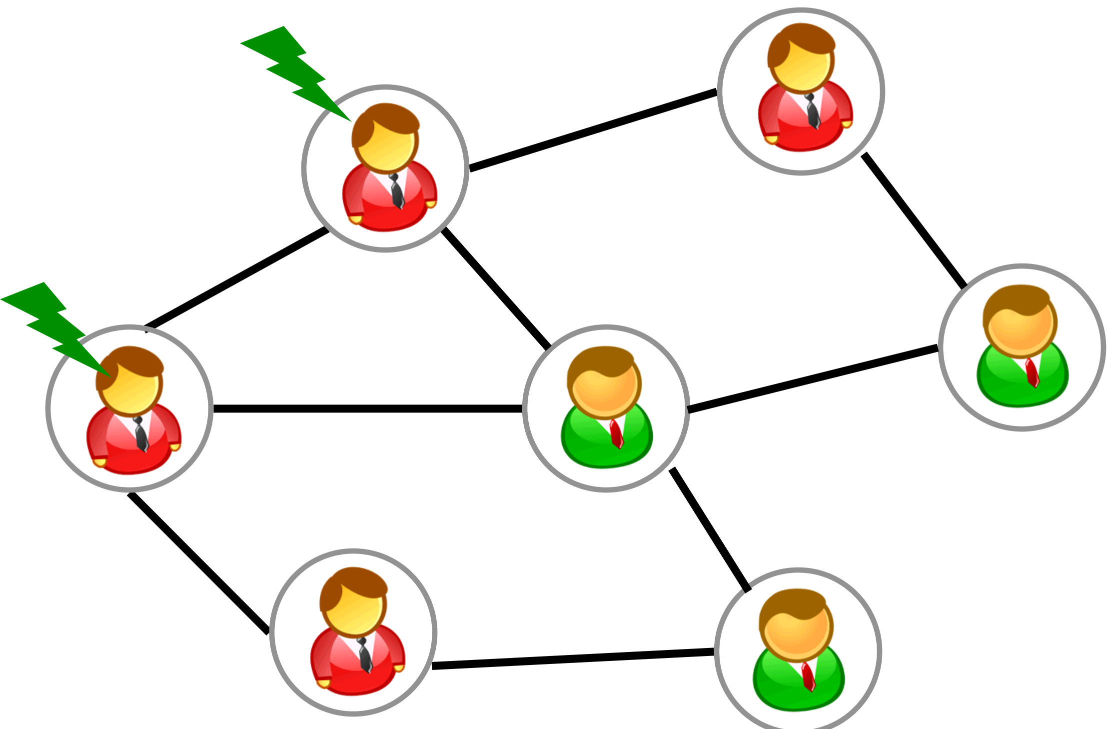
An Example: Virus Propagation/Dissemination



1: Sneeze to neighbors
2: Some neighbors → Sick
3: Try to recover

Q: How to minimize infected population?

An Example: Virus Propagation/Dissemination



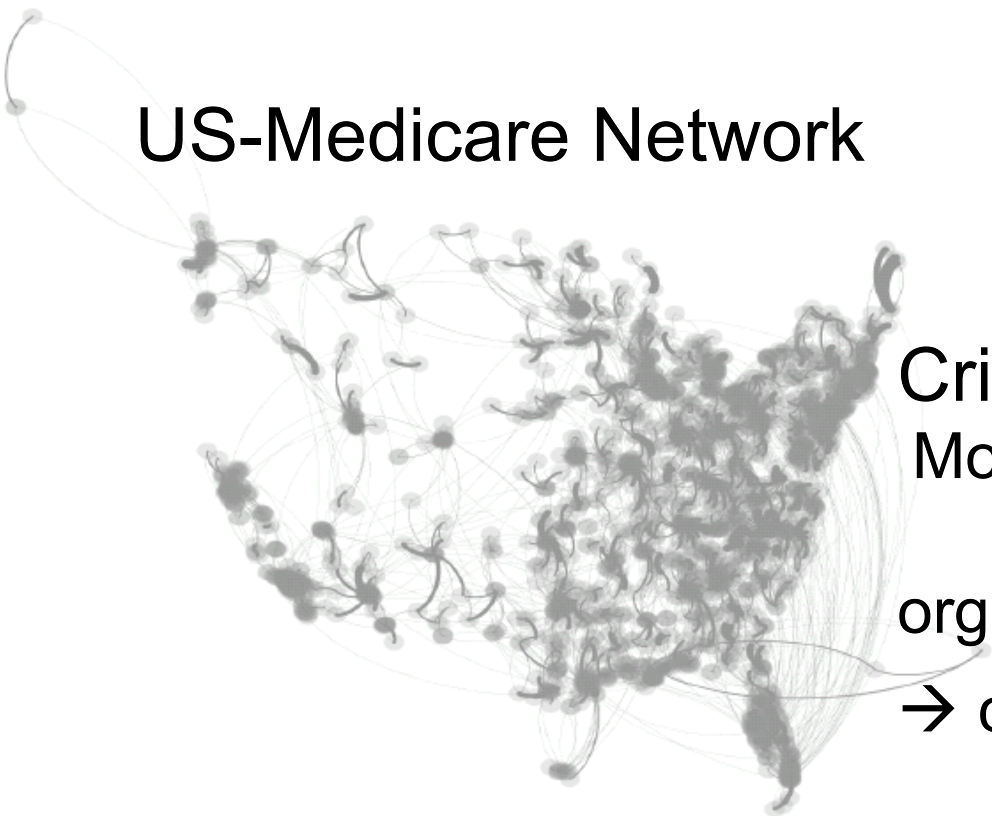
1: Sneeze to neighbors
2: Some neighbors → Sick
3: Try to recover

Q: How to minimize infected population?

- Q1: Understand tipping point
- Q2: Affecting algorithms

Why Do We Care? – Healthcare

US-Medicare Network



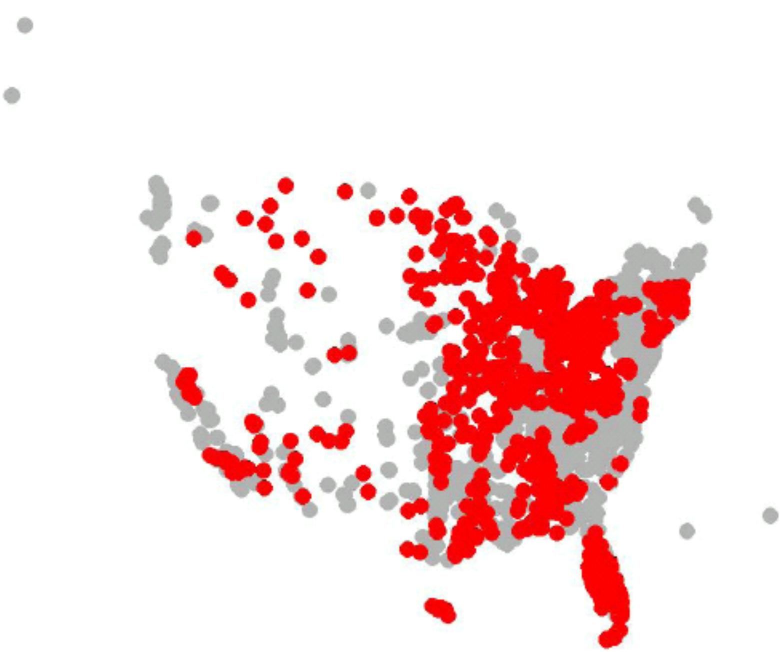
Critical Patient transferring

Move patients → specialized care → highly resistant micro-organism → Infection controlling → costly & limited

Q: How to allocate resource to minimize overall spreading?

SARS costs 700+ lives; $40+ Bn; H1N1 costs Mexico $2.3bn; Flu 2013: one of the worst in a decade, 105 children in US.

Why Do We Care? – Healthcare



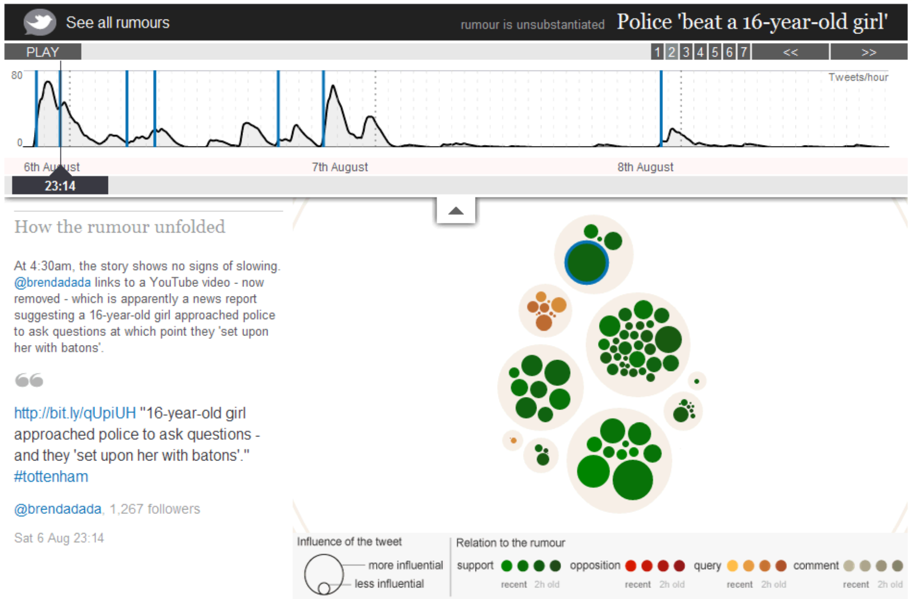
Current Method

Our Method

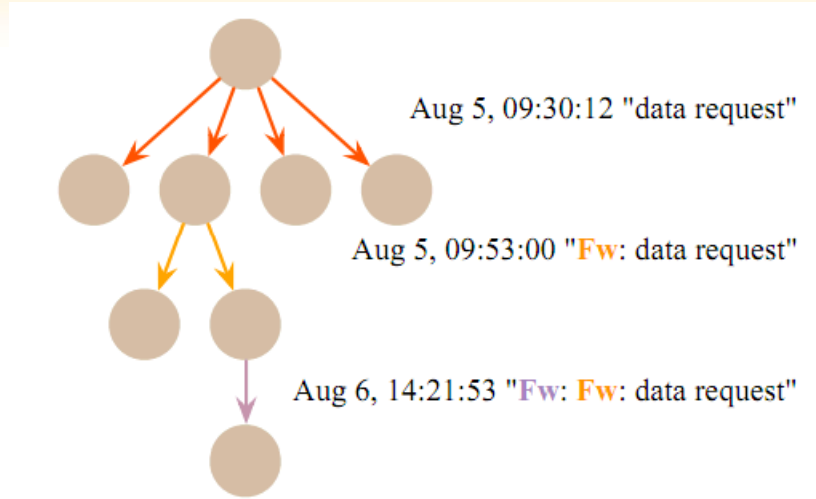
Red: Infected Hospitals after 365 days

B. Aditya Prakash, Lada Adamic, Theodore Iwashnya, Hanghang Tong and Christos Faloutsos: Fractional Immunization on Networks. SDM 2013

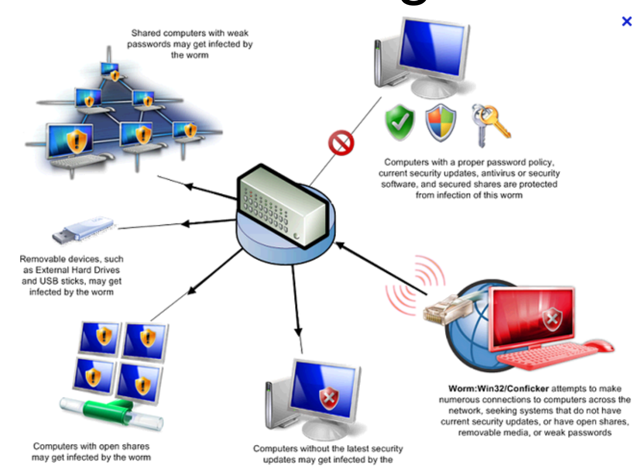
Why Do We Care? (More)



Rumor Propagation



Email Fwd in Organization



Malware Infection

Viral Marketing

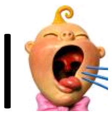
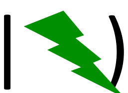
Roadmap

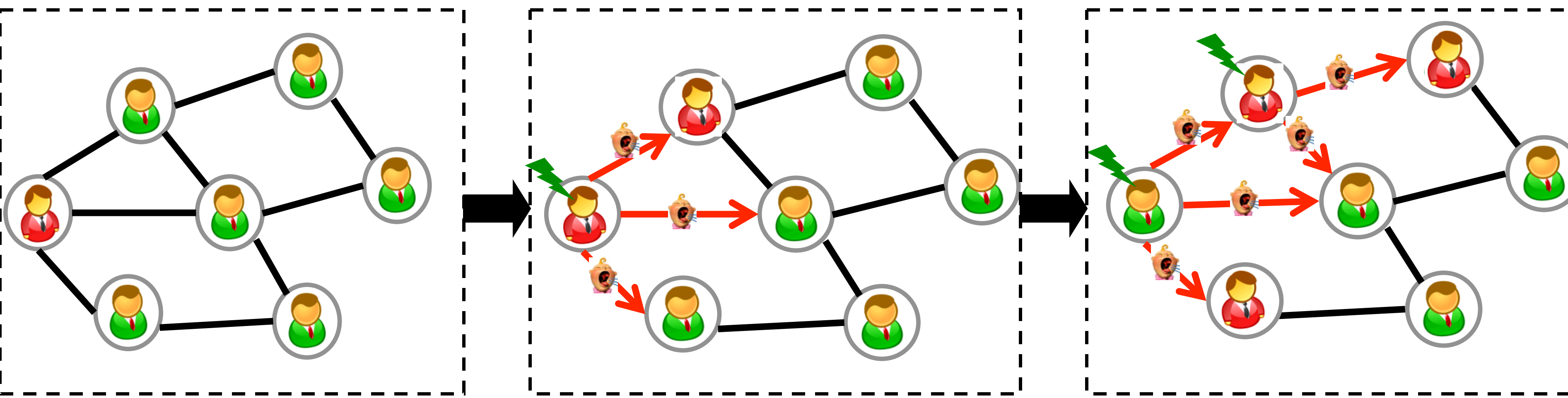
- Motivations and Background
- Part I: GCO Measures
- Part II: GCO Theories & Algorithms
- Part III: GCO Applications
- Part IV: Open Challenges & Future Trends

Part I: GCO Measures

- GCO Measure #1: Epidemic Threshold (***λ)***
- GCO Measure #2: Graph Robustness
- Other GCO Measures
- Comparison of GCO Measures
- Unification of GCO Measures

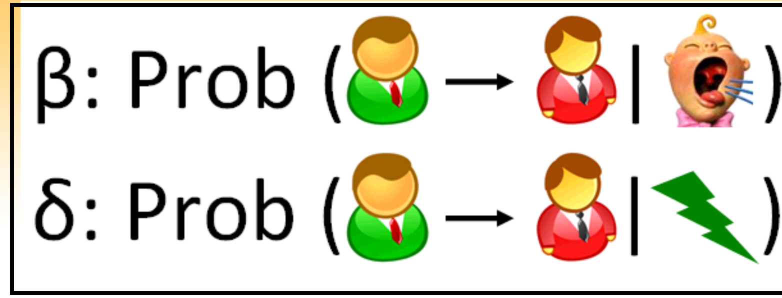
SIS Model (e.g., Flu) (Susceptible-Infected-Susceptible)

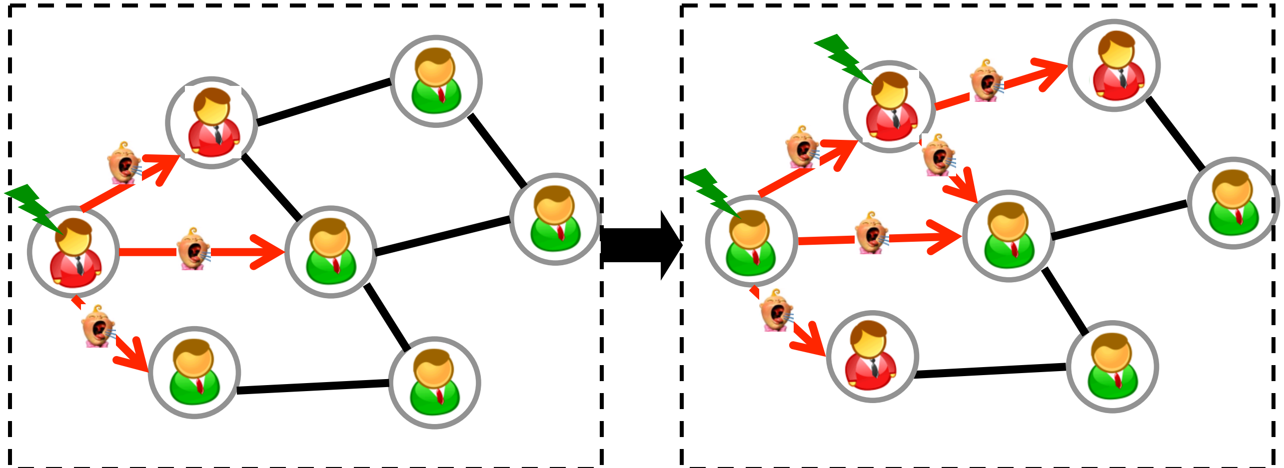
- Each Node Has Two Statuses:  Sick  Healthy
- β: Infection Rate (Prob ( →  | ))
- δ: Recovery Rate (Prob ( →  | ))



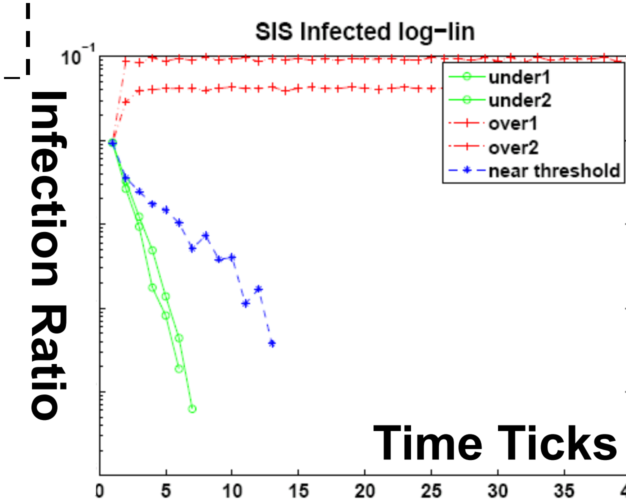
t = 1 *t* = 2 *t* = 3

SIS Model (e.g., Flu)





$$p_{t+1} = H(p_t)$$



Theorem [Chakrabarti+ 2003, 2007]:

If $\lambda \times (\beta/\delta) \leq 1$; no epidemic

for any initial conditions

λ: largest eigenvalue of the graph (~ connectivity of the graph)

β, δ : virus parameters (~strength of the virus)

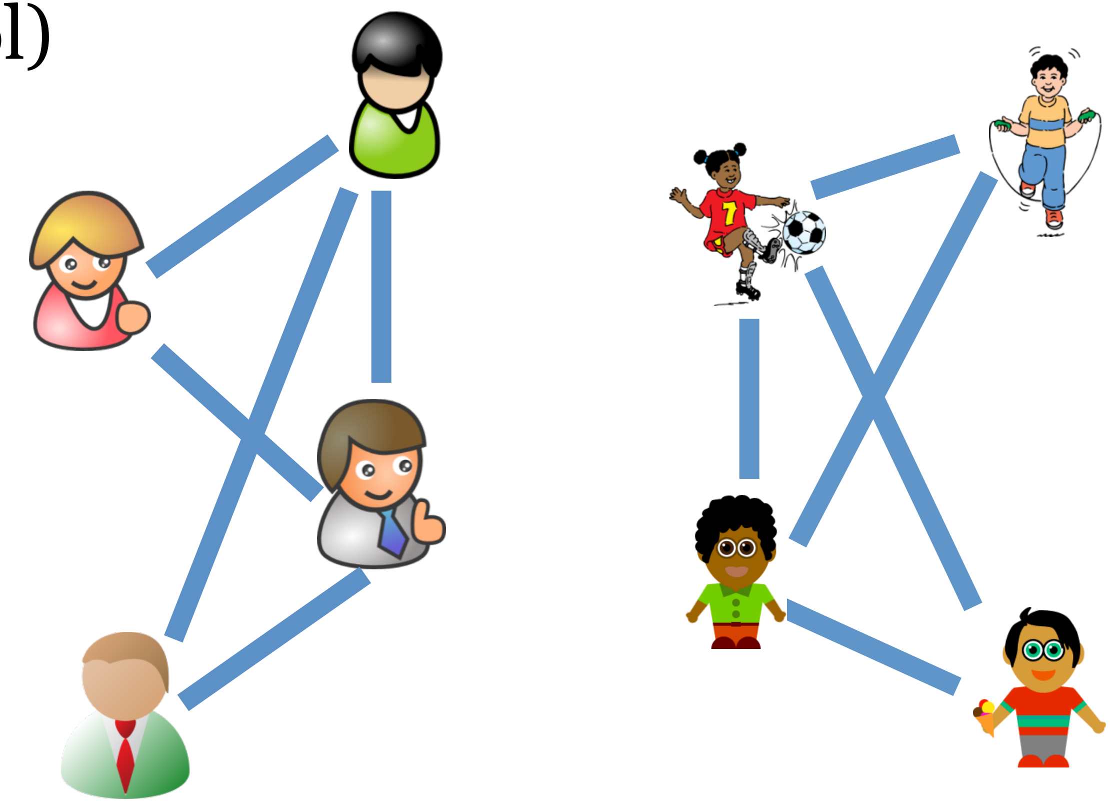
Beyond Static Graphs: Alternating Behavior

DAY
(e.g., work, school)

A_1: adjacency matrix

8

8



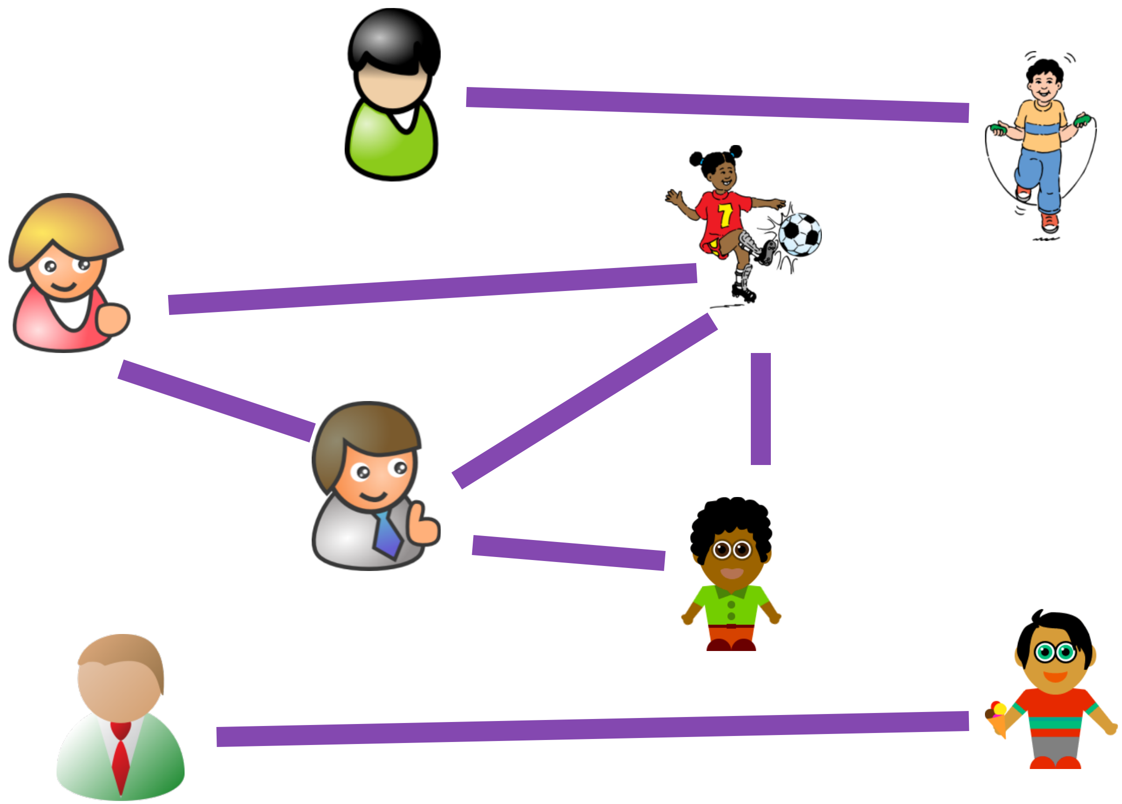
B. Aditya Prakash, Hanghang Tong, Nicholas Valler, Michalis Faloutsos, Christos Faloutsos:Virus Propagation on Time-Varying Networks: Theory and Immunization Algorithms. ECML/PKDD (3) 2010: 99-114
Nicholas Valler, B. Aditya Prakash, Hanghang Tong, Michalis Faloutsos, Christos Faloutsos:Epidemic Spread in Mobile Ad Hoc Networks: Determining the Tipping Point. Networking (1) 2011: 266-280

Beyond Static Graphs: Alternating Behavior

NIGHT
(e.g., home)

A_2: adjacency matrix

$\underbrace{\left.\boxed{A_2}\right\}8}_{8}$

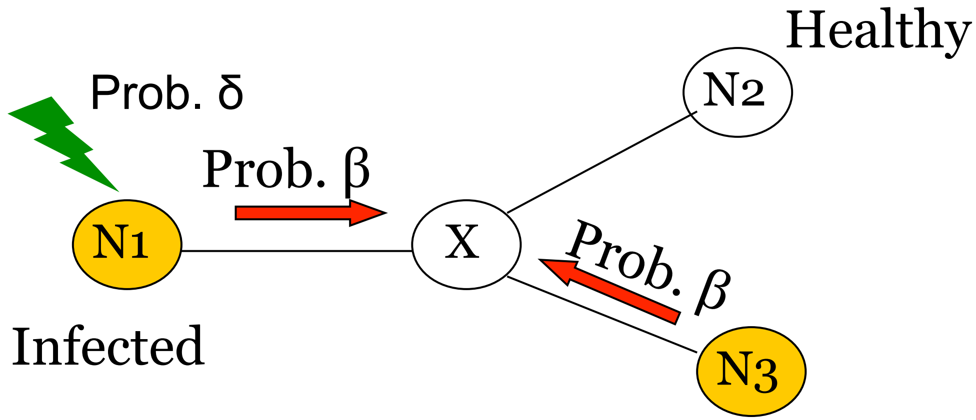


B. Aditya Prakash, Hanghang Tong, Nicholas Valler, Michalis Faloutsos, Christos Faloutsos:Virus Propagation on Time-Varying Networks: Theory and Immunization Algorithms. ECML/PKDD (3) 2010: 99-114
Nicholas Valler, B. Aditya Prakash, Hanghang Tong, Michalis Faloutsos, Christos Faloutsos:Epidemic Spread in Mobile Ad Hoc Networks: Determining the Tipping Point. Networking (1) 2011: 266-280

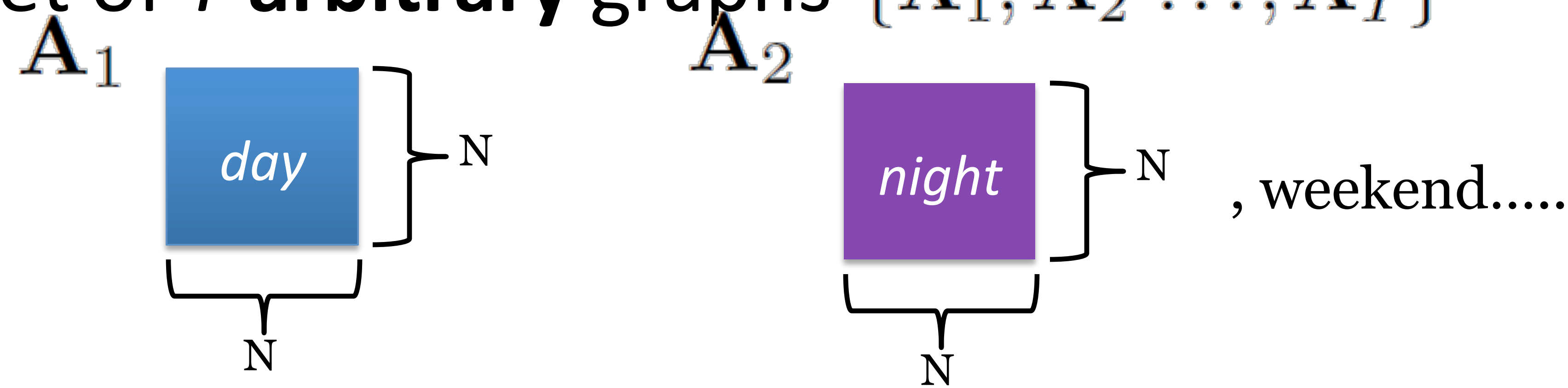
Formal Model Description

[PKDD 2010, Networking 2011]

- SIS model
  - recovery rate δ
  - infection rate β



- Set of T **arbitrary** graphs $\{\mathbf{A}_1, \mathbf{A}_2, \ldots, \mathbf{A}_T\}$



Epidemic Threshold for Alternating Behavior

[PKDD 2010, Networking 2011]

Theorem [PKDD 2010, Networking 2011]:
No epidemic If $\lambda(S) \leq 1$.

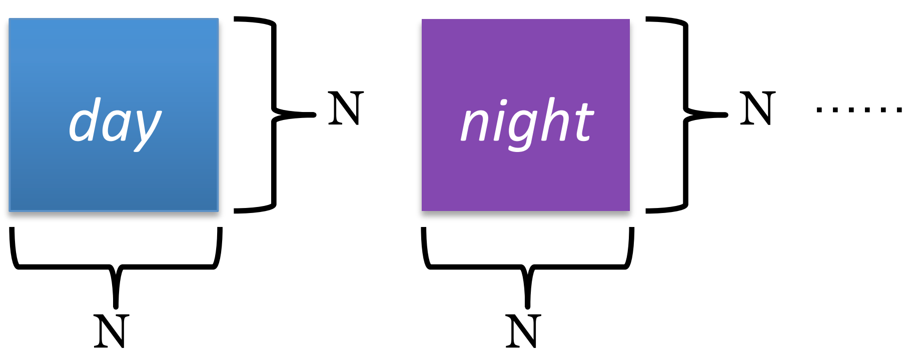
System matrix $S = \Pi_i S_i$

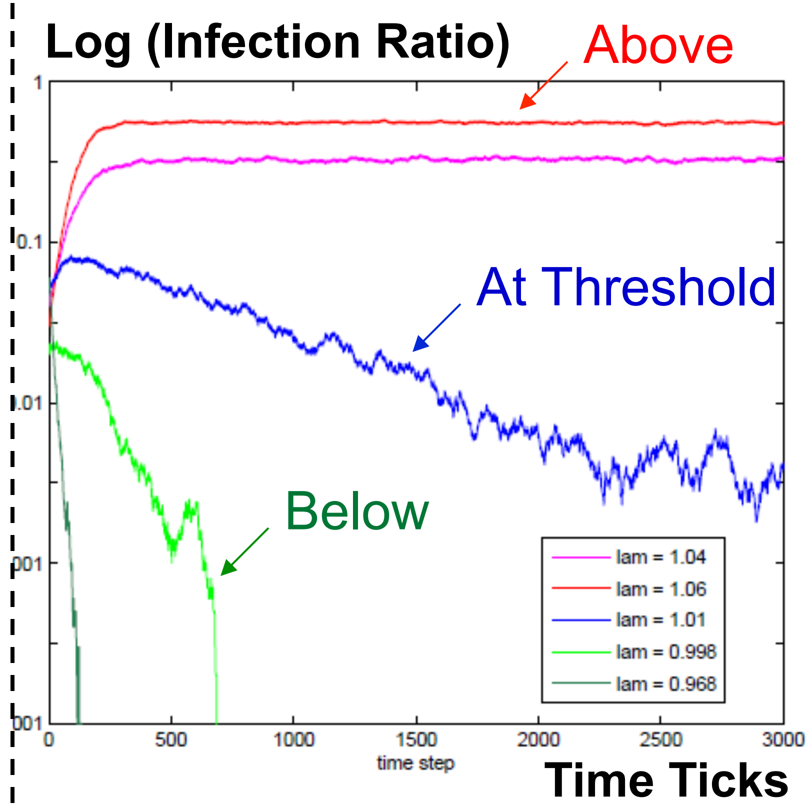
$S_i = (1-\delta)I + \beta A_i$

β: Prob (→ |)

δ: Prob (→ |)

A_i



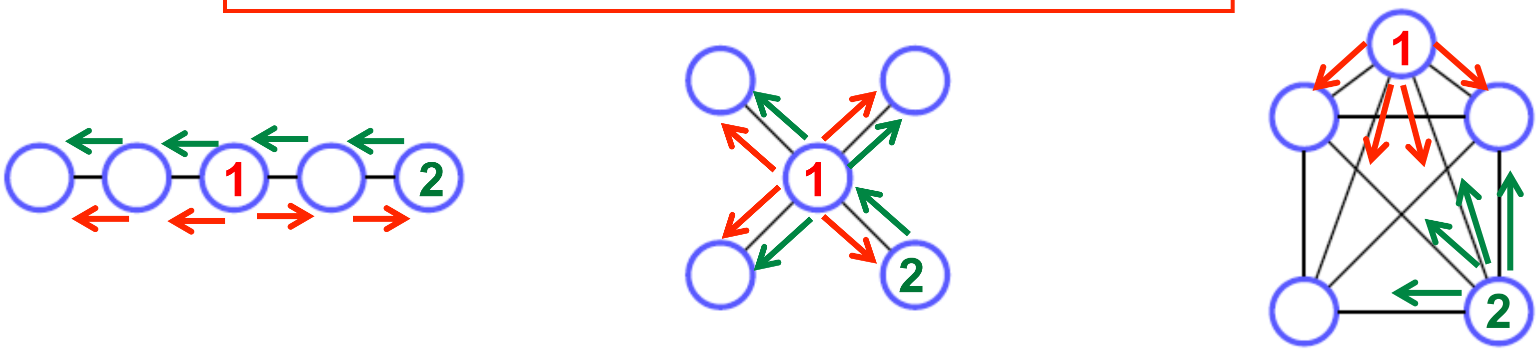


Also generalize to other 25 virus propagation models

Why is $\boldsymbol{\lambda}$ So Important?

- $\boldsymbol{\lambda}$ → **Path Capacity** of a Graph:

$$\boxed{\left(\vec{1}^{*} A^{k} \vec{1}\right)^{1/k} \xrightarrow[k \to \infty]{} \lambda}$$

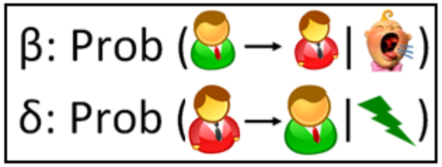


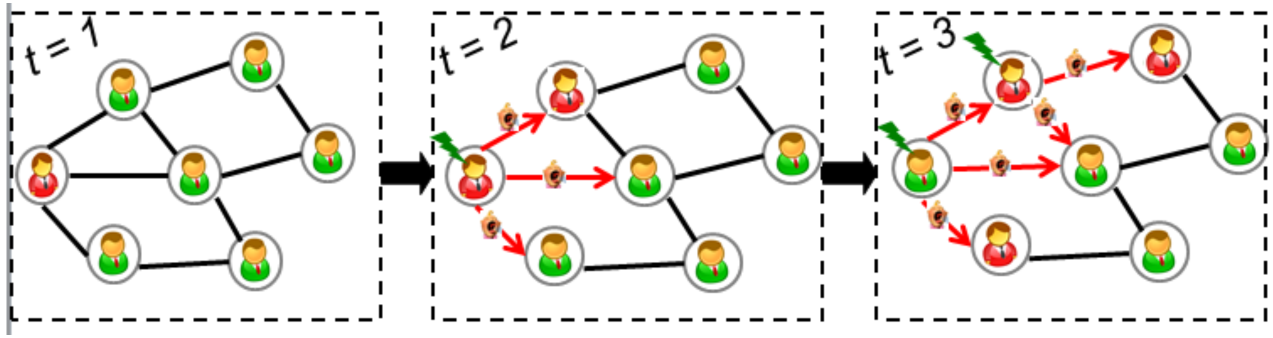
(a)Chain($\lambda_1 = 1.73$) (b)Star($\lambda_1 = 2$) (c)Clique($\lambda_1 = 4$)

Larger λ → better connected

Why is λ So Important?

- **Key 1: Model Dissemination as an NLDS:**





$$p_{t+1} = g\,(p_t)$$

p_t : Prob. vector: nodes being sick at t

g : Non-linear function (graph + virus parameters)

- **Key 2: Asymptotic Stability of NLDS:**

$p = p^* = 0$ is asymptotic stable if $|\lambda\,(J)|<1$, where

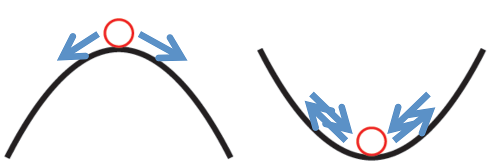
$$J_{k,l} = [\nabla g(\mathbf{p}^*)]_{k,l} = \frac{\partial p_{k,t+1}}{\partial p_{l,t}}\Big|_{\mathbf{p}_t=\mathbf{p}^*}$$

$$\frac{\partial \mathbf{p}_{2t+2}}{\partial \mathbf{p}_{2t+1}}\Big|_{\mathbf{p}_{2t+1}=\mathbf{0}} = (1-\delta)\mathbf{I} + \beta\mathbf{A}_1 = \mathbf{S}_1$$

$$\frac{\partial \mathbf{p}_{2t+1}}{\partial \mathbf{p}_{2t}}\Big|_{\mathbf{p}_{2t}=\mathbf{0}} = (1-\delta)\mathbf{I} + \beta\mathbf{A}_2 = \mathbf{S}_2$$

$$p_{i,2t+1} = 1 - \delta p_{i,2t} - (1-p_{i,2t})\zeta_{2t}(i)$$

$$p_{i,2t+2} = 1 - \delta p_{i,2t+1} - (1-p_{i,2t+1})\zeta_{2t+1}(i)$$



(A) Unstable (B) Stable

$$\zeta_{2t}(i) = \prod_{j\in\mathcal{NE}_2(i)} (p_{j,2t}(1-\beta) + (1-p_{j,2t})) = \prod_{j\in\{1..n\}} (1-\beta\mathbf{A}_2(i,j)p_{j,2t}))$$

$$\zeta_{2t+1}(i) = \prod_{j\in\mathcal{NE}_1(i)} (p_{j,2t+1}(1-\beta) + (1-p_{j,2t+1})) = \prod_{j\in\{1..n\}} (1-\beta\mathbf{A}_1(i,j)p_{j,2t+1}))$$

Beyond λ: Graph/Network Robustness

- Robustness is the ability of a network to continue performing well when it is subject to failures or attacks.
  - random failure (server down)
  - cascading failure (virus propagating)
  - targeted attack (carefully-chosen agents down)
- How to measure the robustness of a given network?
  - interpretable
  - (strictly) monotonic
  - captures redundancy

Hau Chan, Leman Akoglu, Hanghang Tong: Make It or Break It: Manipulating Robustness in Large Networks. SDM 2014: 325-333

Beyond λ: Graph/Network Robustness

- Study of robustness:
  - mathematics, physics, computer science, biology
- A long (!) and profoundly diverse list of measures:
  - vertex/edge connectivity
  - avg. shortest distance
  - max. shortest distance (diameter)
  - efficiency
  - vertex/edge betweenness
  - clustering coefficient
  - largest component fraction/avg. component size
  - total pairwise connectivity
  - average available flows

Beyond λ: Graph/Network Robustness

- …
- algebraic connectivity
- effective resistance
- number of spanning trees

eigenvalues of the Laplacian **L**

- principal eigenvalue λ_1
- spectral gap $\lambda_1 - \lambda_2$
- natural connectivity

eigenvalues of the adjacency **A**

- other (combinatorial) measures:
  – toughness, scattering number, tenacity, integrity, fault diameter, isoperimetric number, min balanced cut, restricted connectivity, …

Beyond λ: Graph/Network Robustness

- …
- algebraic connectivity
- effective resistance
- number of spanning trees

eigenvalues of the Laplacian **L**

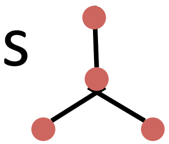
- principal eigenvalue λ_1
- spectral gap $\lambda_1 - \lambda_2$
- natural connectivity

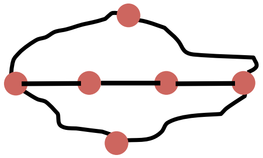
eigenvalues of the adjacency **A**

- other (combinatorial) measures:
  – toughness, scattering number, tenacity, integrity, fault diameter, isoperimetric number, min balanced cut, restricted connectivity, …

A “guide” for “good” robustness measures

- Strict monotonicity
  - improves strictly when edges are added
  - *related: differentiates graphs





- Redundancy
  - accounts for alternative/back-up paths
- Stability
  - does not change drastically by small changes
  - *related: meaningful for disconnected graphs
- Interpretability
  - its meaning is intuitively clear

A “guide” for “good” robustness measures

Measures	S. Monotone	Redundant	Stable	Interpretable
vertex / edge connectivity	✗		✗	✓
avg. shortest distance	✗	✗	✗	✓
diameter	✗	✗	✗	✓
efficiency	✓	✗	✓	✓
vertex / edge betweenness	✓	✗	✗	✓
clustering coefficient	✗		✓	✓
largest component fraction	✗	✗		✓
total pairwise connectivity	✗	✗		✓
avg. available flows		✓	✗	✓
algebraic connectivity	✗		✗	✗
effective resistance	✓	✓	✓	✓
number of spanning trees	✗		✗	
spectral radius / gap			✓	✗
natural connectivity	✓	✓	✓	✓

Hau Chan, Leman Akoglu, Hanghang Tong: Make It or Break It: Manipulating Robustness in Large Networks. SDM 2014: 325-333

Unification of Connectivity Measures

- **Key Idea**: graph connectivity as an **aggregation** over the subgraph connectivity:

$$C(\mathbf{A}) = \sum_{\pi \subseteq \mathbf{A}} f(\pi)$$

  - A: adjacency matrix of the graph
  - π: a non-empty subgraph in A
  - $f(\pi)$: connectivity of the subgraph π
  - $C(A)$: connectivity of graph A

Unification of Connectivity Measures

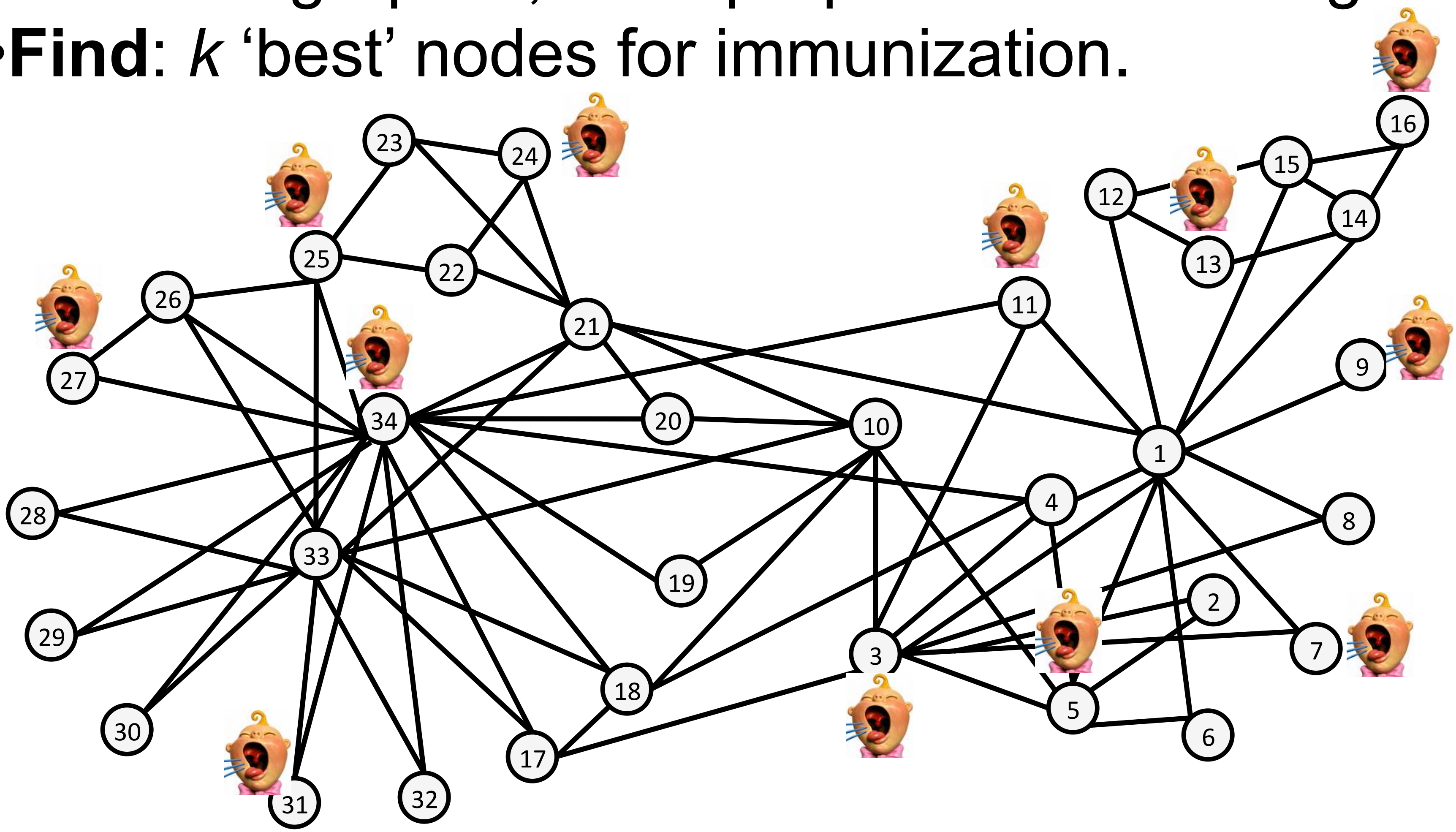
- **Key Idea:** $$C(\mathbf{A}) = \sum_{\pi \subseteq \mathbf{A}} f(\pi)$$
- **Examples**
  - Path Capacity: $$f(\pi) = \begin{cases} \beta^{len(\pi)} & \text{if } \pi \text{ is a valid path of length } len(\pi) \\ 0 & \text{otherwise.} \end{cases}$$
  - Loop Capacity: $$f(\pi) = \begin{cases} 1/len(\pi)! & \text{if } \pi \text{ is a valid loop of length } len(\pi) \\ 0 & \text{otherwise.} \end{cases}$$
  - Triangle Capacity: $$f(\pi) = \begin{cases} 1 & \text{if } \pi \text{ is a triangle} \\ 0 & \text{otherwise.} \end{cases}$$
  - …

Roadmap

- Motivations and Background
- Part I: GCO Measures
- Part II: GCO Theories & Algorithms
- Part III: GCO Applications
- Part IV: Open Challenges & Future Trends

Minimizing Dissemination: Immunization

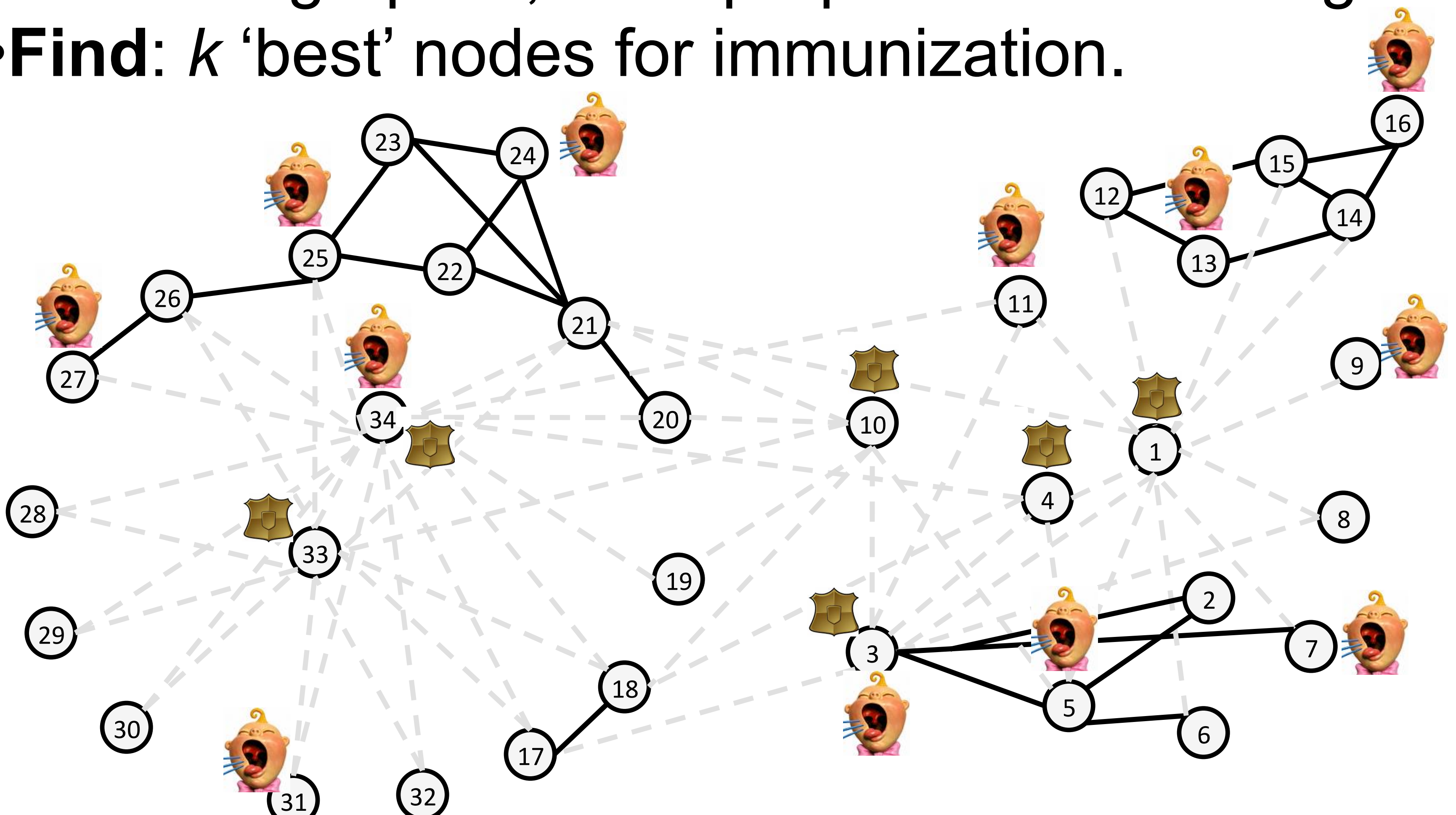
- **Given**: a graph *A*, virus prop model and budget *k*;
- **Find**: *k* 'best' nodes for immunization.



SARS costs 700+ lives; $40+ Bn; H1N1 costs Mexico $2.3bn

Minimizing Dissemination: Immunization

- **Given**: a graph *A*, virus prop model and budget *k*;
- **Find**: *k* 'best' nodes for immunization.

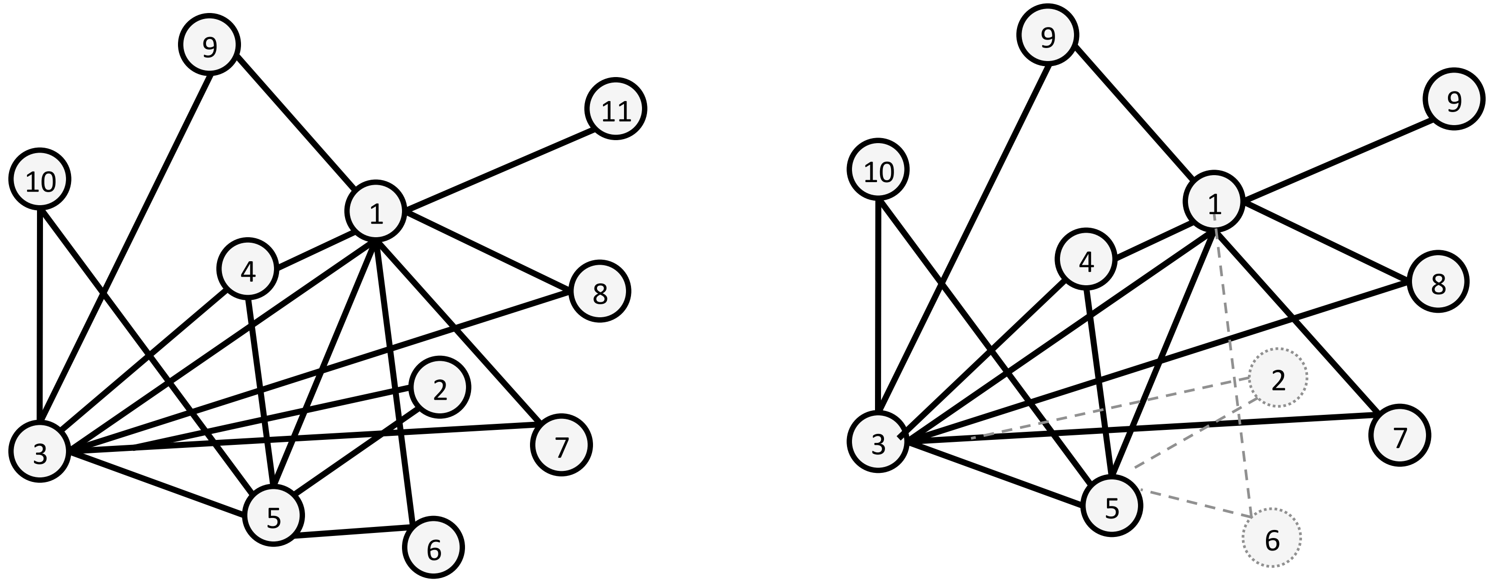


SARS costs 700+ lives; $40+ Bn; H1N1 costs Mexico $2.3bn

Optimal Method

- Select k nodes, whose absence creates the largest drop in λ

$$S = \arg\max_{|S|=k} \lambda - \lambda_S$$



Original Graph: λ Without $\{2, 6\}$: λ_s

Optimal Method

- Select *k* nodes, whose absence creates the largest drop in λ

$$S = \arg\max_{|S|=k} \lambda - \lambda_S$$

Largest eigenvalue w/o subset of nodes *S*

- But, we need $O(\binom{n}{k} \cdot m)$ in time
  - Example: 1,000 nodes, with 10,000 edges
    - It takes 0.01 seconds to compute λ
    - It takes **2,615 years** to find best-*5* nodes !

Theorem: Find Optimal k-node Immunization is NP-Hard

C. Chen, H. Tong, B. Prakash, C. Tsourakakis, T. Eliassi-Rad, C. Faloutsos, D. Chau: Node Immunization on Large Graphs: Theory and Algorithms. IEEE TKDE 2015

Optimal *k*-node immunization is NP-Hard

- **Basic Idea:** Reduction from P1 (known NP-hard)

Given an undirected/unweighted graph G, and k

- **P1 (*k*-independent set problem)**: is there k nodes, no two of which are adjacent?
- **P2 (*k*-node immunization problem)**: is there k nodes, the deletions of which makes the leading eigenvalues ≤ 0

$$A = \begin{bmatrix} S_{kxk} & X_{(k)x(n-k)} \\ X_{(k)x(n-k)} & T_{(n-k)x(n-k)} \end{bmatrix}$$

- **Proof #1: If YES to P1(G,k)→ YES to P2(G, n-k)**

YES to P1 ➡ $S_{kxk} = \mathbf{0} \xrightarrow[\text{Nodes in } T]{\text{Removing}} \lambda(\tilde{A}) = \lambda(\mathbf{0}) = 0$ ➡ YES to P2

- **Proof #2: If NO to P1(G,k)→ NO to P2(G, n-k)**

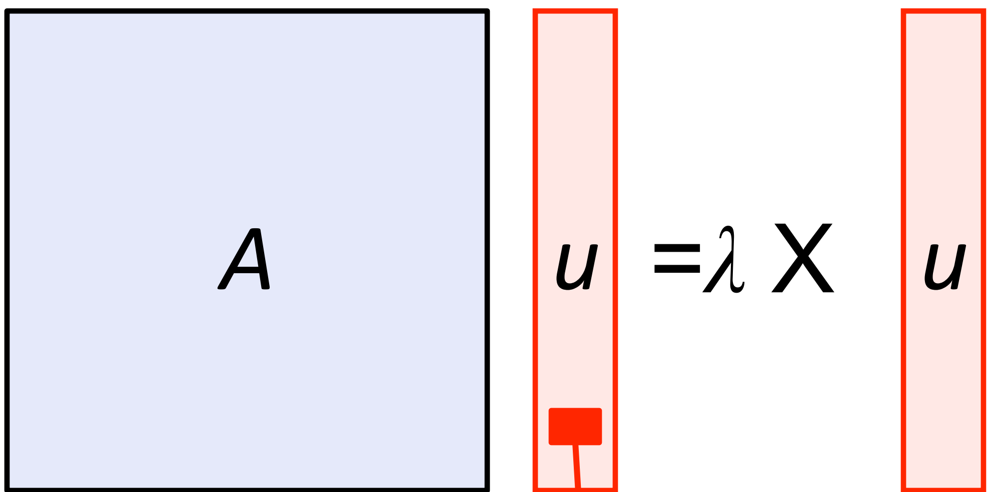
Suppose YES to P2 $\xrightarrow[\text{Nodes in } T]{\text{Removing}} \lambda(\tilde{A}) = \lambda(\mathbf{0}) \leq 0 \xrightarrow{S(i,j)\geq 0}$

➡ $S_{kxk} = \mathbf{0}$ ⟷ Nodes in ***S*** being ind. set ➡ contradict

Netshield to the Rescue

Theorem:

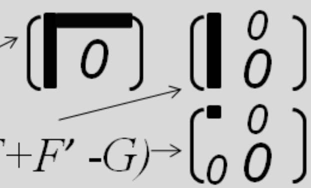
(1) $\lambda - \lambda_s \approx Sv(S) = \sum_{i \in S} 2\lambda u(i)^2 - \sum_{i,j \in S} A(i,j)u(i)u(j)$



$u(i)$: eigen-score

$A_S = A - E$

$= A - (F + F' - G)$



$\lambda_s = \lambda - u' Eu/(u'u) + O(|E|^2)$

$= \lambda - 2u' Fu + 2u' Eu + O(|E|^2)$

$= \lambda - (\sum_{i \in S} 2\lambda u(i)^2 - \sum_{i,j \in S} A(i,j)u(i)u(j)) + O(|E|^2)$

Footnote:
$u(i)$ ~ PageRank(i) ~ in-degree(i)

C. Chen, H. Tong, B. Prakash, C. Tsourakakis, T. Eliassi-Rad, C. Faloutsos, D. Chau: Node Immunization on Large Graphs: Theory and Algorithms. IEEE TKDE 2015

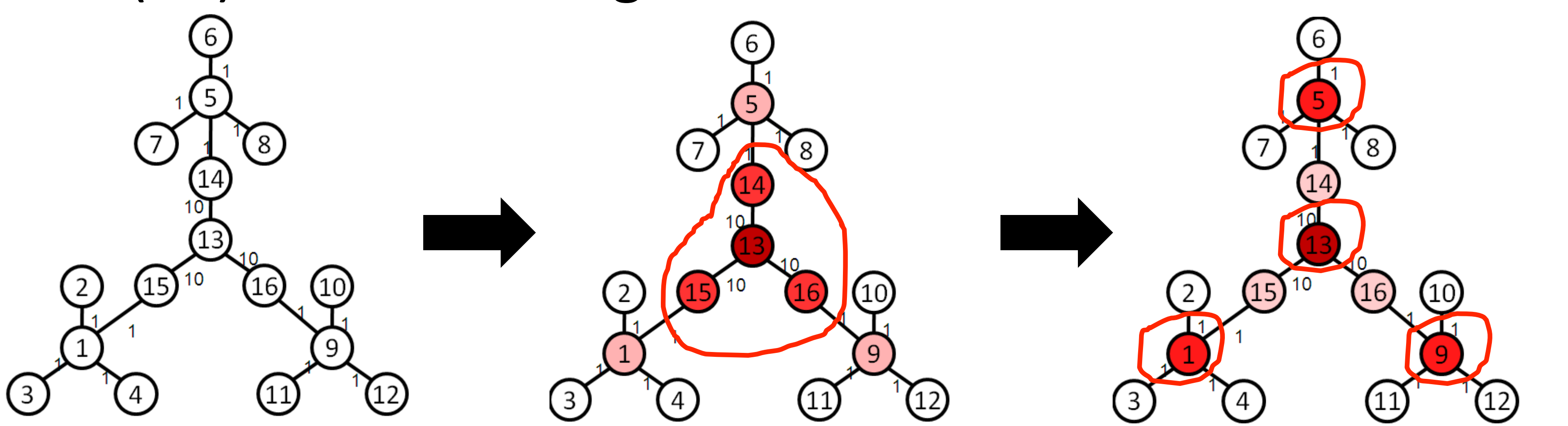
Intuition

Netshield to the Rescue

Theorem:

(1) $\lambda - \lambda_s \approx Sv(S) = \sum_{i\in S} 2\lambda u(i)^2 - \sum_{i,j\in S} A(i,j)u(i)u(j)$

- find a set of nodes *S (e.g. k=4)*, which
  - (C1) each has high eigen-scores
  - (C2) diverse among themselves



Original Graph   Select by C1   Select by C1+C2

(1) $\lambda - \lambda_s \approx Sv(S)$ ✓
(2) $Sv(S)$ is submodular
(3) *Netshield* is near-opt
(4) *Netshield* scales linearly

Netshield to the Rescue

Theorem:

(1) $\lambda - \lambda_s \approx Sv(S) = \sum_{i \in S} 2\lambda u(i)^2 - \sum_{i,j \in S} A(i,j)u(i)u(j)$

(2) Sv(S) is sub-modular (+monotonically non-decreasing)

Corollary:

(3) *Netshield* is near-optimal (wrt max Sv(*S*))

(4) *Netshield* is $O(nk^2+m)$

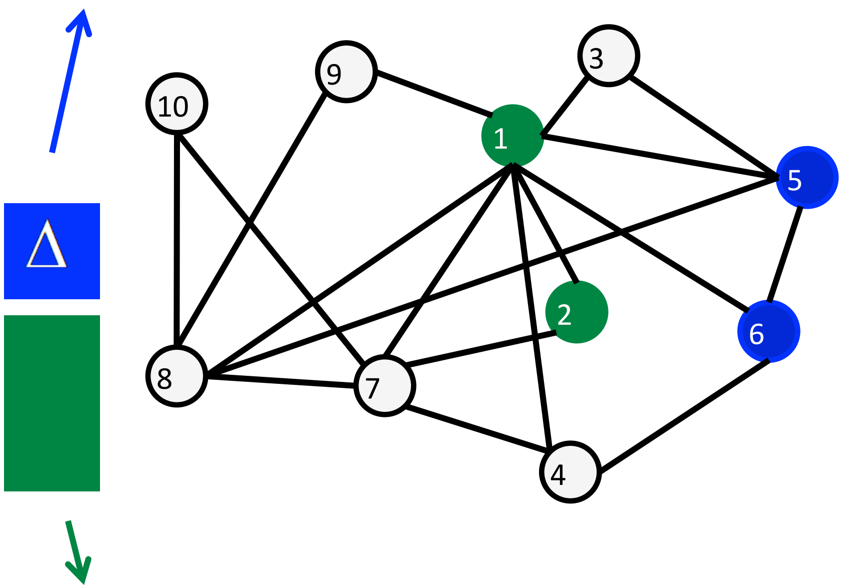
- Example: 1,000 nodes, with 10,000 edges
  - *Netshield* takes **< 0.1 seconds** to find best-*5* nodes !
  - ... as opposed to **2,615 years**

Footnote: near-optimal means $Sv(S^{Netshield}) >= (1-1/e)\, Sv(S^{Opt})$

(1) $\lambda - \lambda_s \approx Sv(S)$ ✓
(2) $Sv(S)$ is submodular
(3) *Netshield* is near-opt
(4) *Netshield* scales linearly

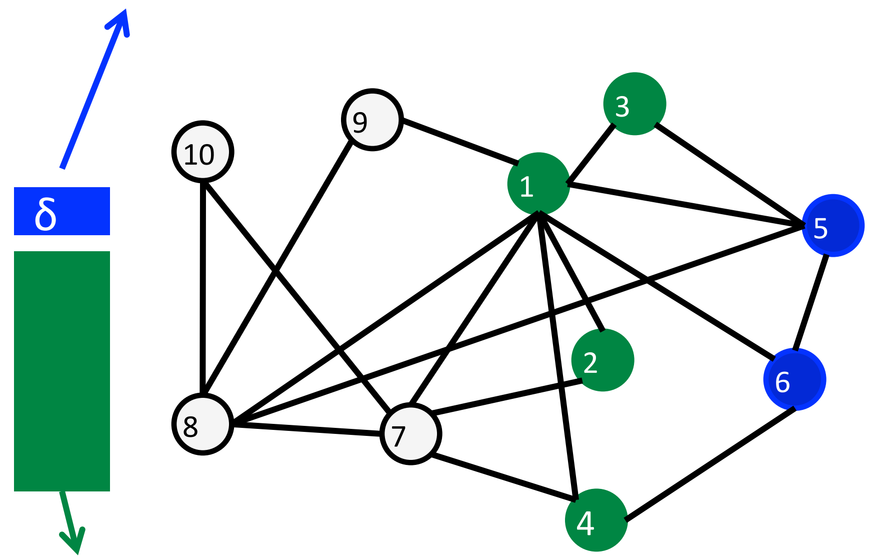
Why *Netshield* is Near-Optimal?

Marginal benefit of deleting {5,6}



Benefit of deleting {1,2}

Marginal benefit of deleting {5,6}

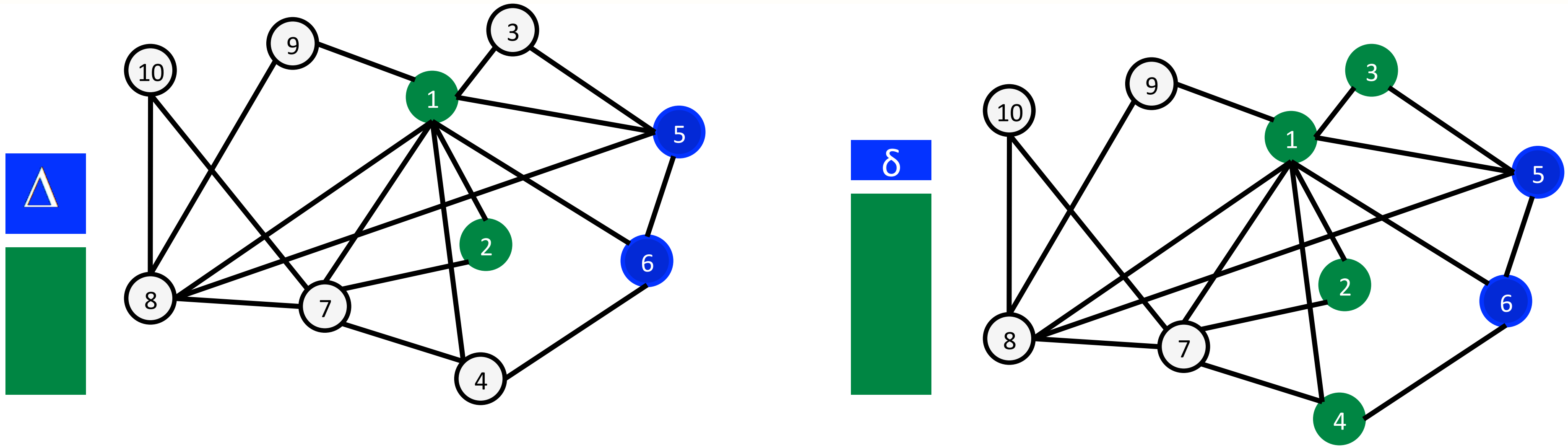


Benefit of deleting {1,2, 3,4}

Δ >= δ ⟺ Sub-Modular (i.e., Diminishing Returns)

Why *Netshield* is Near-Optimal?

(1) $\lambda - \lambda_s \approx Sv(S)$ ✓
(2) $Sv(S)$ is submodular
(3) *Netshield* is near-opt
(4) *Netshield* scales linearly

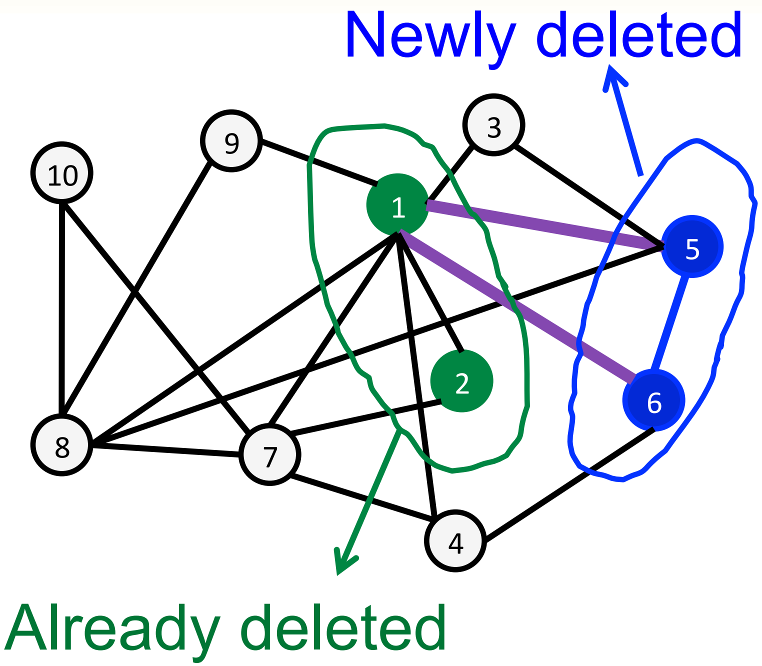


Δ >= δ ⟷ Sub-Modular (i.e., Diminishing Returns)

Theorem: *k*-step greedy alg. to maximize a sub-modular function guarantees (1-1/e) optimal [Nemhauster+ 78]

Why Sv(S) is sub-modular?

(1) λ - $\lambda_s \approx$ Sv(S)
(2) Sv(S) is submodular
(3) *Netshield* is near-opt
(4) *Netshield* scales linearly

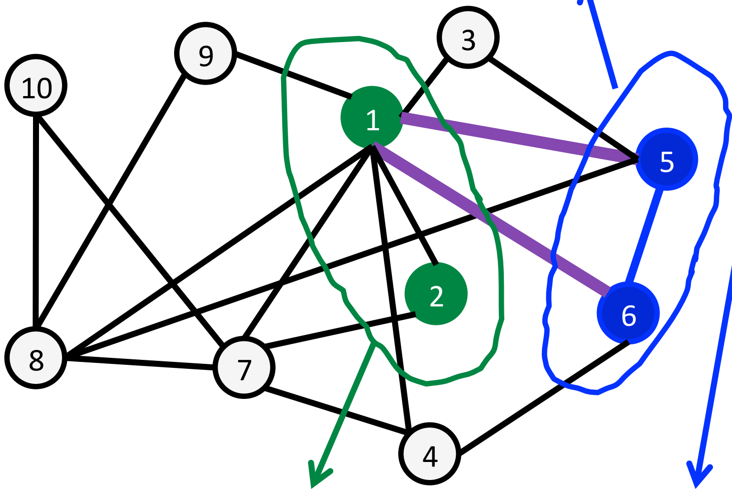


- H. Tong, B. Prakash, C. Tsourakakis, T. Eliassi-Rad, C. Faloutsos, D. Chau: On the Vulnerability of Large Graphs. ICDM 2010: 1091-1096
- C. Chen, H. Tong, B. Prakash, C. Tsourakakis, T. Eliassi-Rad, C. Faloutsos, D. Chau: Node Immunization on Large Graphs: Theory and Algorithms. IEEE TKDE 2015

Why Sv(S) is sub-modular?

(1) $\lambda - \lambda_s \approx Sv(S)$
(2) Sv(S) is submodular
(3) *Netshield* is near-opt
(4) *Netshield* scales linearly

Newly deleted



Already deleted

Marginal Benefit of deleting {5,6}

$$Sv(S_{green} \cup S_{blue}) - Sv(S_{green}) =$$

$$\sum_{i \in S_{blue}} 2\lambda u(i)^2 - \sum_{i,j \in S_{blue}} A(i,j)u(i)u(j) \quad -$$

$$\left(\sum_{i \in S_{blue},\, j \in S_{green}} A(i,j)u(i)u(j) + \sum_{i \in S_{green},\, j \in S_{blue}} A(i,j)u(i)u(j)\right)$$

Pure benefit from {5,6}

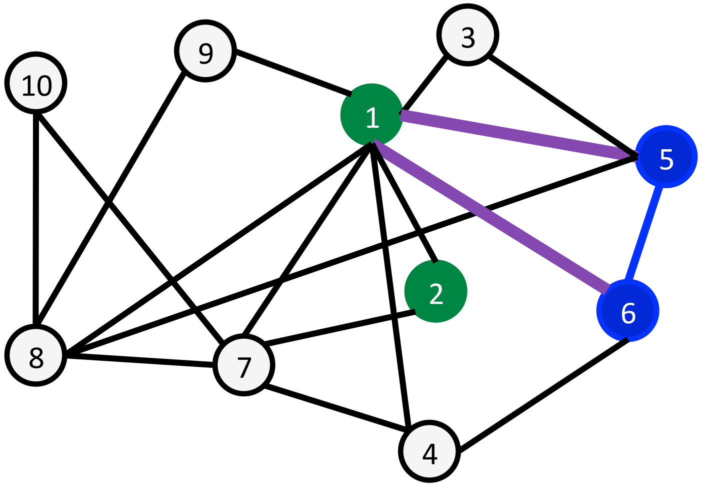
Interaction between {5,6} and {1,2}

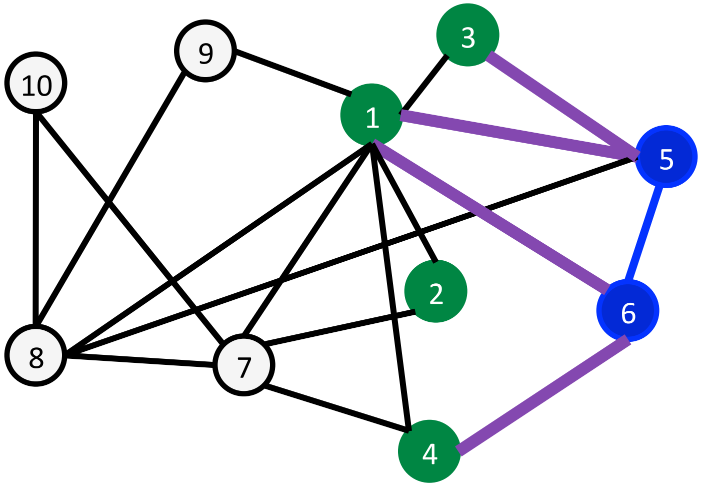
Only purple term depends on {1, 2}!

- H. Tong, B. Prakash, C. Tsourakakis, T. Eliassi-Rad, C. Faloutsos, D. Chau: On the Vulnerability of Large Graphs. ICDM 2010: 1091-1096
- C. Chen, H. Tong, B. Prakash, C. Tsourakakis, T. Eliassi-Rad, C. Faloutsos, D. Chau: Node Immunization on Large Graphs: Theory and Algorithms. IEEE TKDE 2015

Why Sv(S) is sub-modular?

(1) $\lambda - \lambda_s \approx$ Sv(S) ✓
(2) Sv(S) is submodular
(3) *Netshield* is near-opt ✓
(4) *Netshield* scales linearly





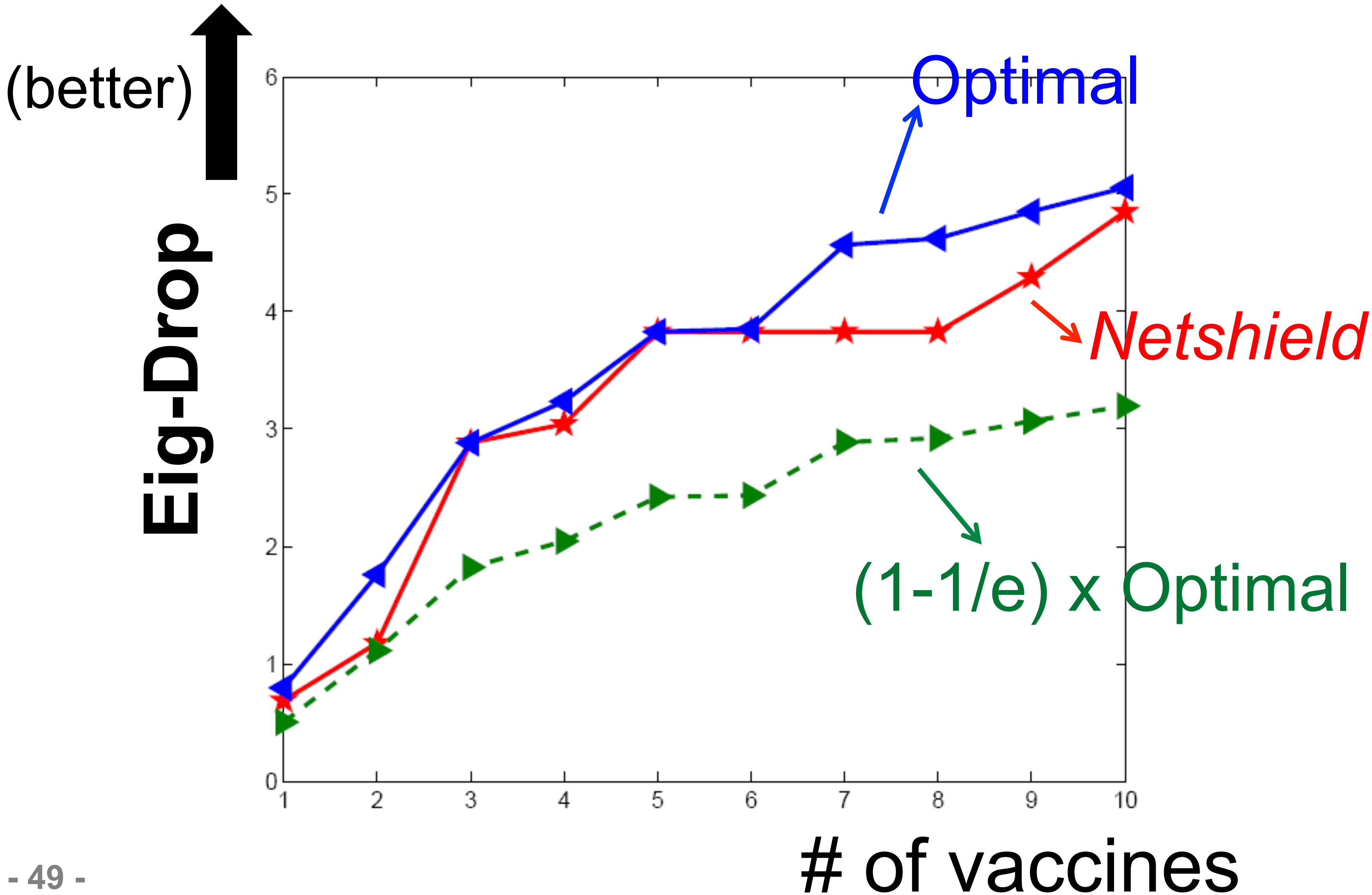
Marginal Benefit = Blue –Purple

More Green ⟷ More Purple ⟷ Less Red

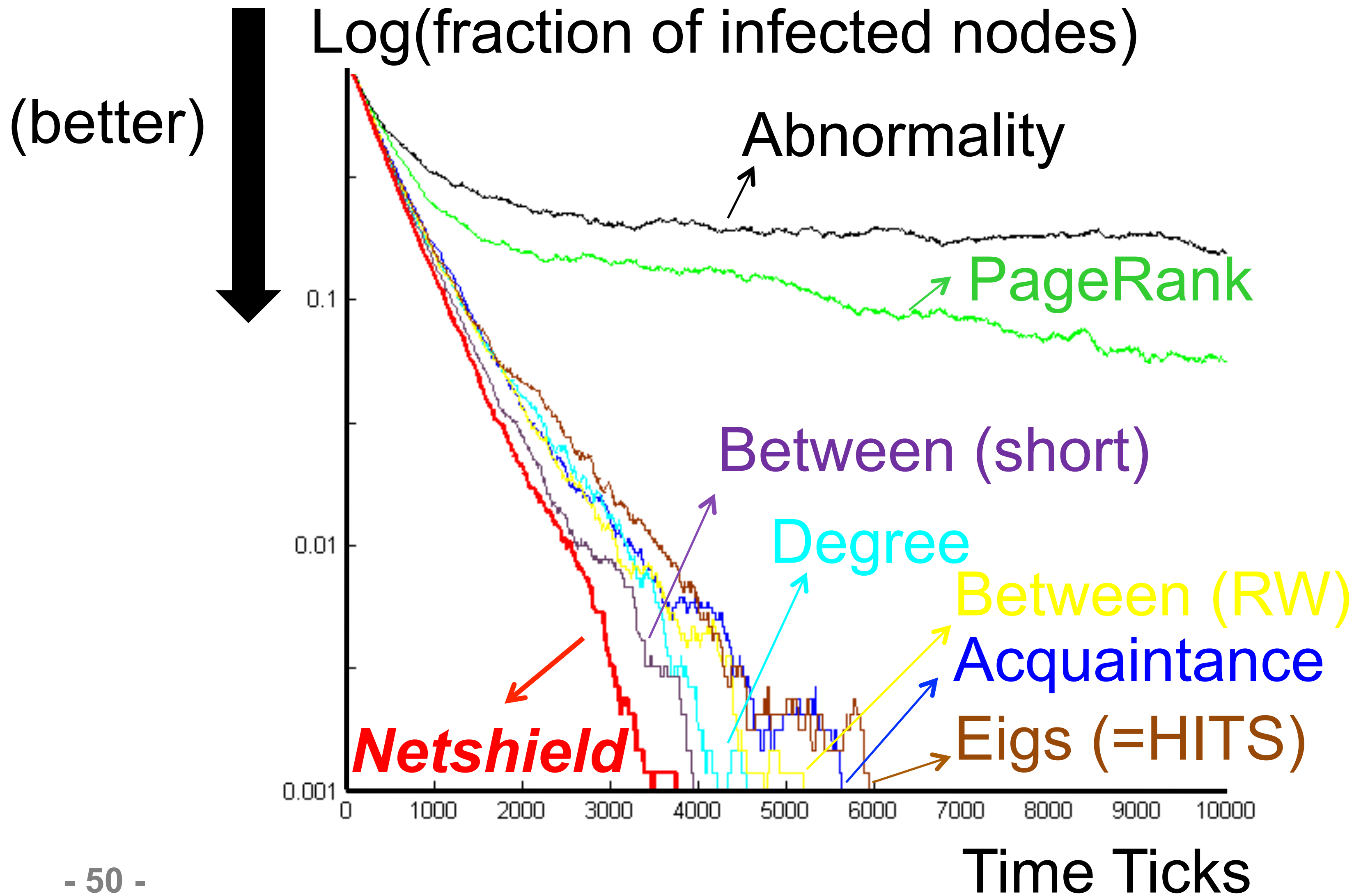
Marginal Benefit of Left >= Marginal Benefit of Right

Footnote: greens are nodes already deleted; blue {5,6} nodes are nodes to be deleted

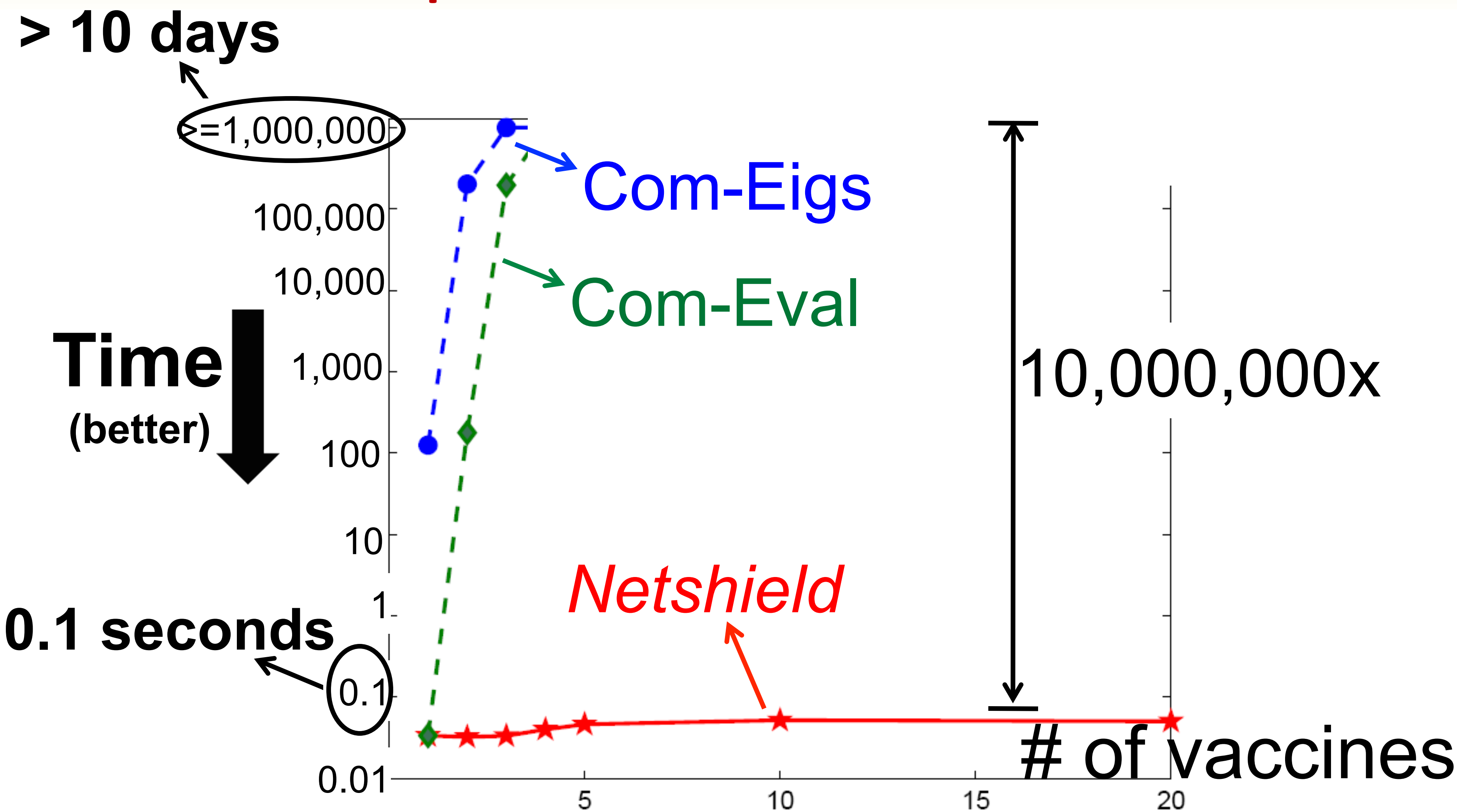
Quality of *Netshield*



Comparison of Immunization

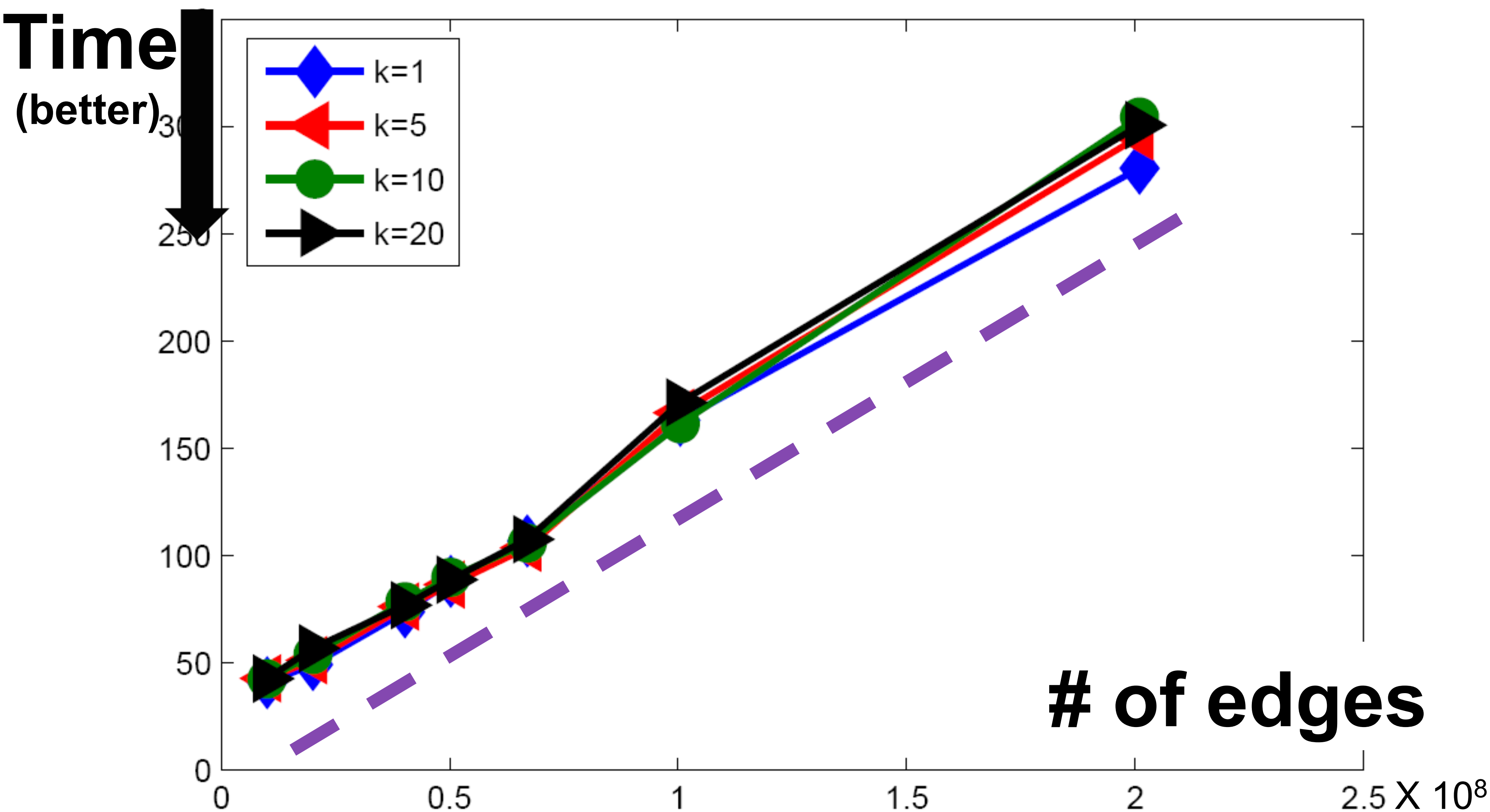


Speed of *Netshield*



NIPS co-authorship Network: 3K nodes, 15K edges

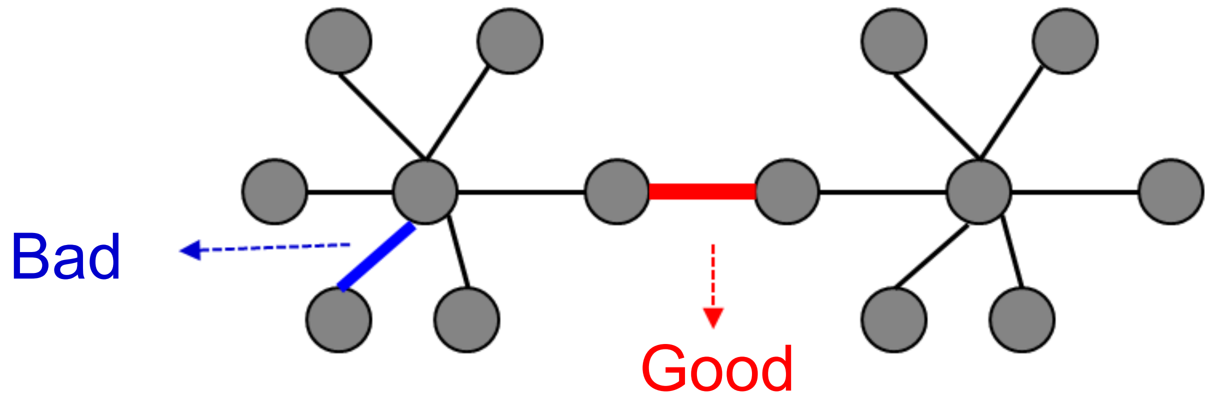
Scalability of *Netshield*



DATA Lab

From Node Deletion to Edge Deletion

- Given: a graph *A*, virus prop model and budget *k*;
- Find: delete *k* 'best' edges from *A* to minimize λ



Our Solutions: 1st order matrix perturbation again!

$$\lambda - \lambda_s \approx Mv(S) = c \sum_{e \in S} u(i_e) v(j_e)$$

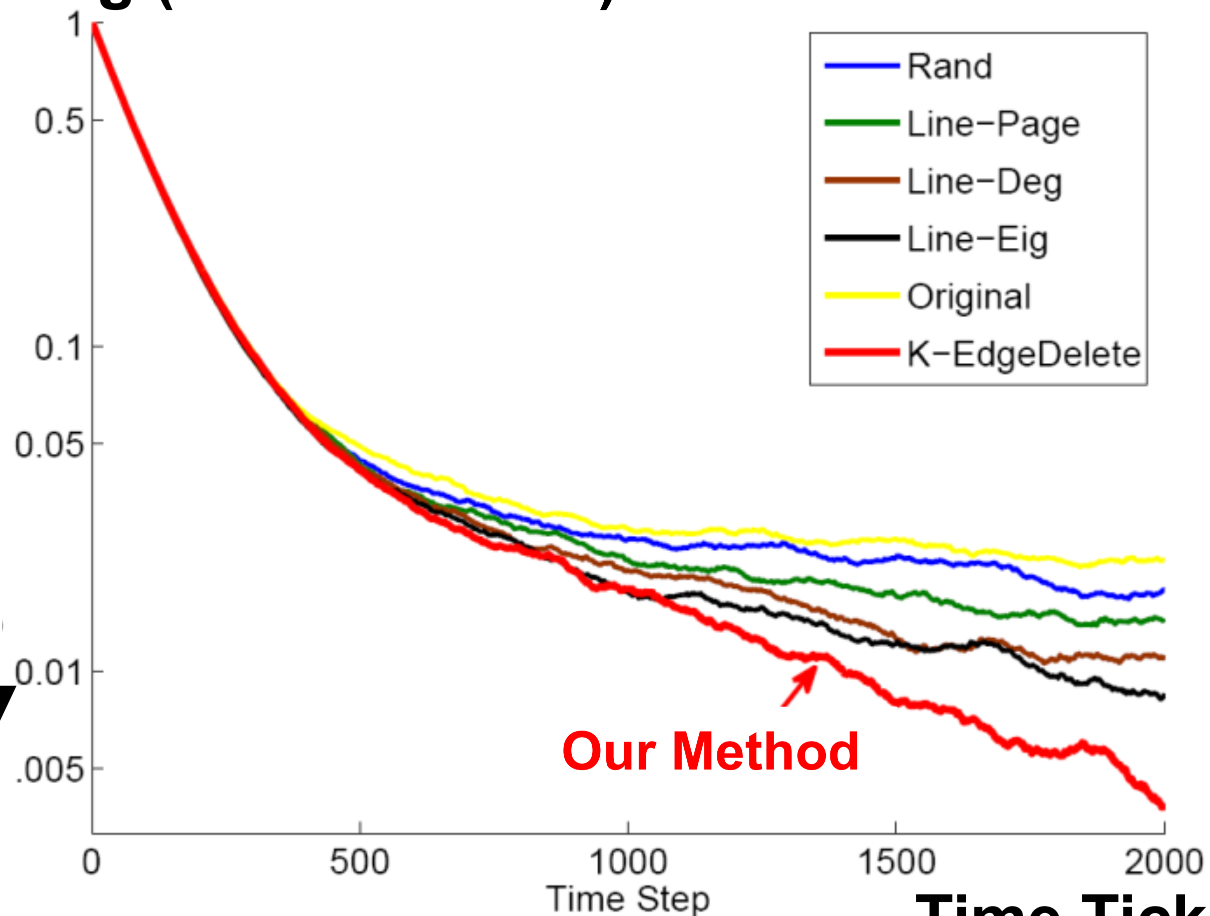
Left eigen-score of source

Right eigen-score of target

Minimizing Propagation: Evaluations

Log (Infected Ratio)

(better)

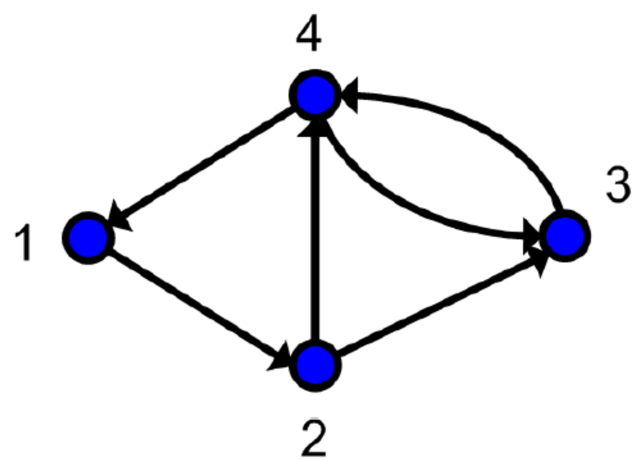


Time Ticks

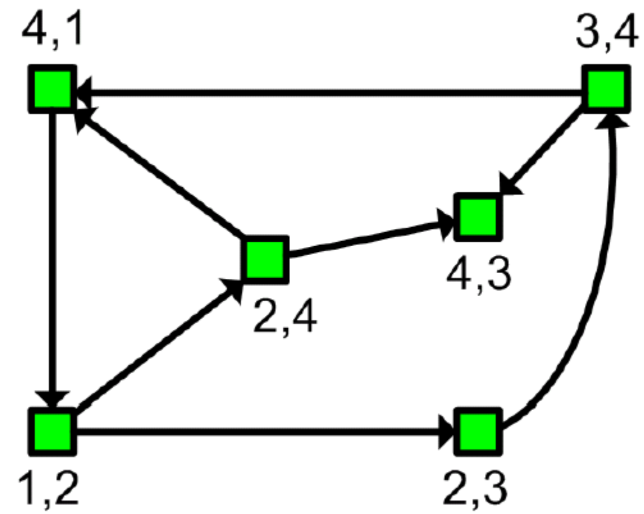
Data set: Oregon Autonomous System Graph (14K node, 61K edges)

Discussions: Node Deletion vs. Edge Deletion

- Observations:
  - Node or Edge Deletion → *λ Decrease*
  - Nodes on A = Edges on its line graph *L(A)*



Original Graph *A*



Line Graph *L(A)*

- Questions?
  - Edge Deletion on *A* = Node Deletion on *L(A)*?
  - Which strategy is better (when both feasible)?

Discussions: Node Deletion vs. Edge Deletion

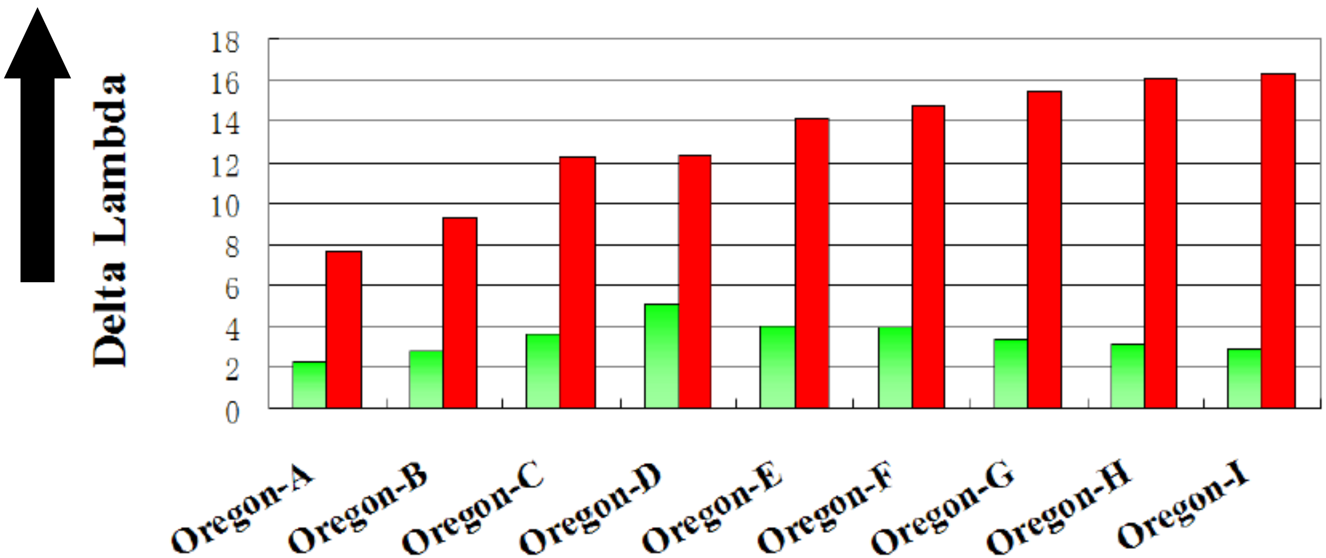
- Q: Is Edge Deletion on *A* = Node Deletion on *L(A)*?
- A: Yes!

Theorem: Line Graph Spectrum.
Eigenvalue of *A* → Eigenvalue of *L(A)*

Discussions: Node Deletion vs. Edge Deletion

- Q: Which strategy is better (when both feasible)?
- A: Edge Deletion > Node Deletion

(better)



Green: Node Deletion (e.g., shutdown a twitter account)
Red: Edge Deletion (e.g., un-friend two users)

Maximizing Dissemination: Edge Addition

•**Given**: a graph *A*, virus prop model and budget *k*;
•**Find**: add *k* 'best' new edges into *A*.

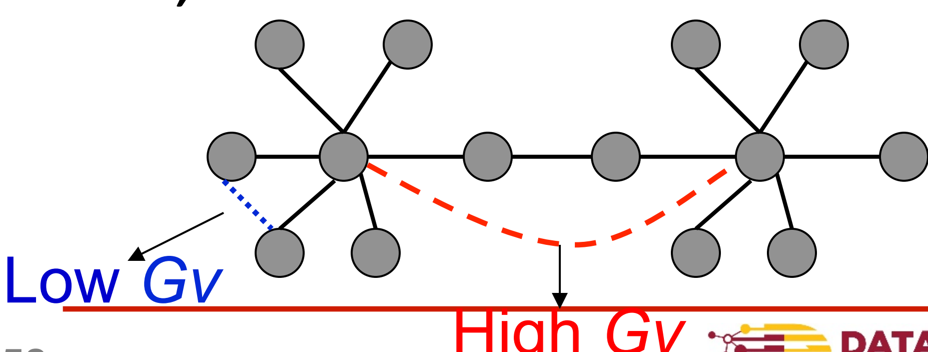
- By 1st order perturbation, we have

$$\lambda_s - \lambda \approx Gv(S) = c \sum_{e \in S} u(i_e) v(j_e)$$

Left eigen-score of source

Right eigen-score of target

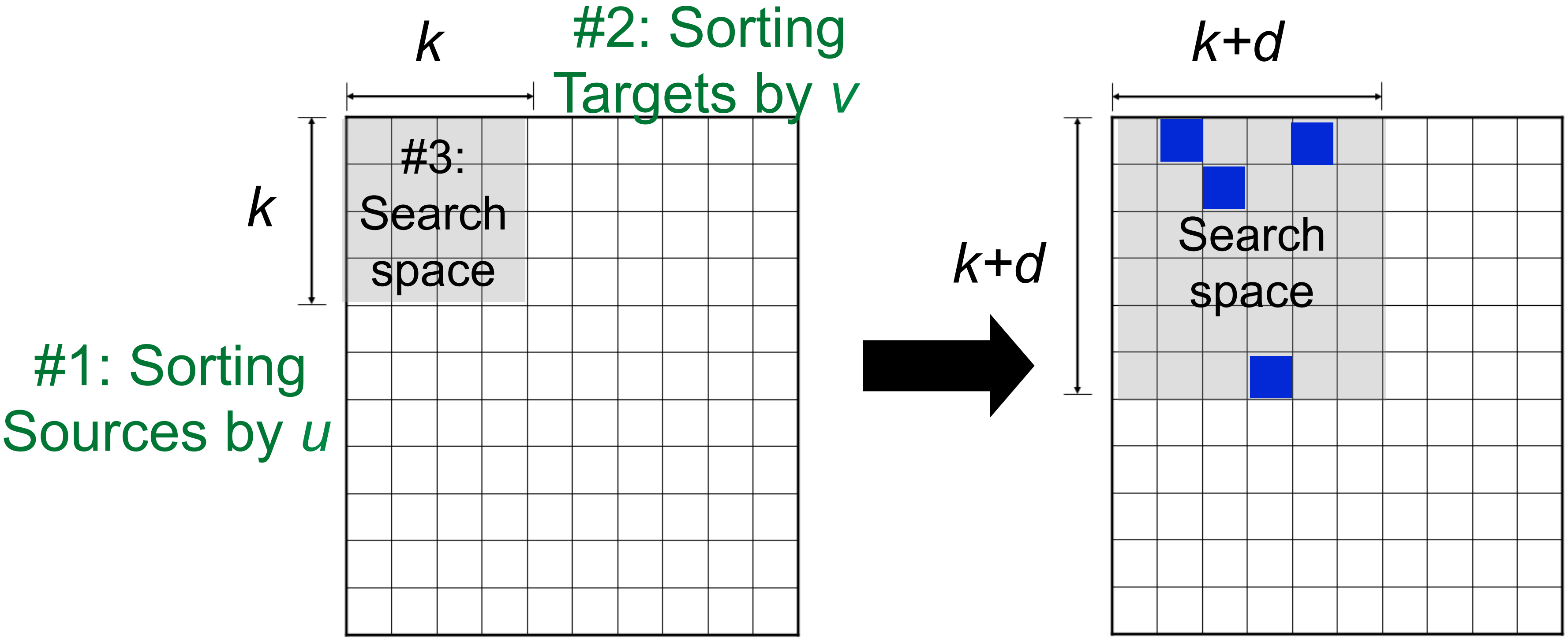
- So, we are done → need O(n^2-m) complexity



Maximizing Dissemination: Edge Addition

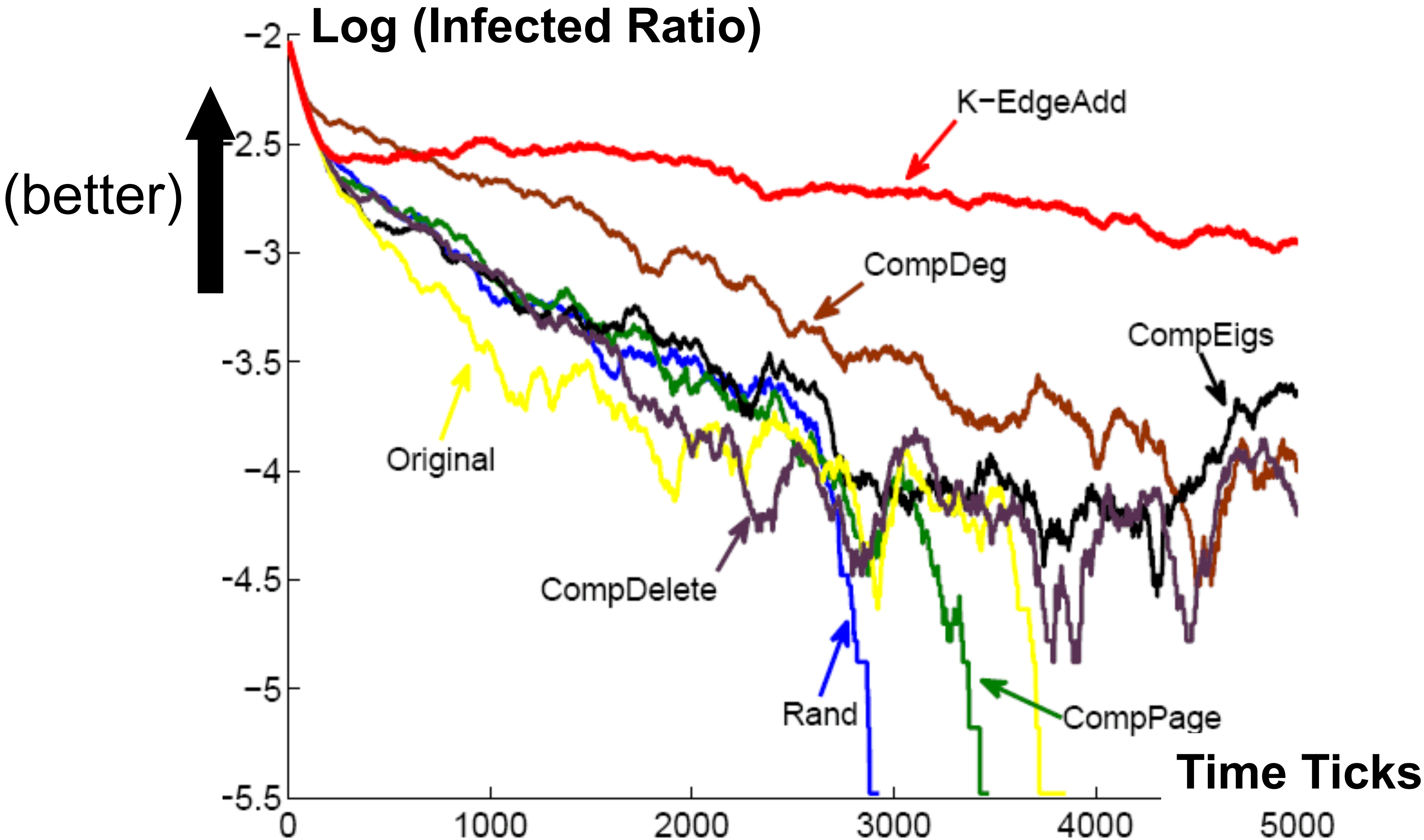
$$\lambda_s - \lambda \approx Gv(S) = c \sum_{e \in S} u(i_e) v(j_e)$$

- Q: How to Find k new edges w/ highest $Gv(S)$?
- A: Modified Fagin's algorithm



Time Complexity: $O(m+nt+kt^2)$, $t = \max(k,d)$:existing edge

Maximizing Dissemination: Evaluation



More on GCO Algorithms

- **M1: Higher Order Variants**
  - \`Better' Matrix Perturbation → Better Approximation of Eigen-gap?
  - C. Chen, H. Tong, B. Prakash, C. Tsourakakis, T. Eliassi-Rad, C. Faloutsos, D. Chau: Node Immunization on Large Graphs: Theory and Algorithms. IEEE TKDE 2015
- **M2: Beyond Full & Symmetric Immunity**
  - Immunizing a node weakens (but not deleting) the incoming (but not the out-going) links
  - B. Aditya Prakash, Lada Adamic, Theodore Iwashnya, Hanghang Tong and Christos Faloutsos: Fractional Immunization on Networks. SDM 2013

More on GCO Algorithms (cont.)

- **M3: Immunization on Dynamic Graphs**
  - Optimize connectivity on Time-Varying Graphs (with alternating behavior)
  - B. Aditya Prakash, Hanghang Tong, Nicholas Valler, Michalis Faloutsos, Christos Faloutsos: Virus Propagation on Time-Varying Networks: Theory and Immunization Algorithms. ECML/PKDD (3) 2010: 99-114
- **M4: Manipulating Network Robustness**
  - Beyond λ: Optimizing an eigen-function of the underlying graph
  - Hau Chan, Leman Akoglu, Hanghang Tong: Make It or Break It: Manipulating Robustness in Large Networks. SDM 2014: 325-333

More on GCO Algorithms (cont.)

- **M5: Robust Network Construction**
  – How to building a `well-connected' network, that is robust to external intentional attack, with resource constraint?
  – Hui Wang, Wanyun Cui, Yanghua Xiao, Hanghang Tong:Robust network construction against intentional attacks. BigComp 2015: 279-286
- **M6: Vaccine Distribution with Uncertainty**
  – Optimizing the connectivity of a 'noisy', uncertain graph.
  – Yao Zhang and B. Aditya Prakash: Scalable Vaccine Distribution in Large Graphs given Uncertain Data. ICDM 2014
  – Code available at: http://people.cs.vt.edu/badityap/CODE/UDAV.zip

More on GCO Algorithms (cont.)

- **M7: Handling Small Eigen-Gap**
  - Optimal edge deletion strategy on a graph with small eigen-gap (e.g., social networks), where matrix-perturbation might collapse.
  - L. Le, T. Eliassi-Rad and H. Tong: MET: A Fast Algorithm for Minimizing Propagation in Large Graphs with Small Eigen-Gaps. SDM 2015
- **M8: Source/Target-Specific Connectivity Optimization**
  - Identifying most important nodes in connecting two nodes, or two groups of nodes
  - Hanghang Tong, Spiros Papadimitriou, Christos Faloutsos, Philip S. Yu, Tina Eliassi-Rad: Gateway finder in large graphs: problem definitions and fast solutions. Inf. Retr. 15(3-4): 391-411 (2012)

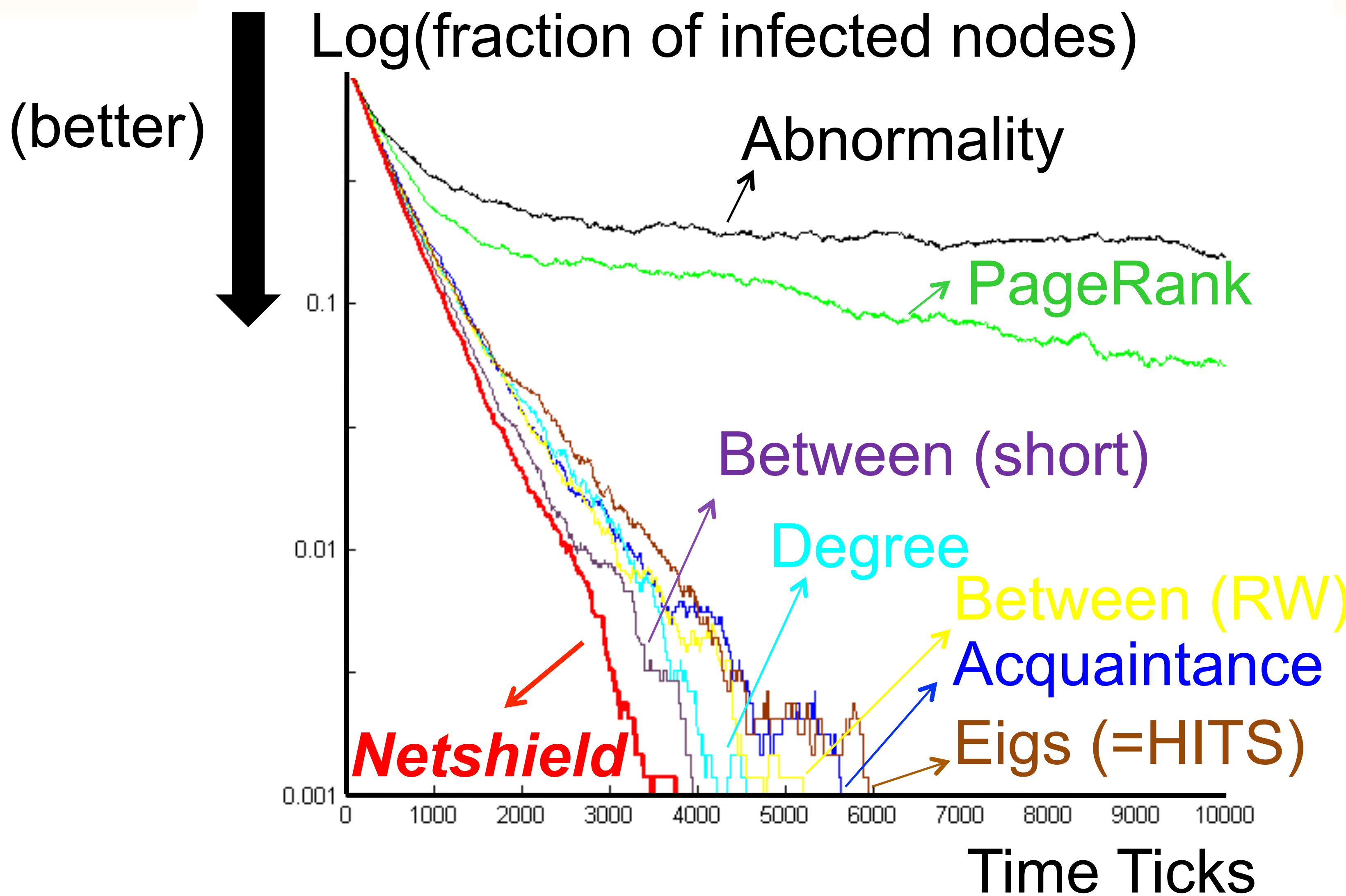
Roadmap

- Motivations and Background
- Part I: GCO Measures
- Part II: GCO Theories & Algorithms
- Part III: GCO Applications
- Part IV: Open Challenges & Future Trends

Part III: Applications

- A1: Immunization
- A2: Optimal Resource Allocation
- A3: Optimal Network Demolition: Collective Influence
- A4: Diversified Ranking on Graphs
- A5: Information Spreading in Context
- A6: Vulnerability of Cyber-Physical Systems
- A7: Team Member Replacement
- A8: Competitive Virus on Composite Networks
- A9: Gateway finder

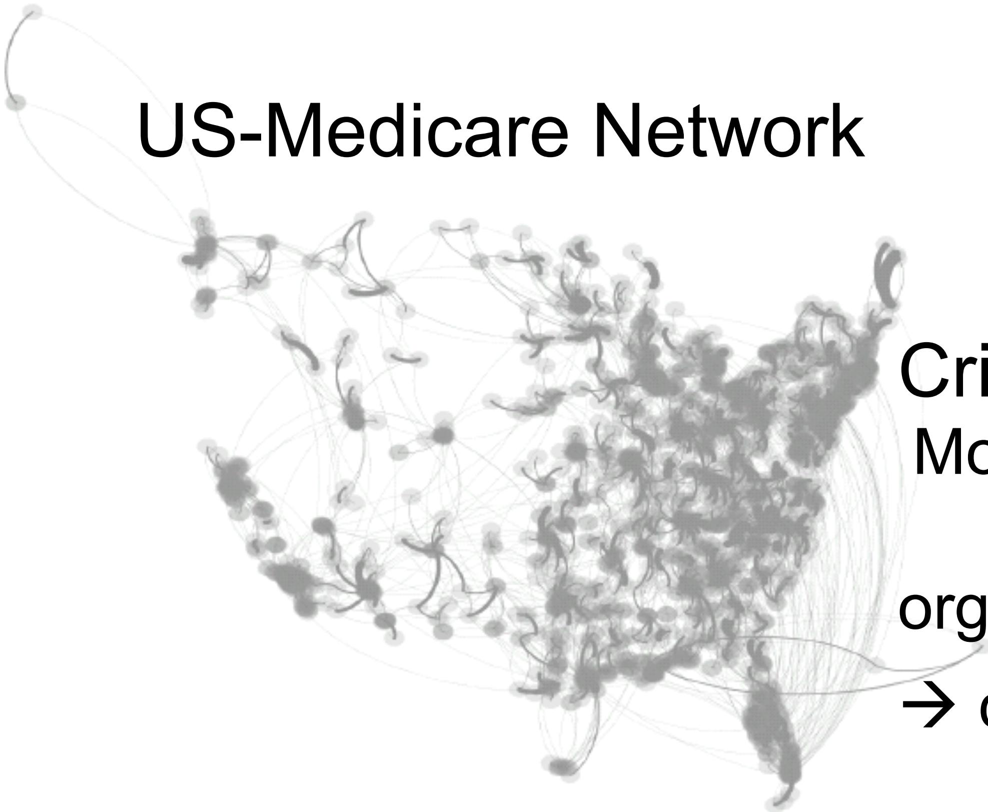
A1: Immunization



- H. Tong, B. Prakash, C. Tsourakakis, T. Eliassi-Rad, C. Faloutsos, D. Chau: On the Vulnerability of Large Graphs. ICDM 2010: 1091-1096
- C. Chen, H. Tong, B. Prakash, C. Tsourakakis, T. Eliassi-Rad, C. Faloutsos, D. Chau: Node Immunization on Large Graphs: Theory and Algorithms. IEEE TKDE 2015

A2: Optimal Recourse Allocation

US-Medicare Network



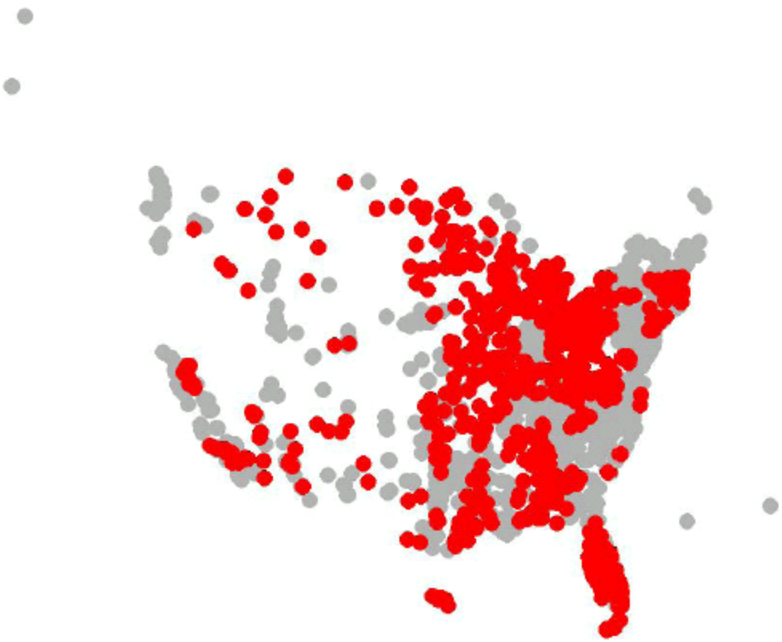
Critical Patient transferring

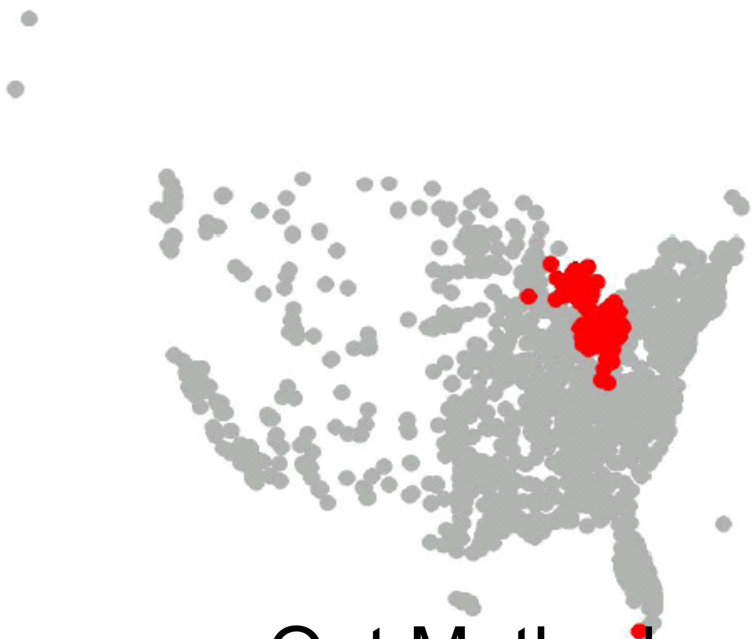
Move patients → specialized care → highly resistant micro-organism → Infection controlling → costly & limited

Q: How to allocate resource to minimize overall spreading?

SARS costs 700+ lives; $40+ Bn; H1N1 costs Mexico $2.3bn; Flu 2013: one of the worst in a decade, 105 children in US.

A2: Optimal Recourse Allocation

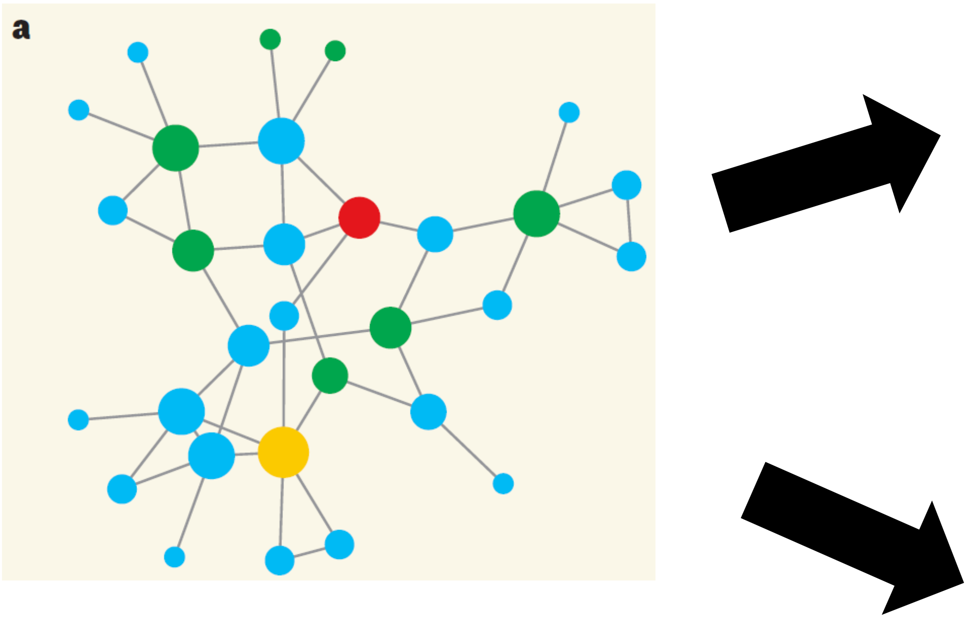


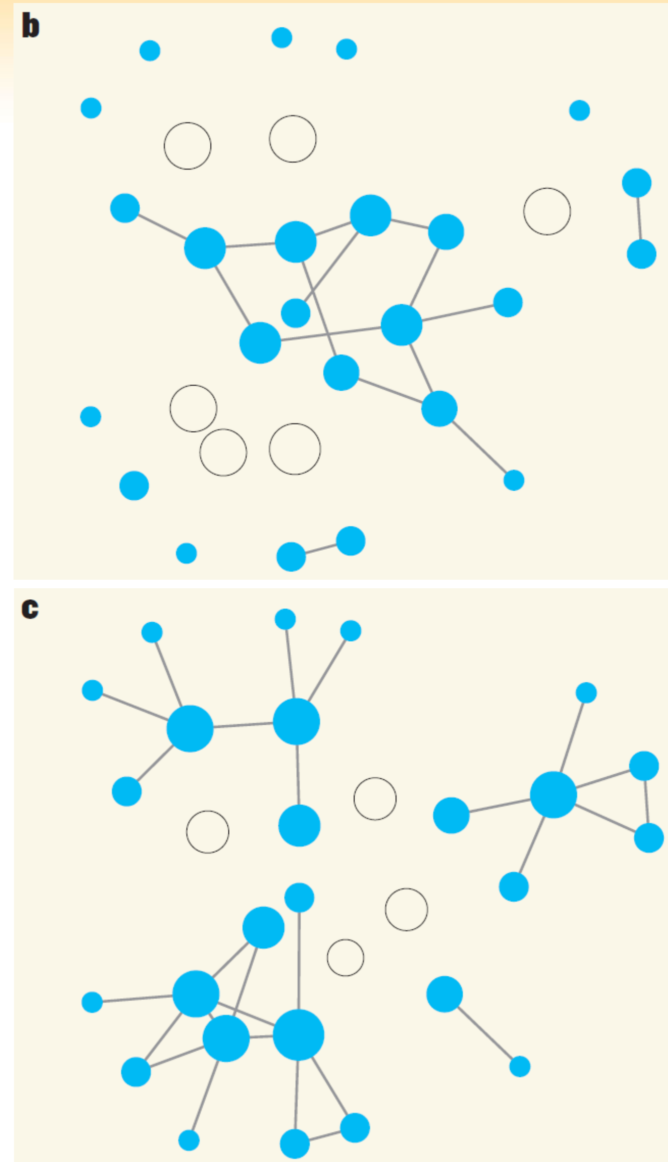


Red: Infected Hospitals after 365 days

B. Aditya Prakash, Lada Adamic, Theodore Iwashnya, Hanghang Tong and Christos Faloutsos: Fractional Immunization on Networks. SDM 2013

A3: Optimal Network Demolition: Collective Influence





(a): the original input network.
(b): removing six (white) nodes w/ highest individual influence scores → GCC of size12.
(c): removing four (white) nodes with highest collective influence→ GCC of size 10.

István A. Kovács & Albert-László Barabási: Network science: Destruction perfected. Nature 524, 38–39, 2015

A4: Diversified Ranking on Large Graphs

- Q: Why Diversity?
- A1: Uncertainty & Ambiguity ***in*** an Information

jaguar | Search

Case 1: Uncertainty from the query

swine flu | Search

About 16,200,000 results (0.07 seconds) Advanced search

Case 2: Uncertainty from the user

"Swine Flu" Pathology
by BS Beetles
Jul 24, 2009 ... "**Swine Flu**" **Pathology**. Figure. CREDIT: MAINES ET AL. The clinical spectrum of disease caused by the swine-origin 2009 A(H1N1) influenza ...
www.sciencemag.org/content/325/5939/367.2.full

Swine Flu Symptoms
Review common **swine flu** symptoms, which can include high fever, cough, runny nose, cough, and body aches, and how to tell the difference between **swine flu** ...
pediatrics.about.com/od/**swineflu**/a/409_symptoms.htm - Cached - Similar

Hanghang Tong, Jingrui He, Zhen Wen, Ravi Konuru, Ching-Yung Lin:
Diversified ranking on large graphs: an optimization viewpoint. KDD 2011: 1028-1036

A4: Why Diversity? (cont.)

- A2: Address uncertainty & ambiguity ***of*** an information need
  - C1: Product search → want different reviews
  - C2: Political issue debate → desire different opinions
  - C3: Legal search → find ALL relevant cases
  - C4: Team assembling → find a set of relevant & diversified experts
- A3: Become a ***better*** and ***safer*** employee
  - ***Better***: A *1%* increase in diversity → an additional *$886* of monthly revenue
  - ***Safer***: A *1%* increase in diversity → an increase of *11.8%* in job retention

Hanghang Tong, Jingrui He, Zhen Wen, Ravi Konuru, Ching-Yung Lin:
Diversified ranking on large graphs: an optimization viewpoint. KDD 2011: 1028-1036

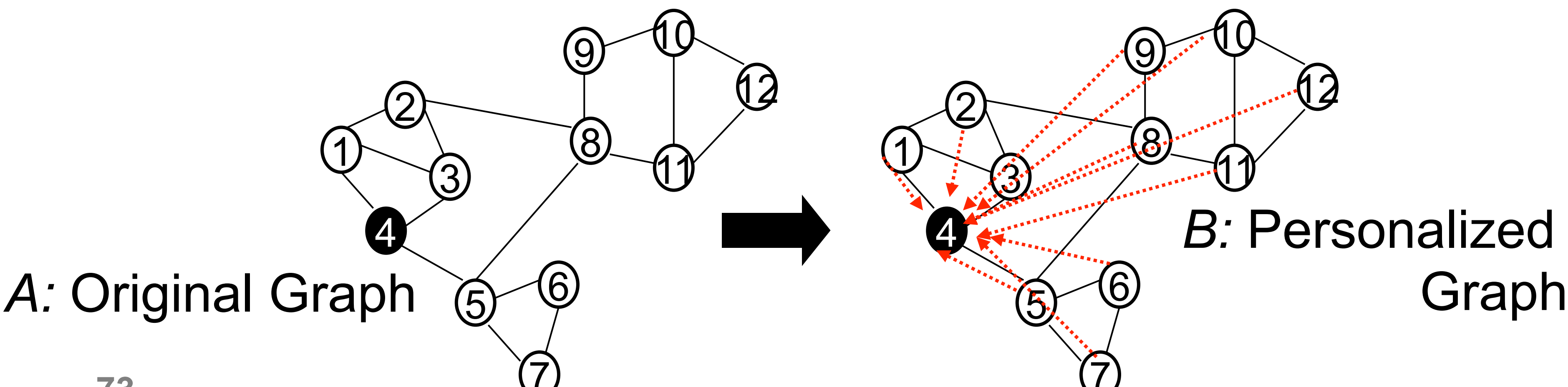
A4: Our Solutions (10 sec. introduction!)

- Problem 1 (Evaluate/measure a given top-k ranking list)
- A1: A weighted sum between relevance and similarity

weight relevance diversity

$$g(\mathcal{S}) = w \sum_{i \in \mathcal{S}} \mathbf{r}(i) - \sum_{i,j \in \mathcal{S}} \mathbf{B}(i,j)\mathbf{r}(j)$$

- Problem 2 (Find a near optimal top-k ranking list)
- A2: A greedy algorithm (near-optimal, linear scalability)



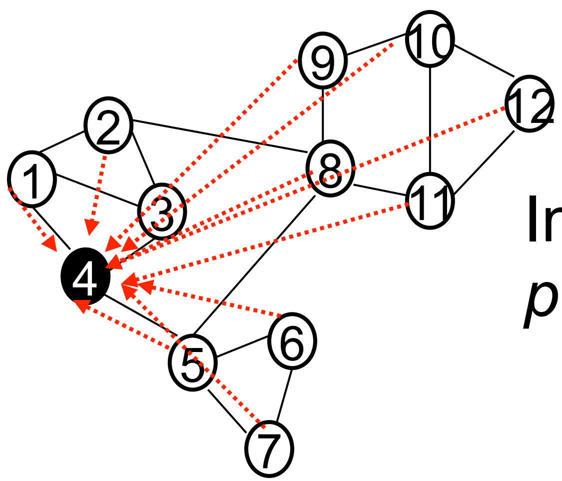
A Special Case of Dragon = Generalized *Netshild*

r = *B* *r*

- Fact 1: The largest eigenvalue of ***B*** is *1*
- Fact 2: ***r*** is the corresponding right eigenvector of ***B***
- Fact 3: The corresponding left eigenvector of ***B*** is ***1***

For *w=2, g(S)*~=drop in the largest eigenvalue of ***B***

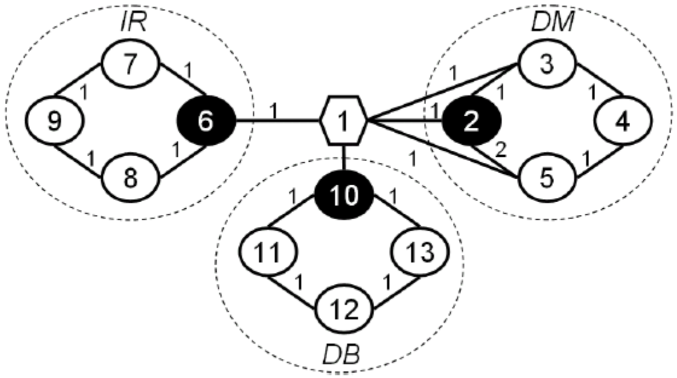
- Dragon (w=2) = Netshield on directed graphs



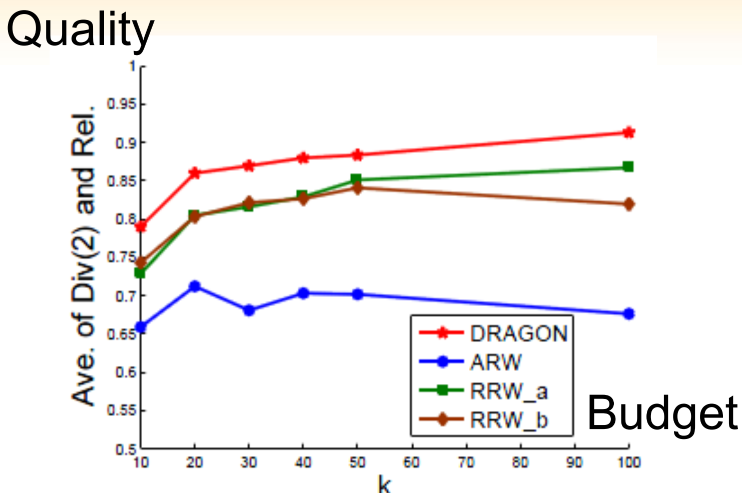
Intuition: find *k* nodes to disconnect *the personalized graph* ***B*** as much as possible

Hanghang Tong, Jingrui He, Zhen Wen, Ravi Konuru, Ching-Yung Lin:
Diversified ranking on large graphs: an optimization viewpoint. KDD 2011: 1028-1036

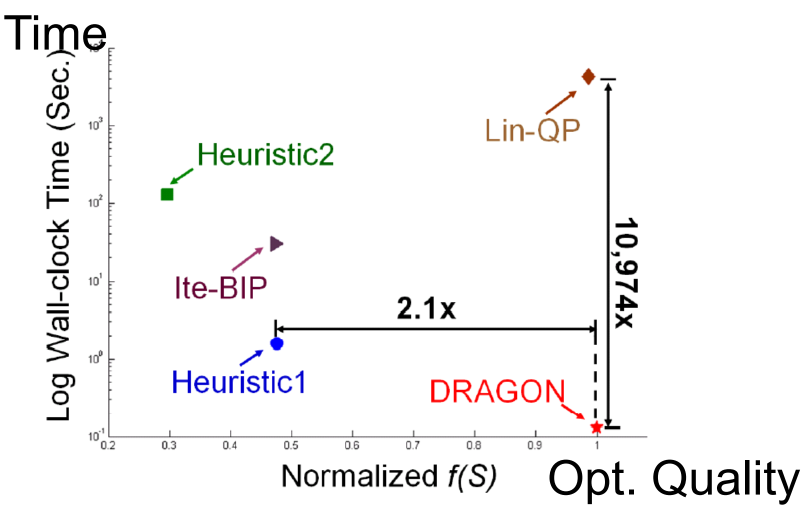
A4: Experimental Results



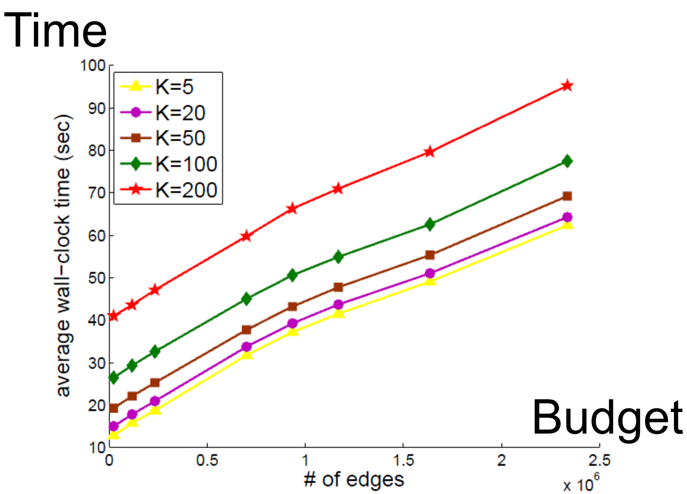
An Illustrative Example



Compare w/ alternative choices



Quality-Time Balance



Scalability

Hanghang Tong, Jingrui He, Zhen Wen, Ravi Konuru, Ching-Yung Lin:
Diversified ranking on large graphs: an optimization viewpoint. KDD 2011: 1028-1036

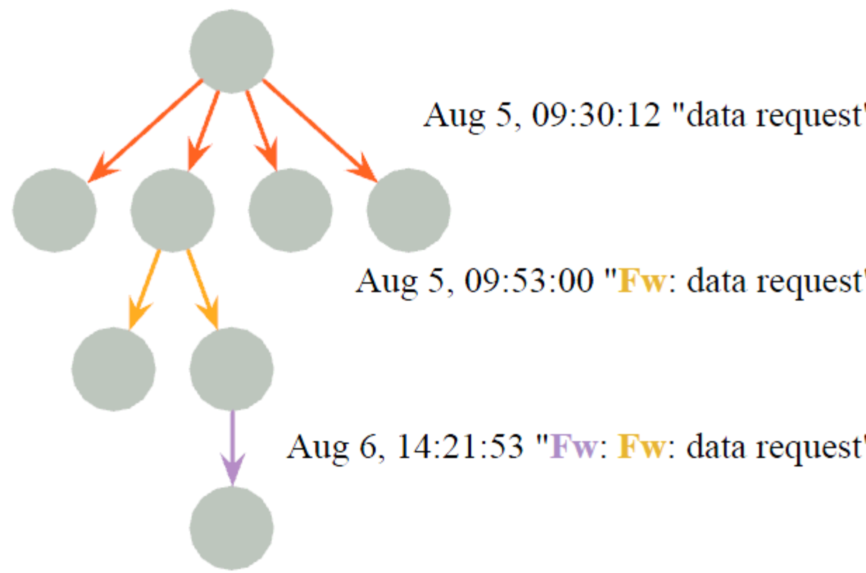
A5: Information Spreading in Context

- Micro-Behavior

Q1: What does information spreading depend on?

- Macro-Behavior

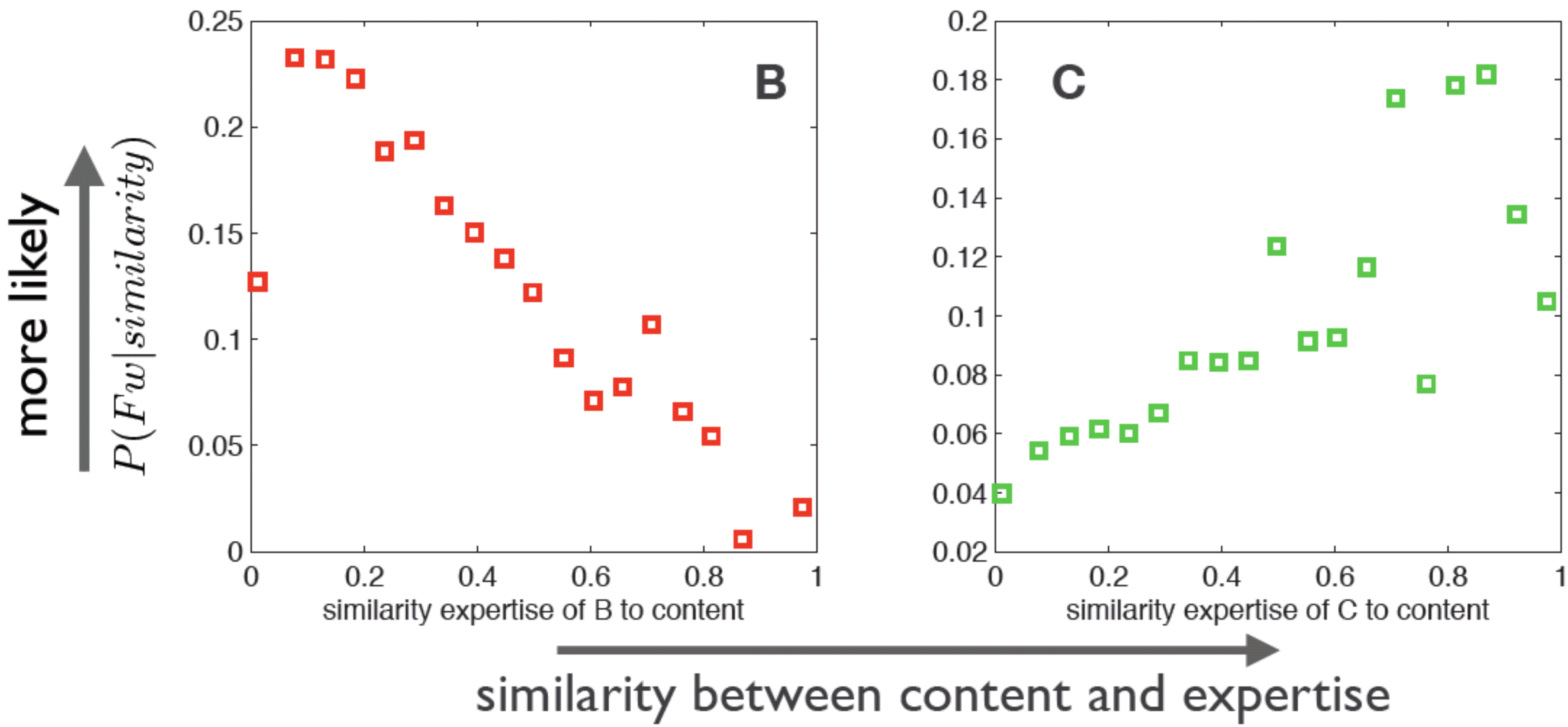
Q2: How does the tree look Like (depth, width, size), and why?



Data: 8000+ IBM employees emails, 2000+ Fw threads, information about the individuals (performance, dept, job role), content of emails

Dashun Wang, Zhen Wen, Hanghang Tong, Ching-Yung Lin, Chaoming Song, Albert-László Barabási: Information spreading in context. WWW 2011: 735-744

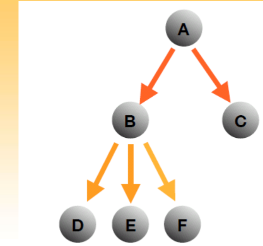
A5: Information Spread (*whether or not*) vs. Content

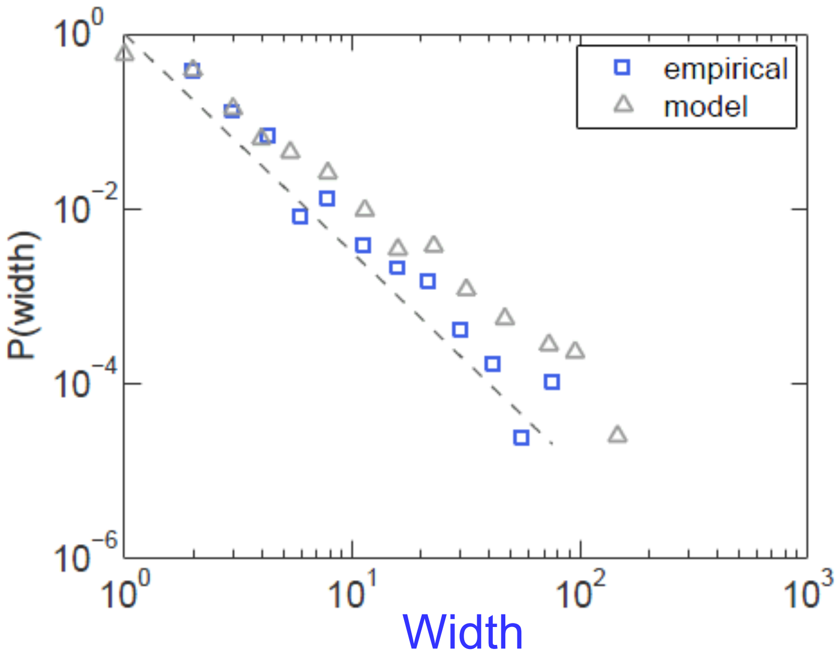


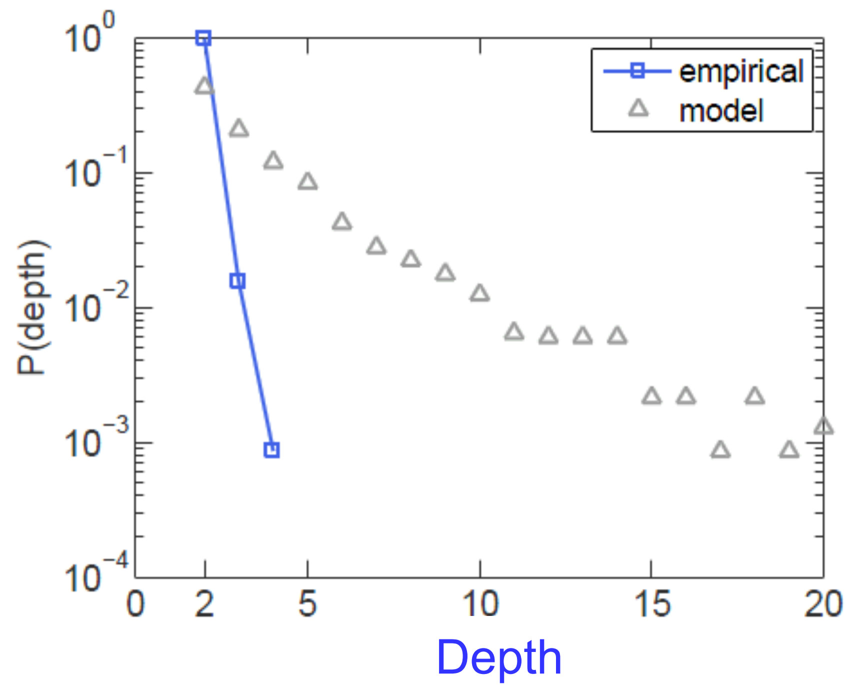
Information is more likely non-expert → expert

Dashun Wang, Zhen Wen, Hanghang Tong, Ching-Yung Lin, Chaoming Song, Albert-László Barabási: Information spreading in context. WWW 2011: 735-744

A5: The Structure of Information Spreading





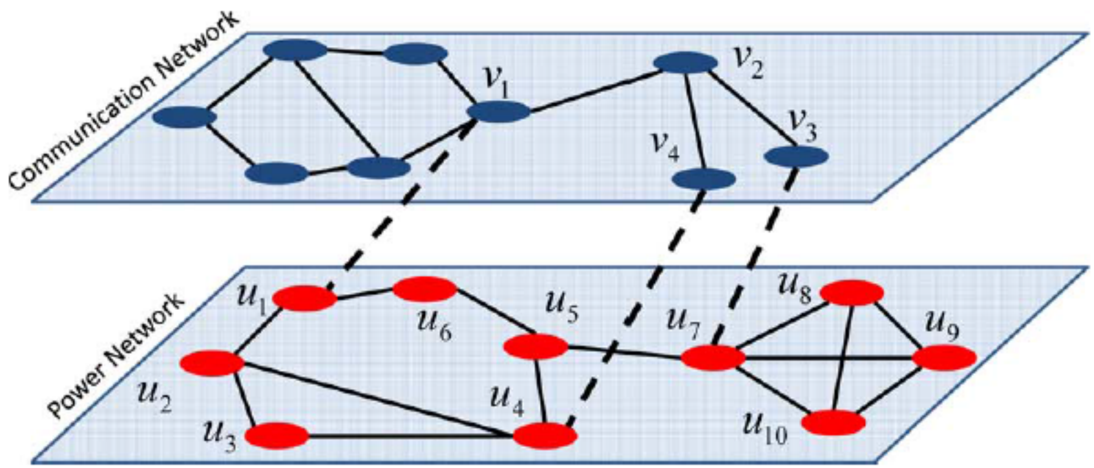


1) The trees are ***fat and shallow*** (instead of ***thin and deep*** as in Kleinberg's chain-letter setting)
2) Can be explained by a simple branch model (w/ decaying branching factors)

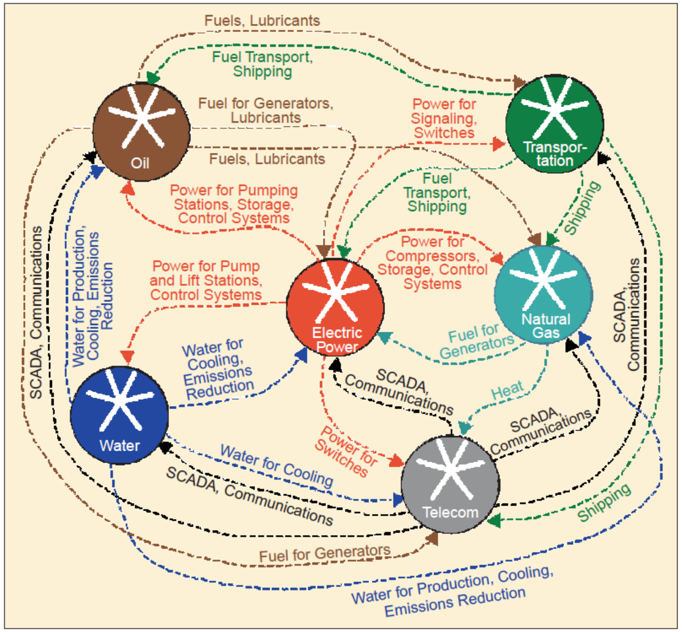
Dashun Wang, Zhen Wen, Hanghang Tong, Ching-Yung Lin, Chaoming Song, Albert-László Barabási: Information spreading in context. WWW 2011: 735-744

A6: Vulnerability of Cyber-Physical Systems

- A Two-layered CPS
  - Blue: communication networks
  - Red: Power grid
  - Dashed line: cross-layer inter-dependency



- Examples of Infrastructure Interdependencies



- Q: which node(s) and/or link(s) dysfunctions will lead to a catastrophic failure of the entire system?

- Rinaldi, Steven M., James P. Peerenboom, and Terrence K. Kelly. "Identifying, understanding, and analyzing critical infrastructure interdependencies." Control Systems, IEEE 21.6 (2001): 11-25.
- Nguyen, Duy T., Yilin Shen, and My T. Thai. "Detecting critical nodes in interdependent power networks for vulnerability assessment." Smart Grid, IEEE Transactions on 4.1 (2013): 151-159.
- Vespignani, Alessandro. "Complex networks: The fragility of interdependency." Nature 464.7291 (2010): 984-985.

A7: Team Member Replacement

Problem Definition:

Given: (1) A labelled social network $G := \{A, L\}$

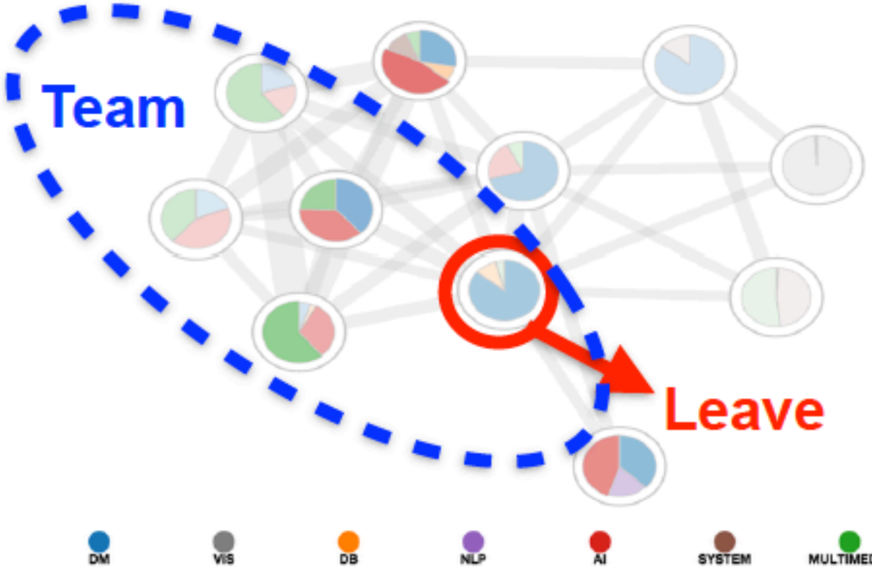
Adj. Matrix

Skill Indicator

(2) A team $G(\mathcal{T})$

(3) A team member $p \in \mathcal{T}$

Recommend: A "best" alternative $q \notin \mathcal{T}$ to replace the person p's role in the team $G(\mathcal{T})$

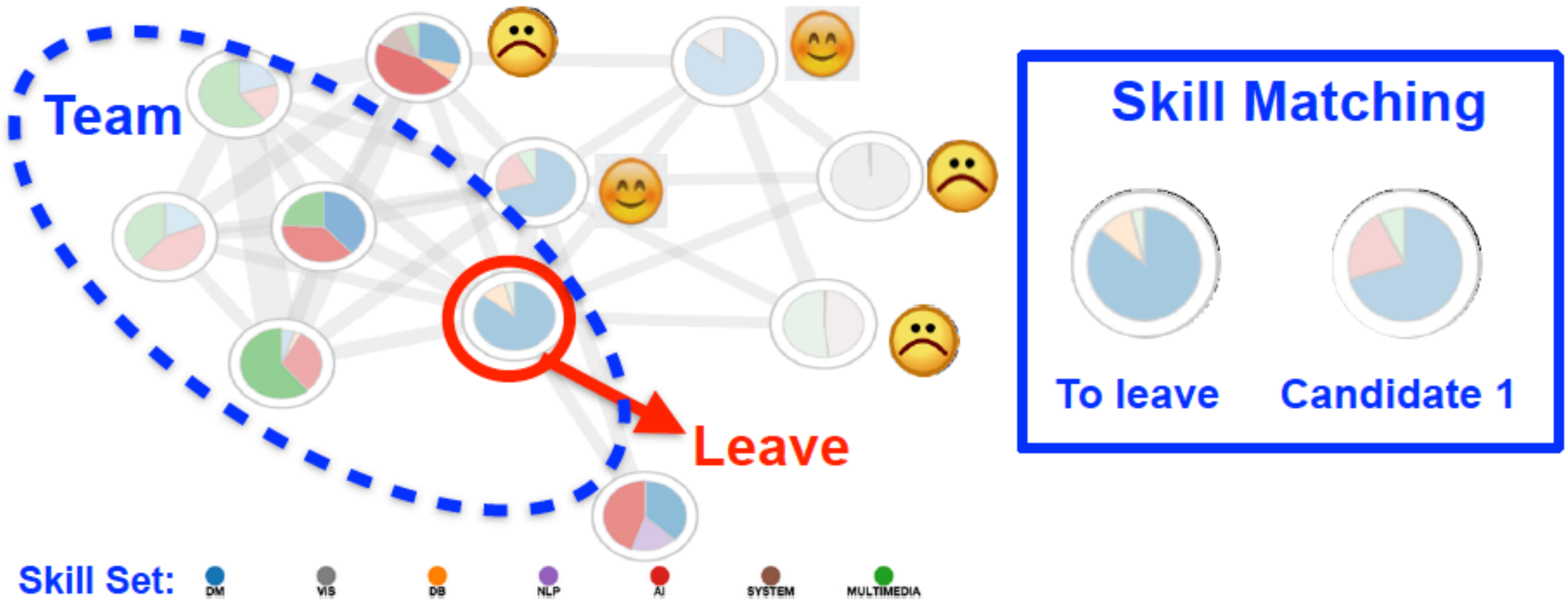


Q: who is a good candidate to replace the person to leave

- Liangyue Li, Hanghang Tong, Nan Cao, Kate Ehrlich, Yu-Ru Lin, Norbou Buchler:Replacing the Irreplaceable: Fast Algorithms for Team Member Recommendation. WWW 2015: 636-646

A7: Team Member Replacement

Objective 1: A good candidate should have a similar skill set

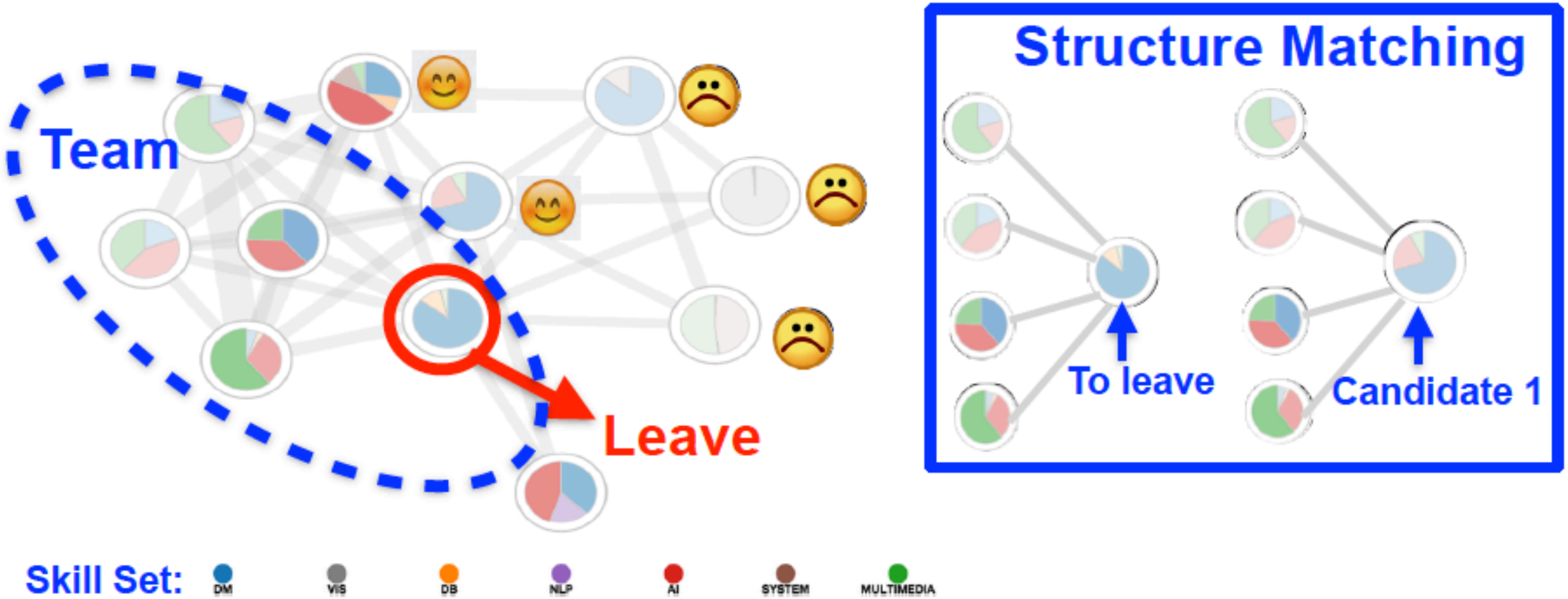


New team will have similar skill set as the old team to complete the task

- Liangyue Li, Hanghang Tong, Nan Cao, Kate Ehrlich, Yu-Ru Lin, Norbou Buchler:Replacing the Irreplaceable: Fast Algorithms for Team Member Recommendation. WWW 2015: 636-646

A7: Team Member Replacement

Objective 2: A good candidate should have a similar network structure

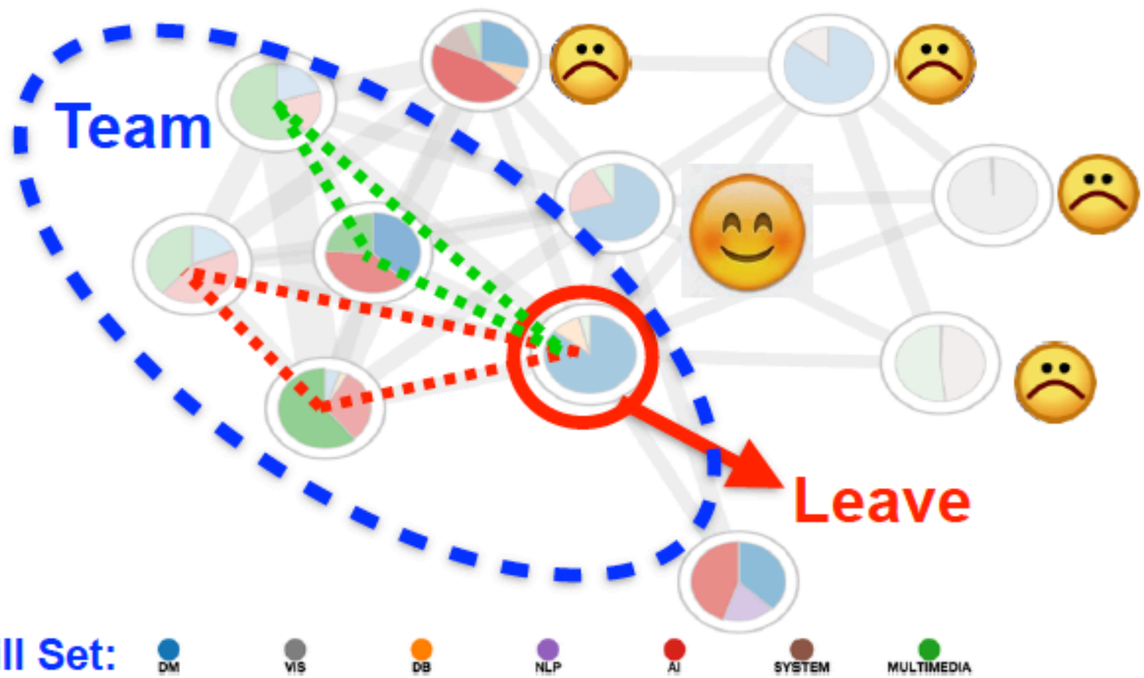


New team will have similar network structure as the old team to collaborate effectively

- Liangyue Li, Hanghang Tong, Nan Cao, Kate Ehrlich, Yu-Ru Lin, Norbou Buchler:Replacing the Irreplaceable: Fast Algorithms for Team Member Recommendation. WWW 2015: 636-646

A7: Team Member Replacement

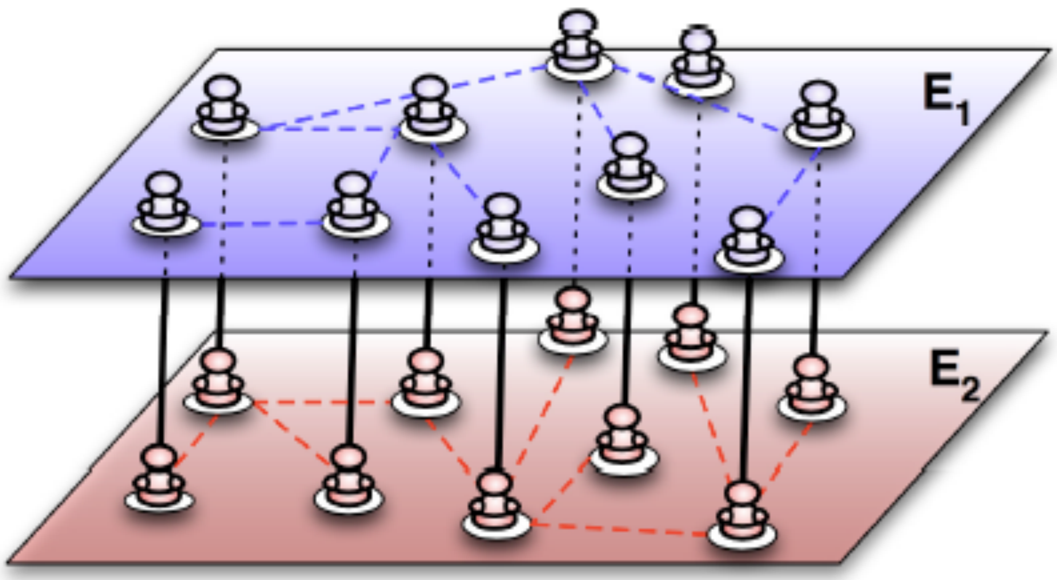
The two objectives should be fulfilled simultaneously!



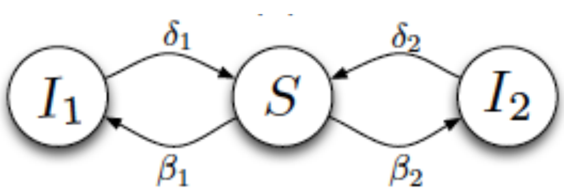
New team will have similar skill and communication configuration for each sub-task

- Liangyue Li, Hanghang Tong, Nan Cao, Kate Ehrlich, Yu-Ru Lin, Norbou Buchler:Replacing the Irreplaceable: Fast Algorithms for Team Member Recommendation. WWW 2015: 636-646

A8: Competitive Virus on Composite Networks

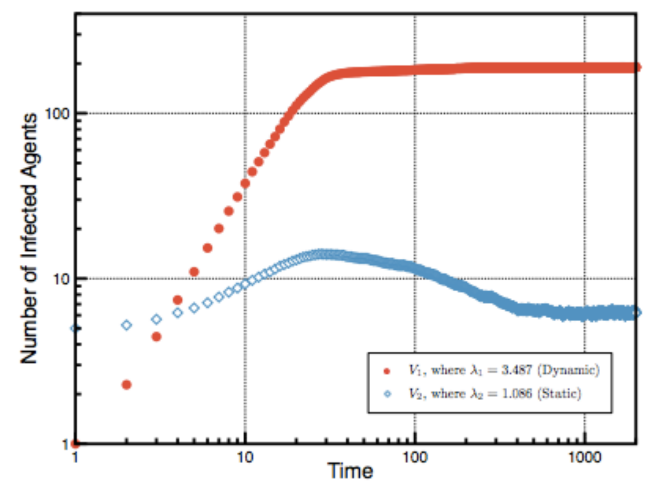


An example of composite network: a single set of nodes with two distinct sets of links



Virus Model: $S\ I_1\ I_2\ S$

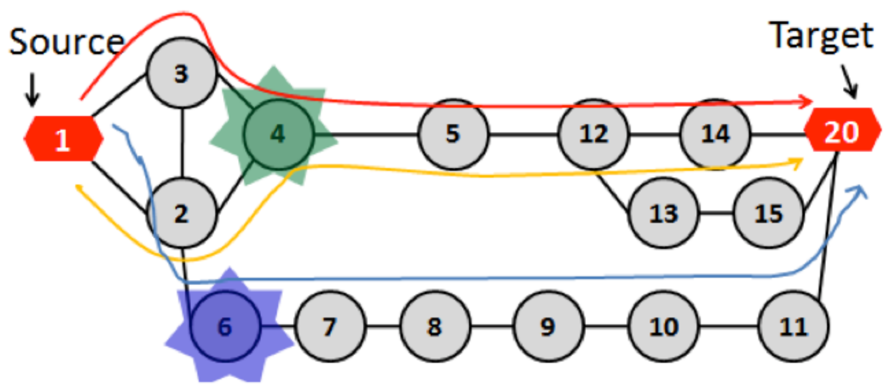
- Q: Which virus will win?
  - `virus': smartphone malware, memes, ideas
- A: if $\lambda_1 > \lambda_2$ V1 will win.
  - λ_1 and λ_2: leading eigen-values of system matrices.
- Results

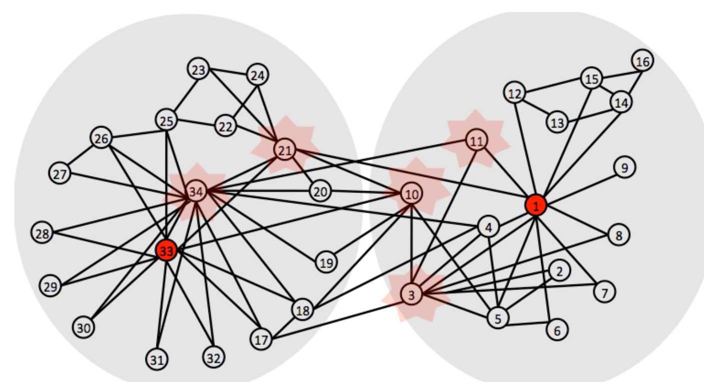


- Xuetao Wei, Nicholas Valler, B. Aditya Prakash, Iulian Neamtiu, Michalis Faloutsos, Christos Faloutsos: Competing Memes Propagation on Networks: A Network Science Perspective. IEEE Journal on Selected Areas in Communications 31(6): 1049-1060 (2013)

A9: Gateway Finder

- **Problem Definition**: Given a source (s) or a source group; and a target (t) or a target group,
  - **Q1 (Metric)**: how to measure the gateway-ness for a subset of nodes (I)?
  - **Q2 (Algorithm):** how to find a subset of k nodes with highest gateway-ness score?
- **Solutions**: Find the set whose removal causes maximal decrease of the proximity from source to target (e.g., block most paths).



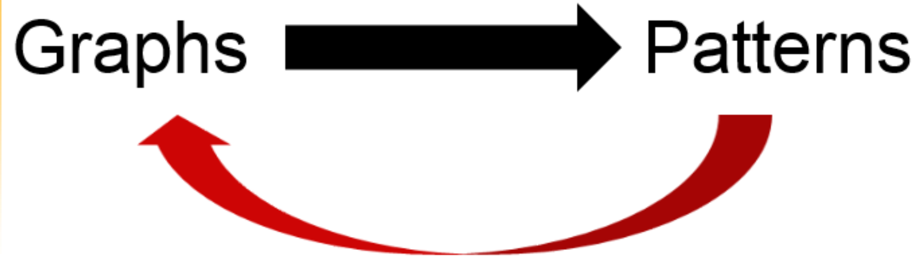


• Hanghang Tong, Spiros Papadimitriou, Christos Faloutsos, Philip S. Yu, Tina Eliassi-Rad: Gateway finder in large graphs: problem definitions and fast solutions. Inf. Retr. 15(3-4): 391-411 (2012)

Part IV: Future Trends

- N1: Learn *k* in GCO Problem
- N2: Sense-Making of GCO: How/Why?
- N3: GCO Tracking & Attribution
- N4: GCO on Multi-layered Networks
- N5: Min-Max GCO Problem
- N6: Super-Robust Network Problem
- N7: Optimal Graph Construction Problem
- N8: GCO Scalability: Challenges & Opportunities

N1: Learn *k* in GCO



Graph Connectivity Optimization (GCO) - This Tutorial

Given:
(1) an initial graph
(2) a graph operation (e.g., deleting ***k*** nodes, adding ***k*** new links)
(3) a mining task

Find:
an 'optimal' graph

- **Q**: what is the minimum ***k***, to reduce the epidemic threshold below 1, given the strength of the virus and connectivity of the population?

N2: Sense-Making of GCO:
What/Who → How/Why?

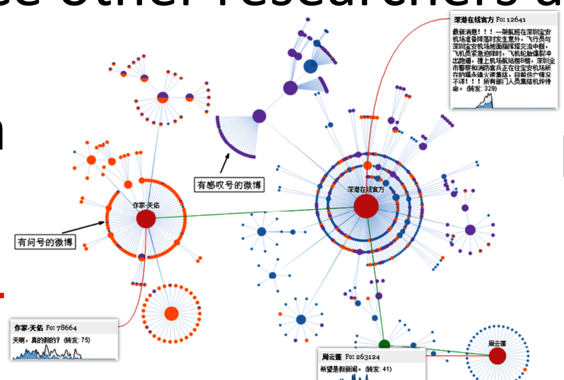
Graph Connectivity Optimization (GCO) - This Tutorial

Given:
(1) an initial graph
(2) a graph operation (e.g., deleting ***k*** nodes, adding ***k*** new links)
(3) a mining task

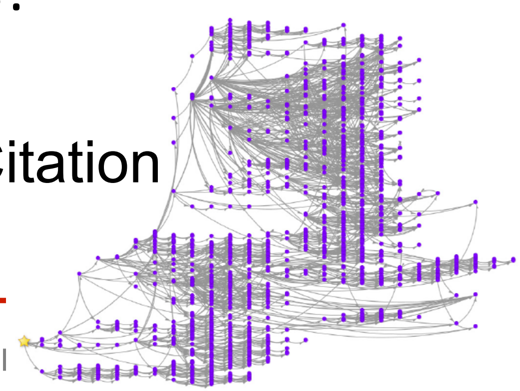
→ **Find**: an 'optimal' graph

- **Current: A Typical GCO Instance**
  - **Given:** a social network,
  - **Find:** ***who*** or ***which links*** are the most important, in bridging different communities?
- **Next: From Who/Who to How/Why**
  - **Q1**: Given an critical power-line in power-grid, explain ***why*** it is important (in maintaining the graph connectivity)
  - **Q2**: Given an influential author in scholarly network, find ***how*** s/he influence other researchers and/or fields?

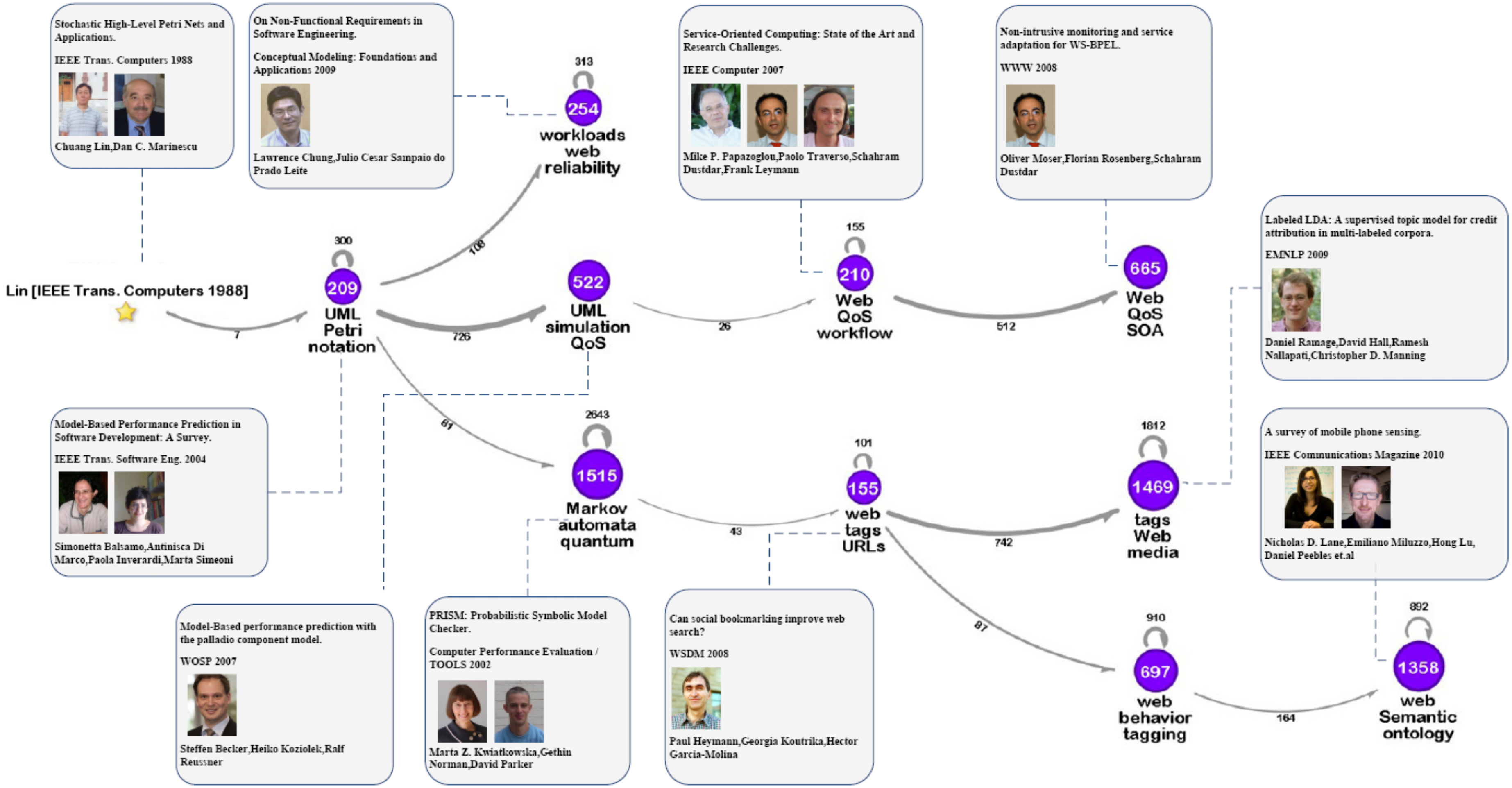
Retweeting Graph in Chinese Weibo



Reversed Citation Graph



N2: A Flow-based Summarization Solution



The influence graph of “Stochastic High-Level Petri Net and Applications”

- Lei Shi, Hanghang Tong, Jie Tang, Chuang Lin: Flow-Based Influence Graph Visual Summarization. ICDM 2014: 983-988

N3: GCO Tracking & Attribution

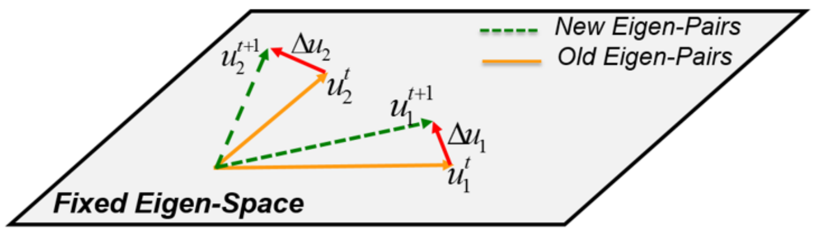
Graph Connectivity Optimization (GCO) - This Tutorial

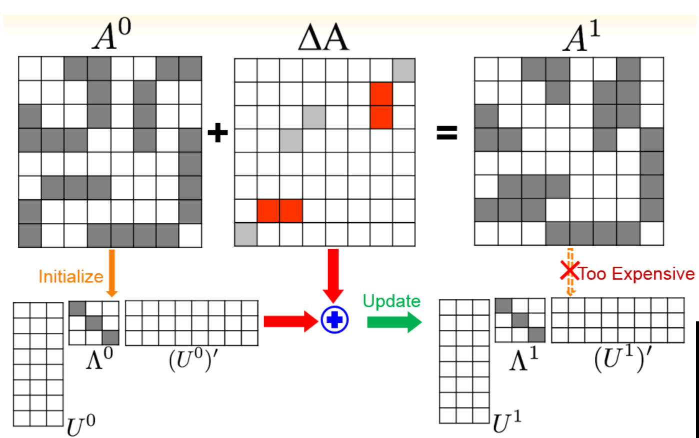
Given:
(1) an initial graph
(2) a graph operation (e.g., deleting ***k*** nodes, adding ***k*** new links)
(3) a mining task

→ **Find**: an 'optimal' graph

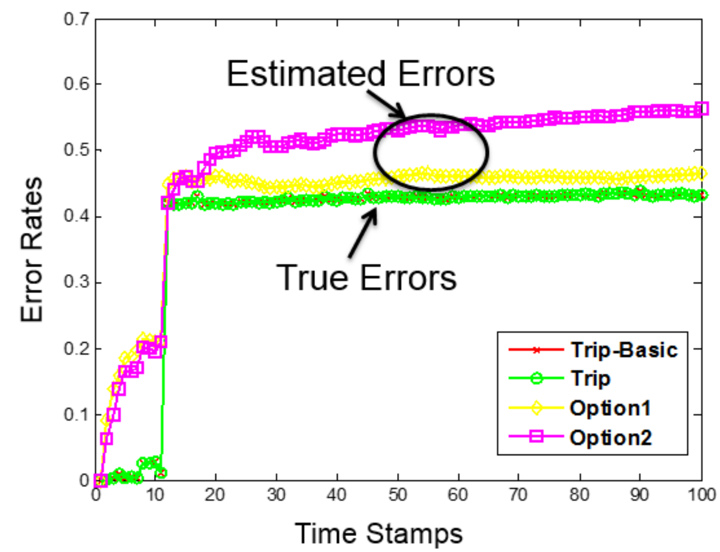
- **Observations**
  - **#1:** Graphs are changing over time
  - **#2:** Many graph connectivity measures can be expressed as an ***eigen-function*** of the adjacency matrix

- **Solutions: Tracking eigen-function**





- **Results**



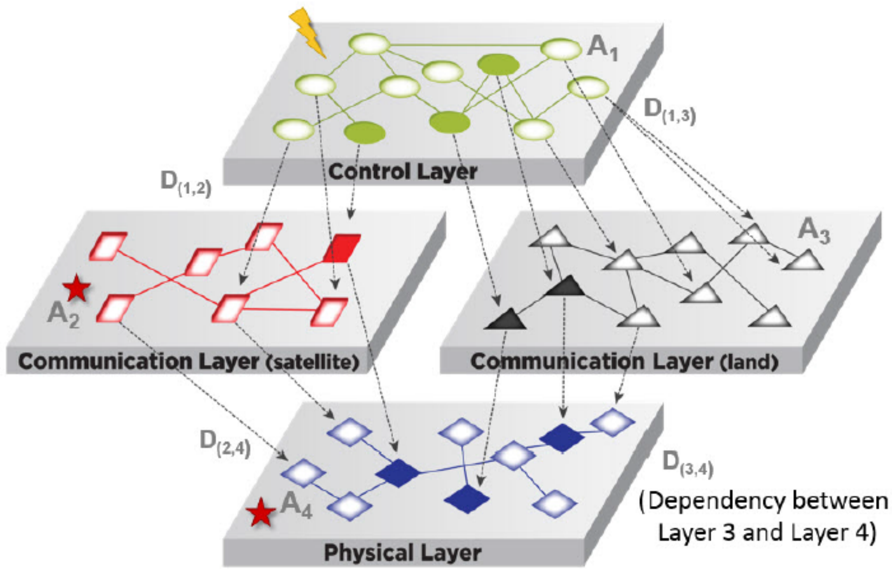
- C. Chen and H. Tong: "Fast Eigen-Functions Tracking on Dynamic Graphs". SDM 2015

Graph Connectivity Optimization (GCO) - This Tutorial

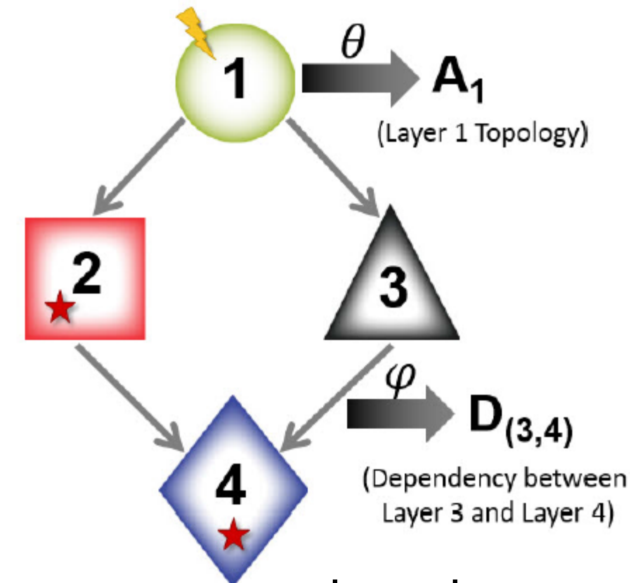
Given:
(1) an initial graph
(2) a graph operation (e.g., deleting ***k*** nodes, adding ***k*** new links)
(3) a mining task

→ **Find**: an ‘optimal’ graph

N4: GCO on Multi-layered Networks



A four-layered network

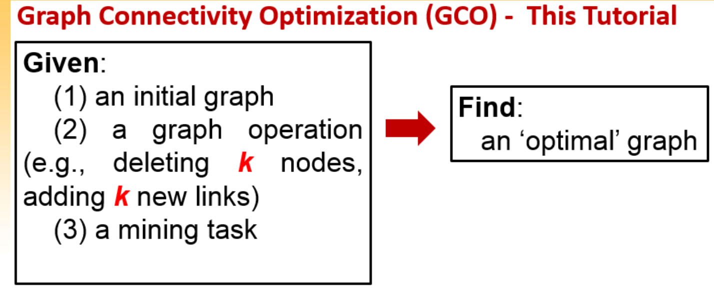


layer-layer dependency network

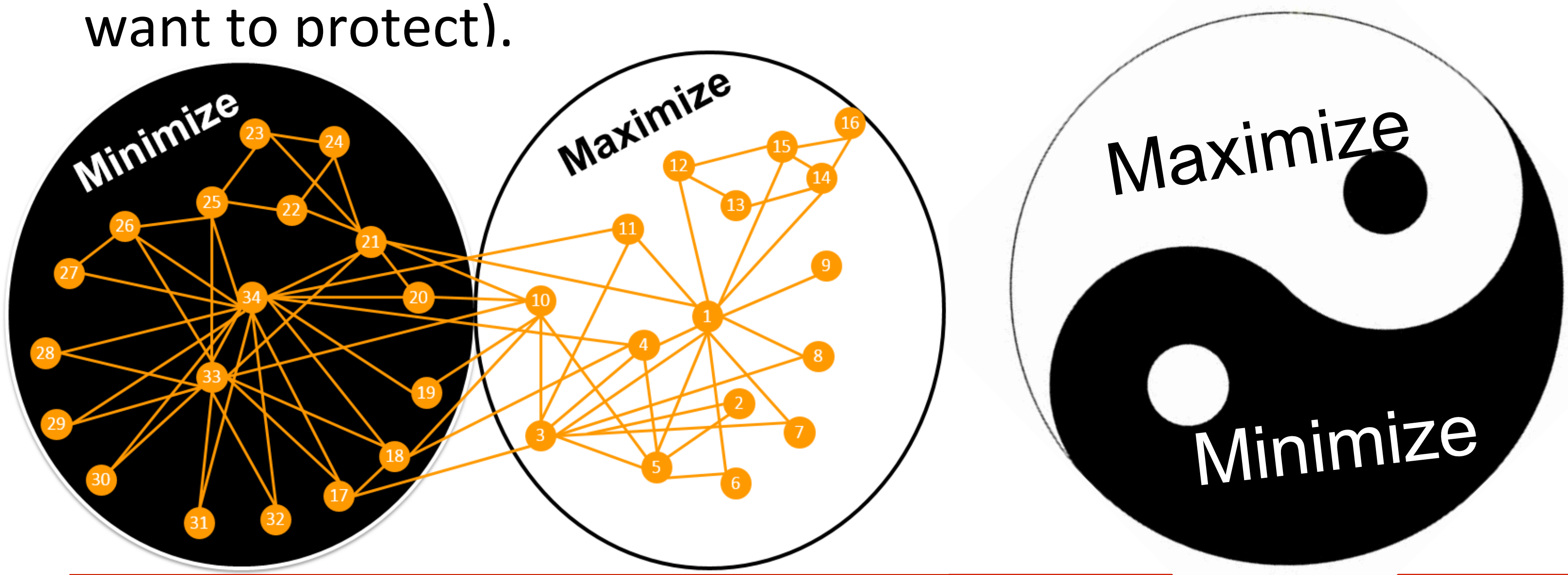
- **A Multi-layered Network Model (Mulan)**
  - A Quintuple: $\Gamma = < \mathbf{G}, \mathcal{A}, \mathcal{D}, \theta, \varphi >$
- **Q:** How to find an optimal node set in the ***control layer***, to minimize the connectivity of the ***target layer(s)***?

- C. Chen, J. He, N. Bliss and H. Tong: “On the Connectivity of Multi-layered Networks: Models, Measures and Optimal Control” ICDM 2015.

N5: Min-Max GCO Problem (Angels & Demons)



- **Given:** two inter-connected networks (or two inter-connected components within the same network);
- **Find:** the optimal graph operation, that
  - *minimizes* the connectivity of the (adversarial) network, and
  - *maximizes* the connectivity of the other network (the one we want to protect).



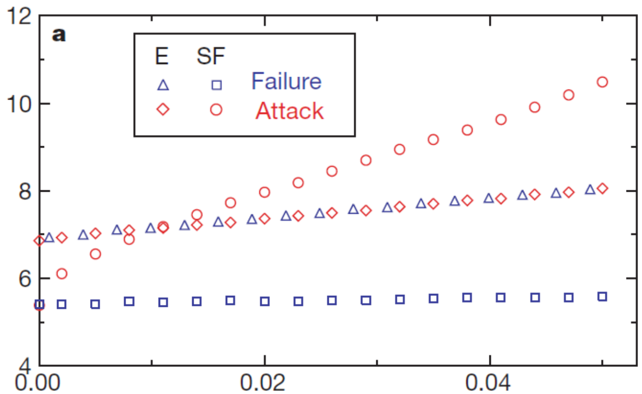
N6: Super-Robust Network

Graph Connectivity Optimization (GCO) - This Tutorial

Given:
(1) an initial graph
(2) a graph operation (e.g., deleting ***k*** nodes, adding ***k*** new links)
(3) a mining task

Find:
an 'optimal' graph

- **Observations** (Nature 2000):
  - **Scale-free Networks** (e.g., power-law): resilient to random failure, but vulnerable to targeted attack
  - **Exponential Networks** (e.g., ER, Small-World model): resilient to targeted attacks.



- X: fraction of removed nodes
- Y: diameter of the residual network
- E: ER model; SF: scale-free
- Blue: (random) failure
- Red: (intentional) attack

- **Q1:** How to design a robust network that is resilient to both failure and attacks?
- **Q2:** If we know the type of attack (e.g., HDA, or even based on GCO algorithms), How to tailor the GCO-defending algorithms (e.g., knowing your enemies)?

- Albert, Réka, Hawoong Jeong, and Albert-László Barabási. "Error and attack tolerance of complex networks." nature 406.6794 (2000): 378-382.
- István A. Kovács & Albert-László Barabási: Network science: Destruction perfected. Nature 524, 38–39, 2015

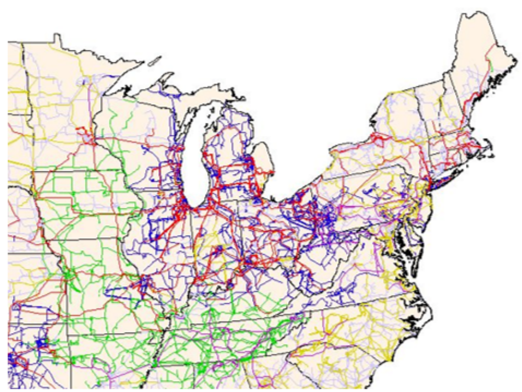
N7: Optimal Graph Construction

Graph Connectivity Optimization (GCO) - This Tutorial

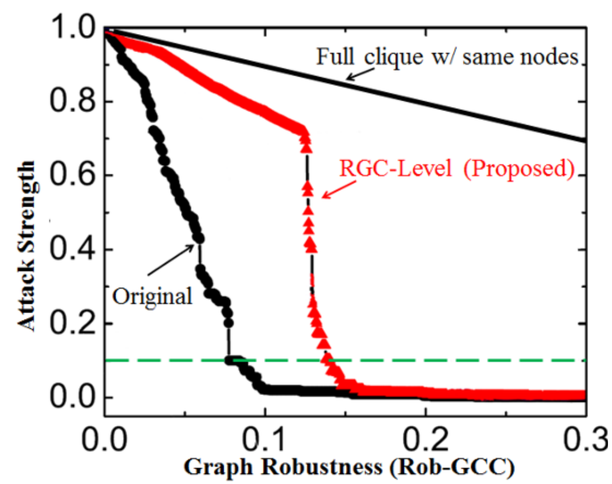
Given:
(1) an initial graph
(2) a graph operation
(3) a mining task

Find:
an 'optimal' graph

- **Q: What if the initial graph does not exist?**
- **Robust Network Construction again intentional attacks (e.g., HDA)**
  - **Given:** (1) the number of nodes *n* of the graph, and (2) its desired degree vector *d* (i.e., node capacity);
  - **Output:** a graph *A* with (1) *n* nodes, (2) the maximal robustness, (3) deg(*A*) = *d*
- **An Effective Heuristic**
  - **H1:** Avoid disassortative mix by degree
  - **H2:** Large loop coverage



US Power Grid



• Hui Wang, Wanyun Cui, Yanghua Xiao, Hanghang Tong: Robust network construction against intentional attacks. BigComp 2015: 279-286

N8: GCO Scalability: Challenges & Opportunities

- **Challenges: How to Scale-up & Speed-up**
  - E1: O(m) or better on a single machine
  - E2: Parallelism (implementation, decouple, analysis)
- **Opportunities**:
  - Solving GCO problems **trivially** by scale?
  - **Conjecture**: when the initial graph is big enough, (1) adding any new links will make little improvement, and (2) the graph becomes impossible to demolish with any limited budget.
  - Is this true? If so, where is the tipping point?

Acknowledgement

Lada A. Adamic, Leman Akoglu, Albert-László Barabási, Norbou Buchler, Nadya Bliss, Nan Cao, Polo Chau, Tina Eliassi-Rad, Kate Erhlich, Christos Faloutsos, Michalis Faloutsos, Jingrui He, Theodore J. Iwashyna, Yu-Ru Lin, Qiaozhu Mei, B. Aditya Prakash, Lei Shi, Chaoming Song, Boleslaw K. Szymanski, Jie Tang, Dashun Wang, Yanghua Xiao, Lei Xie, Lei Ying

Reference

- Hanghang Tong, B. Aditya Prakash, Tina Eliassi-Rad, Michalis Faloutsos, Christos Faloutsos: Gelling, and melting, large graphs by edge manipulation. CIKM 2012: 245-254
- Hui Wang, Wanyun Cui, Yanghua Xiao, Hanghang Tong: Robust network construction against intentional attacks. BigComp 2015: 279-286
- Lei Shi, Hanghang Tong, Jie Tang, Chuang Lin: Flow-Based Influence Graph Visual Summarization. ICDM 2014: 983-988
- B. Aditya Prakash, Lada Adamic, Theodore Iwashnya, Hanghang Tong and Christos Faloutsos: Fractional Immunization on Networks. SDM 2013
- Hau Chan, Leman Akoglu, Hanghang Tong: Make It or Break It: Manipulating Robustness in Large Networks. SDM 2014: 325-333

Reference

- Hanghang Tong, Spiros Papadimitriou, Christos Faloutsos, Philip S. Yu, Tina Eliassi-Rad: Gateway finder in large graphs: problem definitions and fast solutions. Inf. Retr. 15(3-4): 391-411 (2012)

- Hanghang Tong, Jingrui He, Zhen Wen, Ravi Konuru, Ching-Yung Lin: Diversified ranking on large graphs: an optimization viewpoint. KDD 2011: 1028-1036

- Nicholas Valler, B. Aditya Prakash, Hanghang Tong, Michalis Faloutsos, Christos Faloutsos: Epidemic Spread in Mobile Ad Hoc Networks: Determining the Tipping Point. Networking (1) 2011: 266-280

- Dashun Wang, Zhen Wen, Hanghang Tong, Ching-Yung Lin, Chaoming Song, Albert-László Barabási: Information spreading in context. WWW 2011: 735-744

- Hanghang Tong, B. Aditya Prakash, Charalampos E. Tsourakakis, Tina Eliassi-Rad, Christos Faloutsos, Duen Horng Chau: On the Vulnerability of Large Graphs. ICDM 2010: 1091-1096

Reference

- Yao Zhang and B. Aditya Prakash: Scalable Vaccine Distribution in Large Graphs given Uncertain Data. ICDM 2014

  Code available at: http://people.cs.vt.edu/badityap/CODE/UDAV.zip

- L. Le, T. Eliassi-Rad and H. Tong: MET: A Fast Algorithm for Minimizing Propagation in Large Graphs with Small Eigen-Gaps. SDM 2015

- István A. Kovács & Albert-László Barabási: Network science: Destruction perfected. Nature 524, 38–39, 2015

- Rinaldi, Steven M., James P. Peerenboom, and Terrence K. Kelly. "Identifying, understanding, and analyzing critical infrastructure interdependencies." Control Systems, IEEE 21.6 (2001): 11-25.

- Nguyen, Duy T., Yilin Shen, and My T. Thai. "Detecting critical nodes in interdependent power networks for vulnerability assessment." Smart Grid, IEEE Transactions on 4.1 (2013): 151-159.

Reference

- Xuetao Wei, Nicholas Valler, B. Aditya Prakash, Iulian Neamtiu, Michalis Faloutsos, Christos Faloutsos: Competing Memes Propagation on Networks: A Network Science Perspective. IEEE Journal on SAC 31(6): 1049-1060 (2013)

- Liangyue Li, Hanghang Tong, Nan Cao, Kate Ehrlich, Yu-Ru Lin, Norbou Buchler:Replacing the Irreplaceable: Fast Algorithms for Team Member Recommendation. WWW 2015: 636-646

- C. Chen, H. Tong, B. Prakash, C. Tsourakakis, T. Eliassi-Rad, C. Faloutsos, D. Chau: Node Immunization on Large Graphs: Theory and Algorithms. IEEE TKDE 2015

- B. Aditya Prakash, Hanghang Tong, Nicholas Valler, Michalis Faloutsos, Christos Faloutsos: Virus Propagation on Time-Varying Networks: Theory and Immunization Algorithms. ECML/PKDD (3) 2010: 99-114