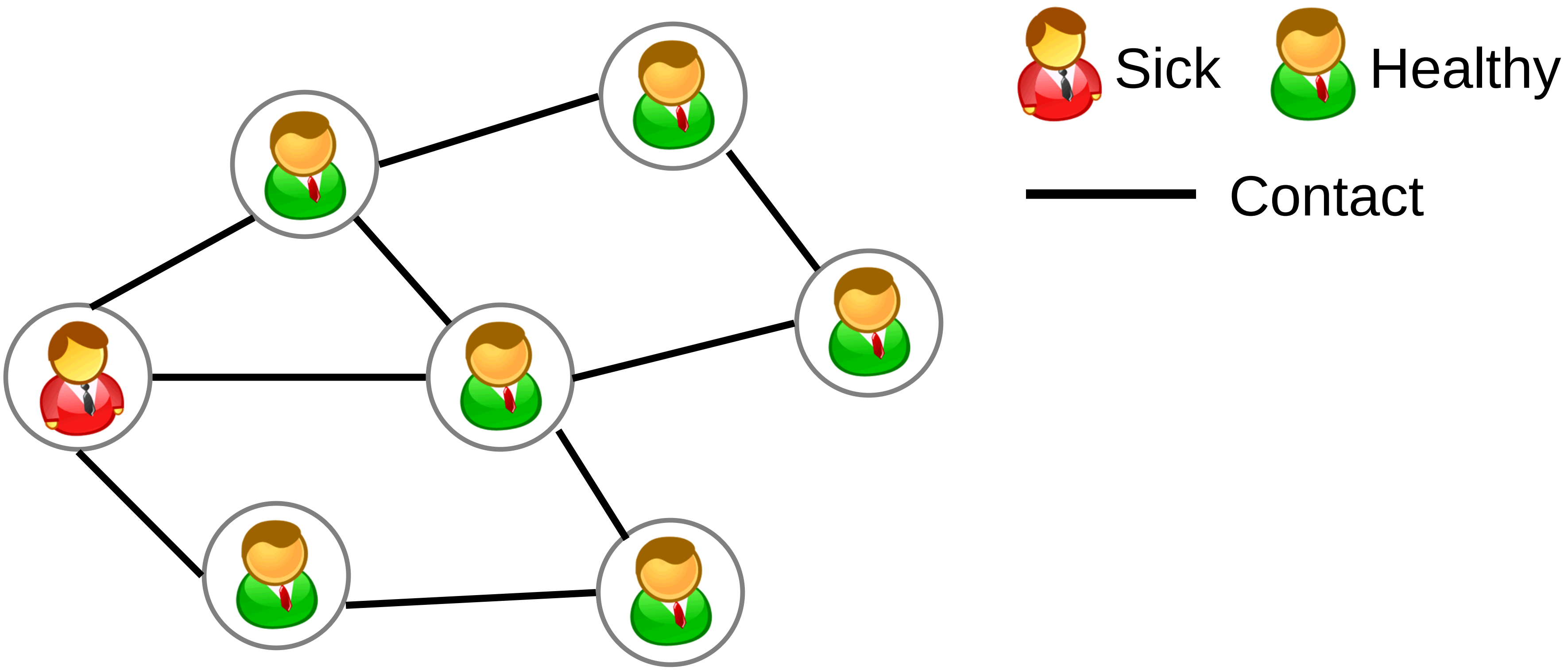
The Dynamics of Dissemination on Graphs: Theory and Algorithms

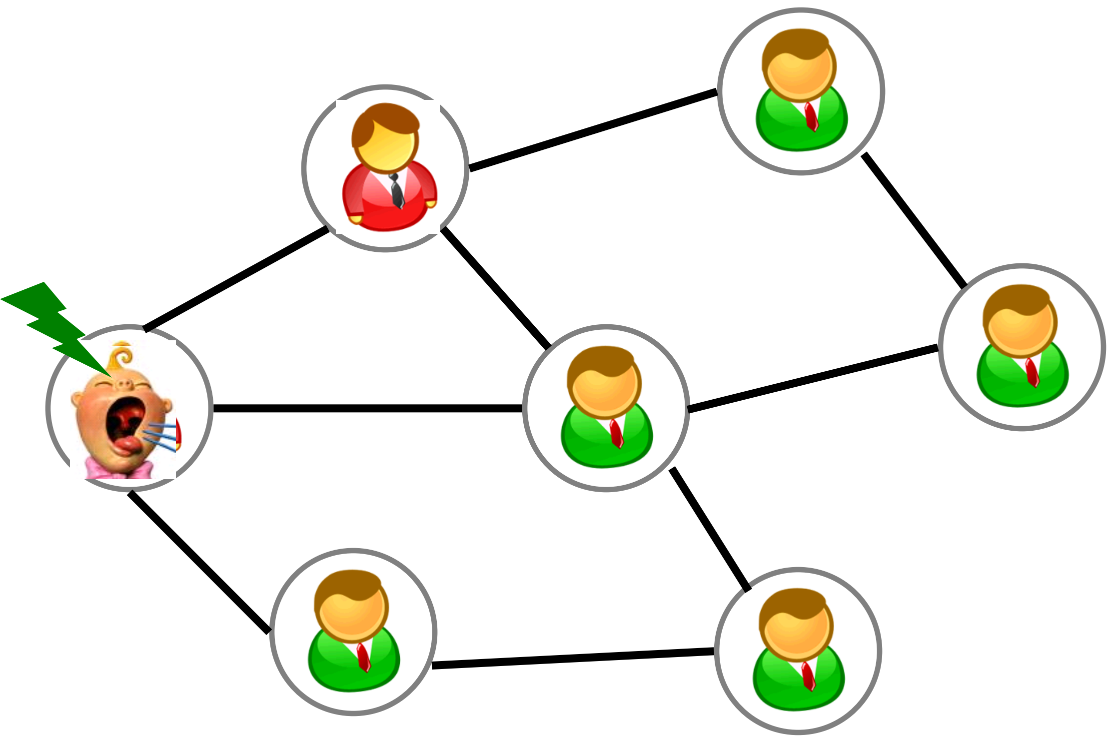
Hanghang Tong

City College, CUNY
Hanghang.tong@gmail.com
http://www-cs.ccny.cuny.edu/~tong/

An Example: Virus Propagation/Dissemination



An Example: Virus Propagation/Dissemination



Sick Healthy

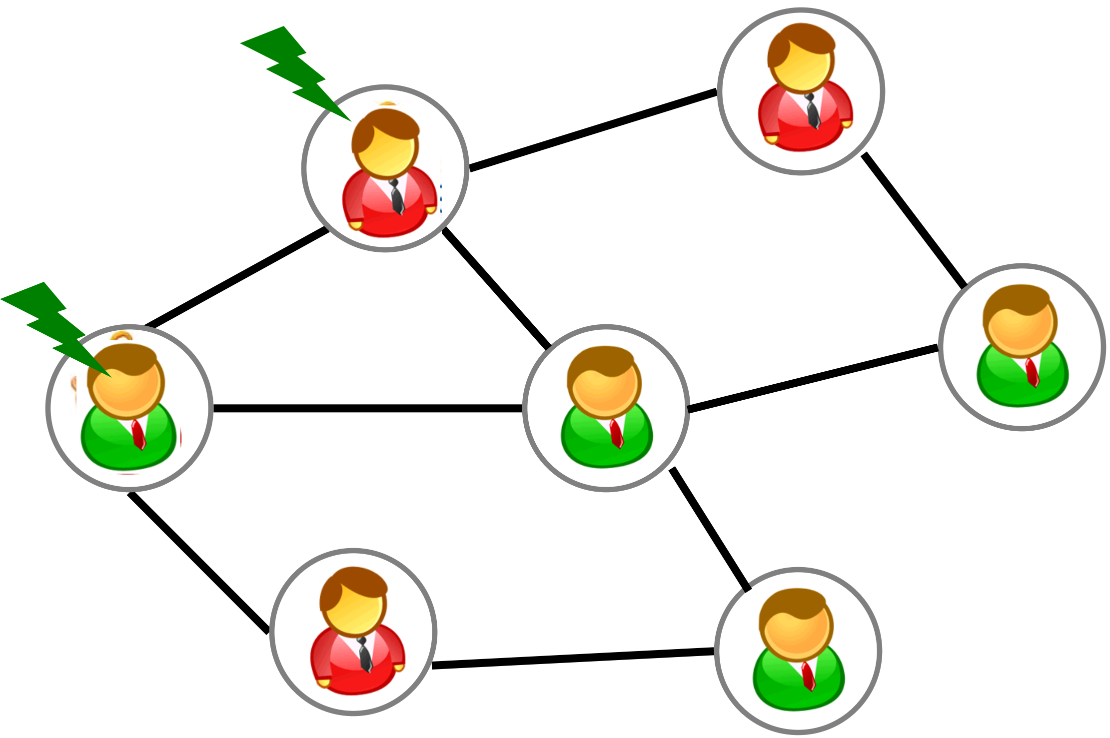
Contact

1: Sneeze to neighbors

2: Some neighbors → Sick

3: Try to recover

An Example: Virus Propagation/Dissemination



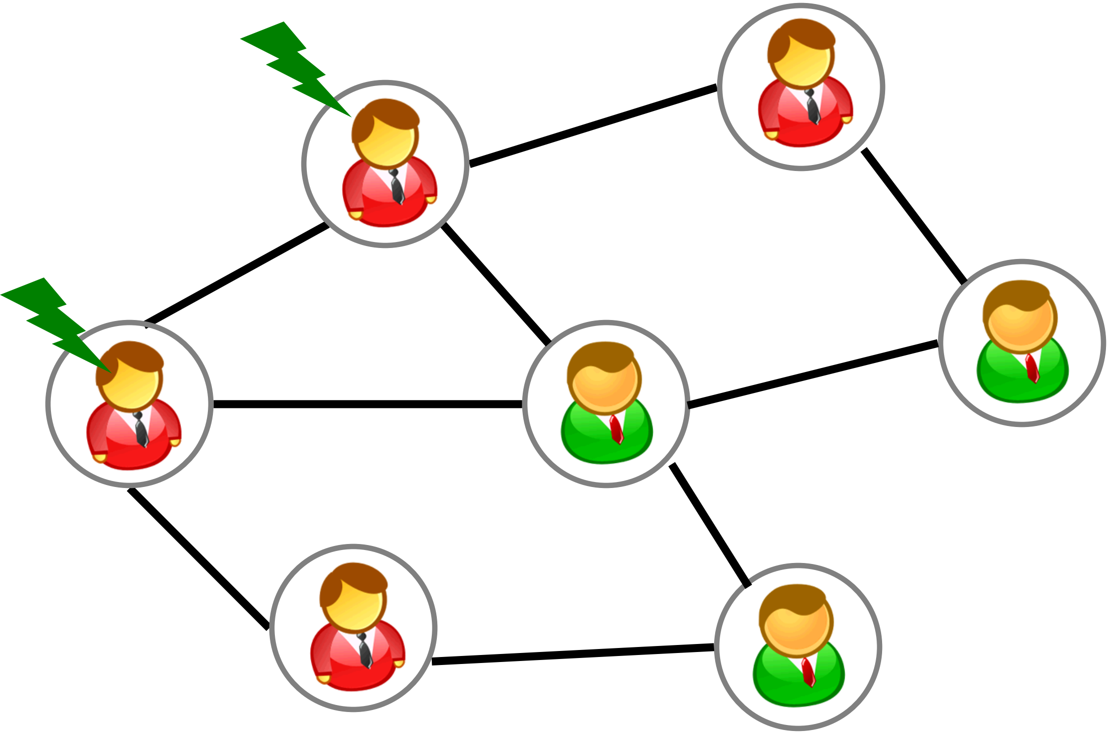
1: Sneeze to neighbors

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

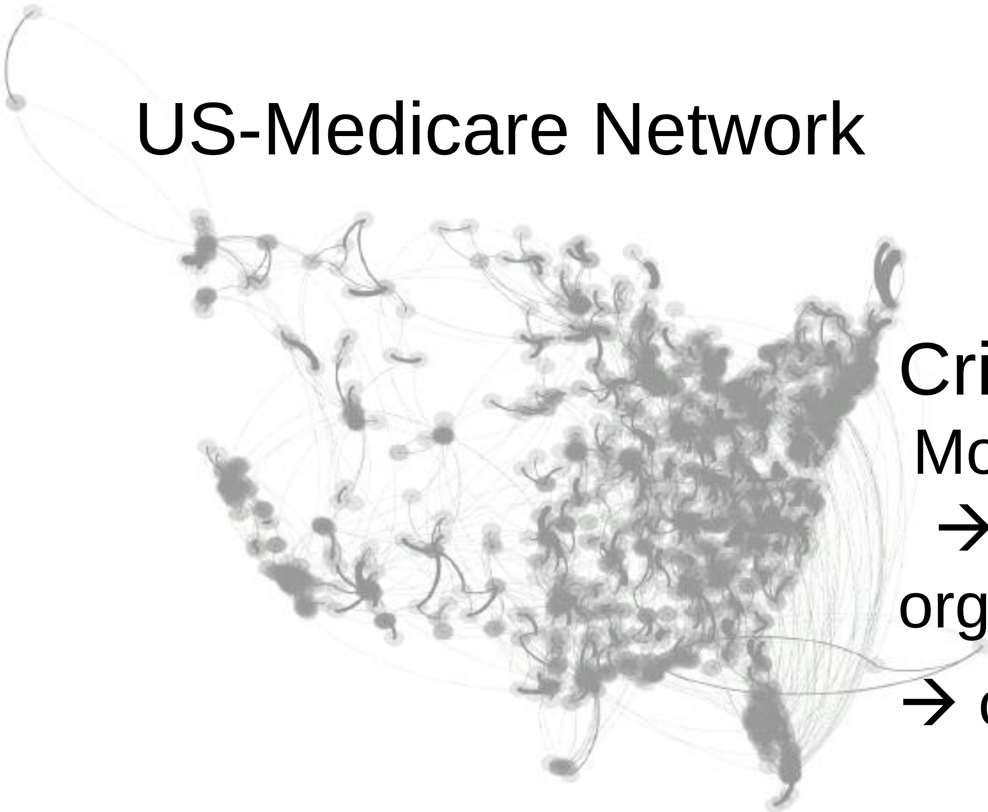
Q: How to minimize infected population?

- Q1: Understand tipping point
- Q2: Minimize the propagation
- Q3: Maximize the propagation

Why Do We Care? – Healthcare

[SDM'13b]

US-Medicare Network



Critical Patient transferring

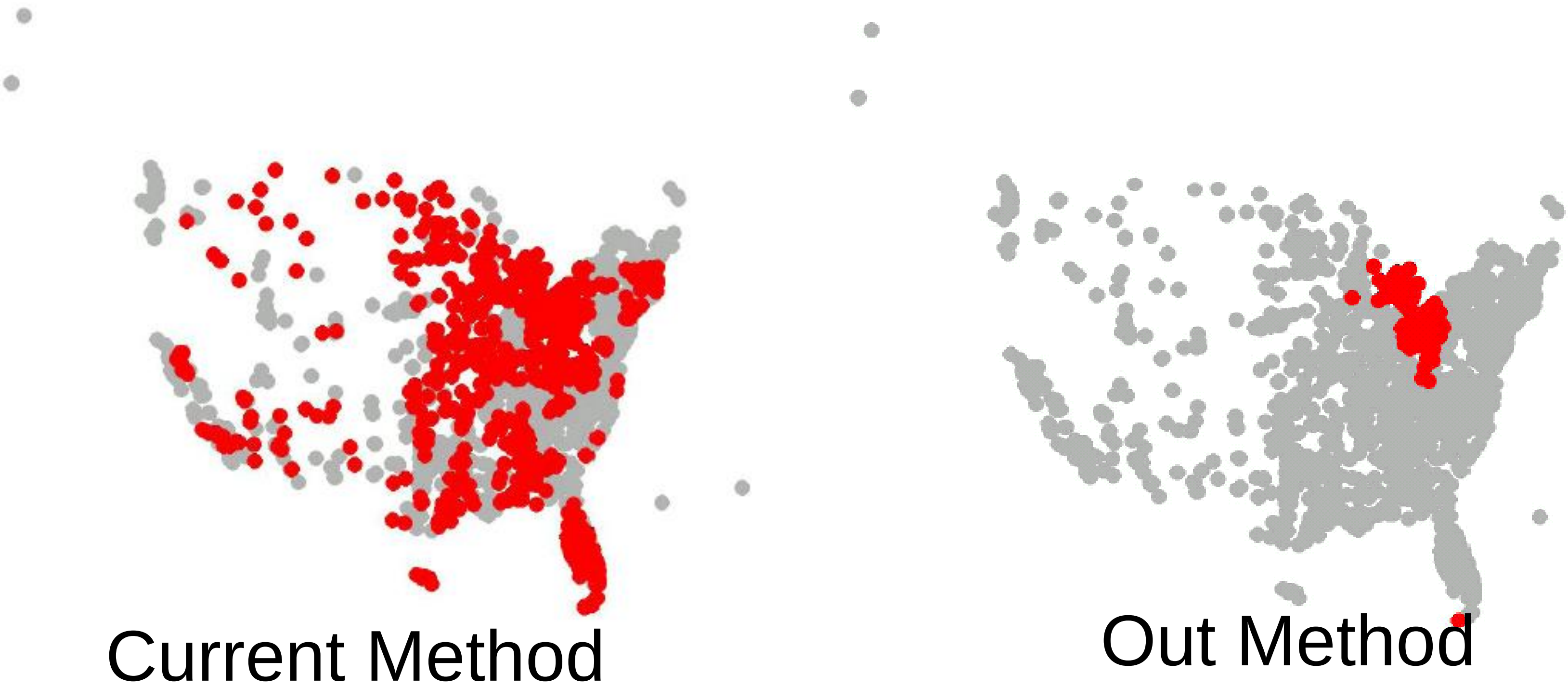
Move patients → specialized care → highly resistant micro-organism → Infection controlling → costly & limited

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

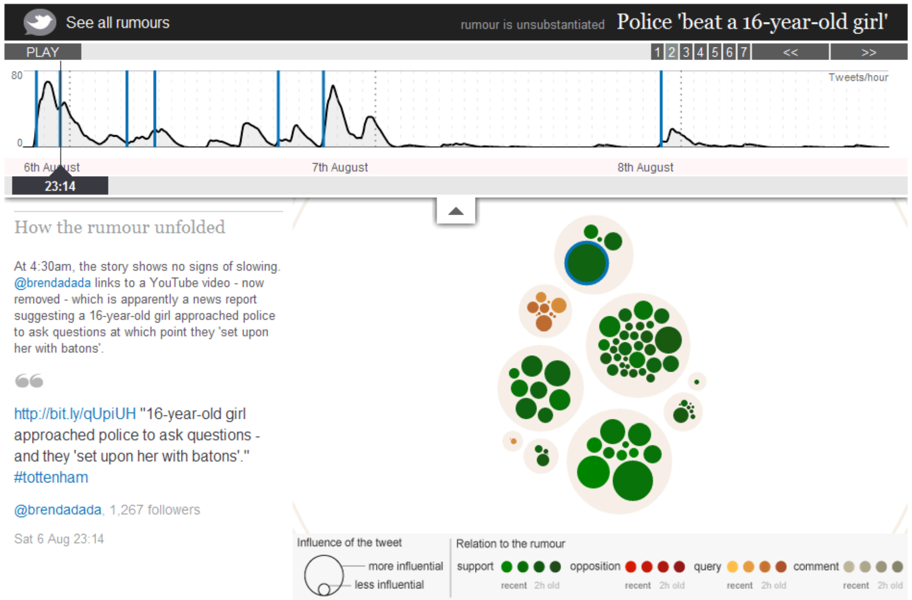
[SDM'13b]



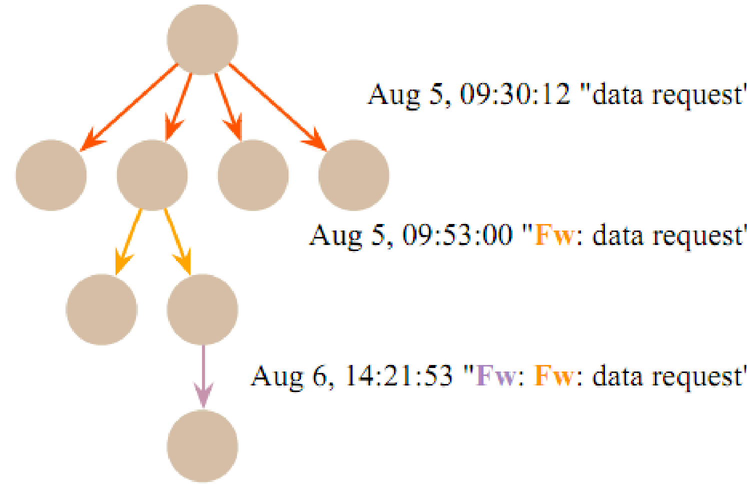
Red: Infected Hospitals after 365 days

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

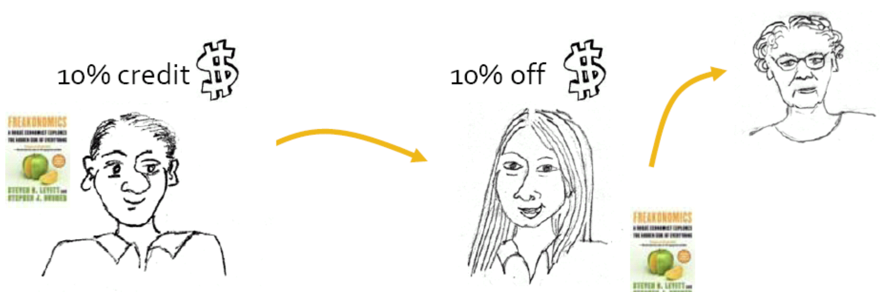
Why Do We Care? (More)



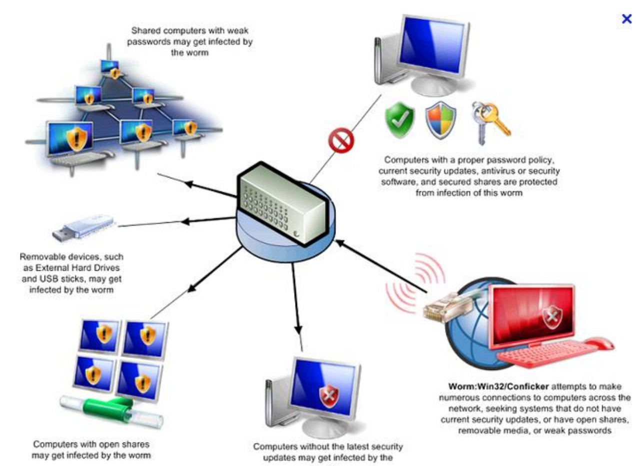
Rumor Propagation



Email Fwd in Organization

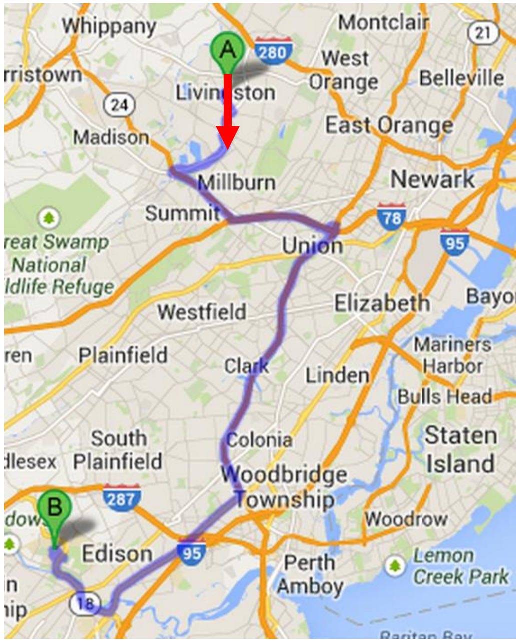


Viral Marketing



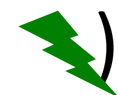
Malware Infection

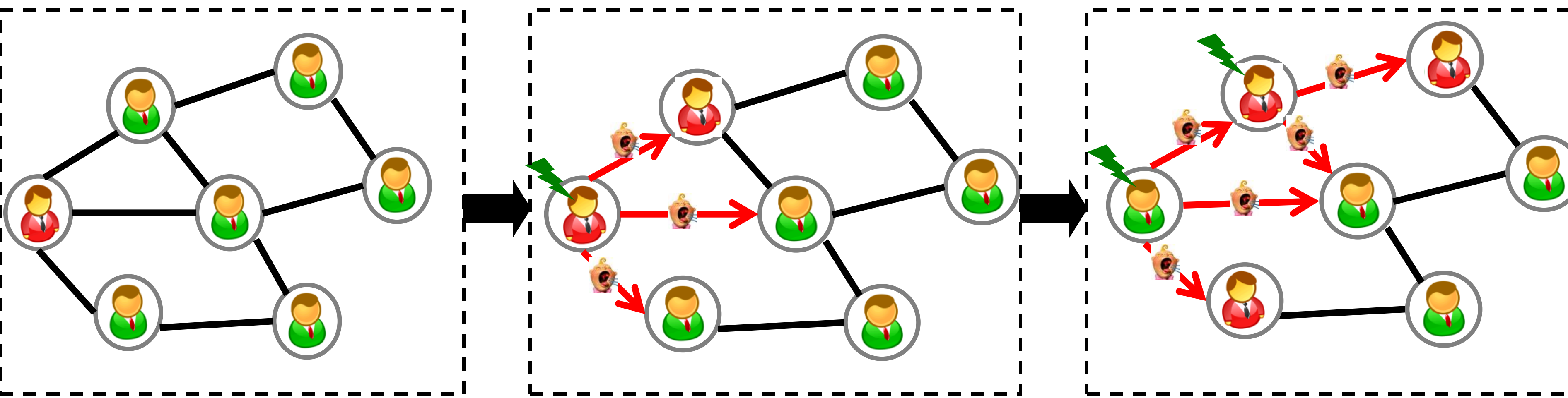
Roadmap



- ✔ Motivations
- ➡ Q1: Theory – Tipping Point
- Q2: Minimize the propagation
- Q3: Maximize the propagation
- Conclusions

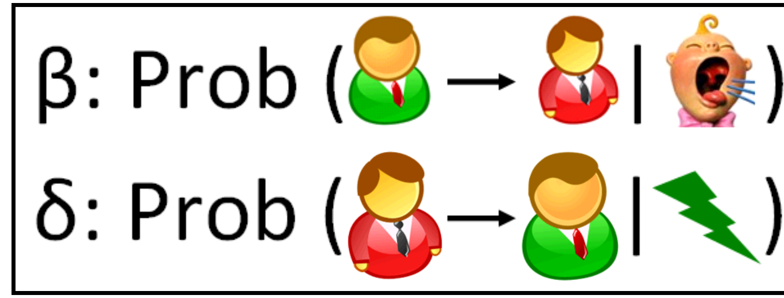
SIS Model (e.g., Flu)
(Susceptible-Infected-Susceptible)

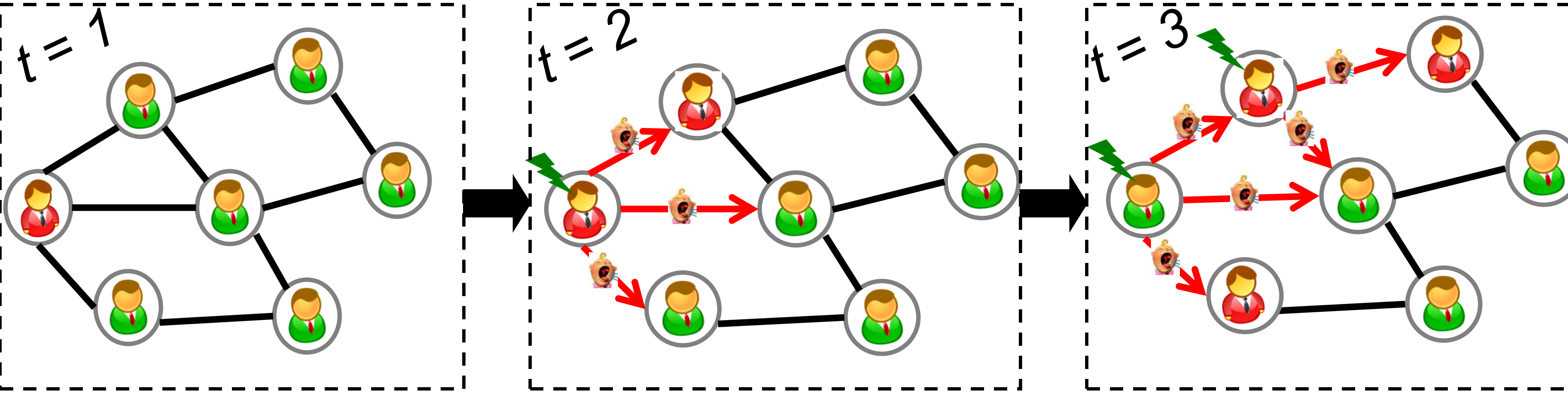
- Each Node Has Two Status:  Sick  Healthy
- β: Infection Rate (Prob ( →  | ))
- δ: Recovery Rate (Prob ( →  | ))



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

SIS Model as A NLDS





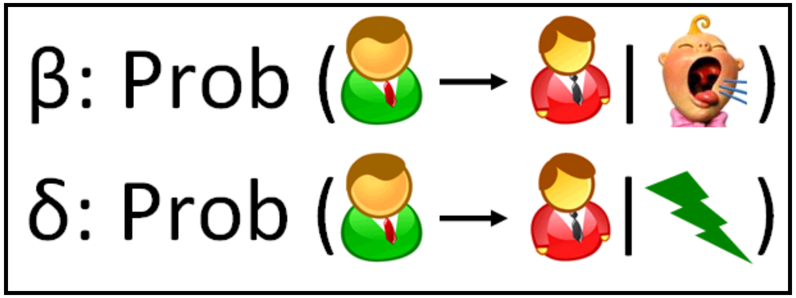
Prob. vector: nodes being sick at (*t+1*)

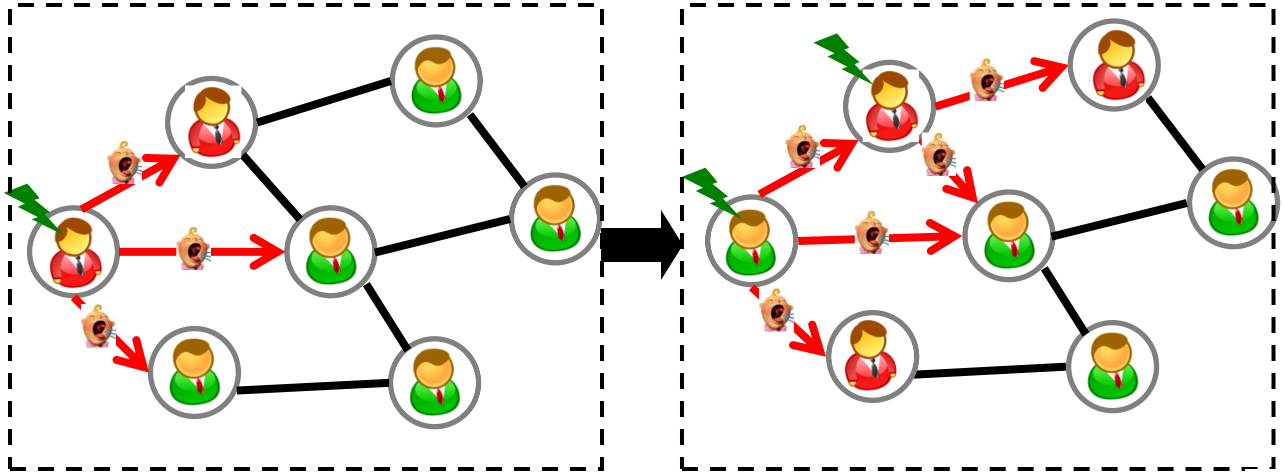
$$p_{t+1} = g\,(p_t)$$

Prob. vector: nodes being sick at *t*

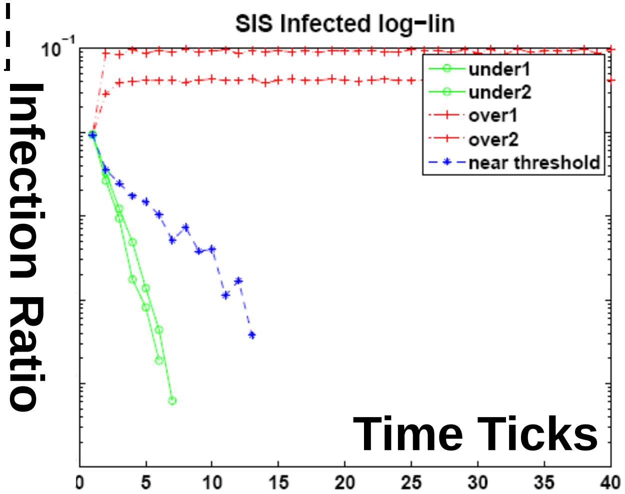
Non-linear function: depends on
(1) graph structures
(2) virus parameters (β, δ)

SIS Model (e.g., Flu)





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



Theorem [Chakrabarti+ 2003, 2007]:

If ***λ x (β/δ) ≤ 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

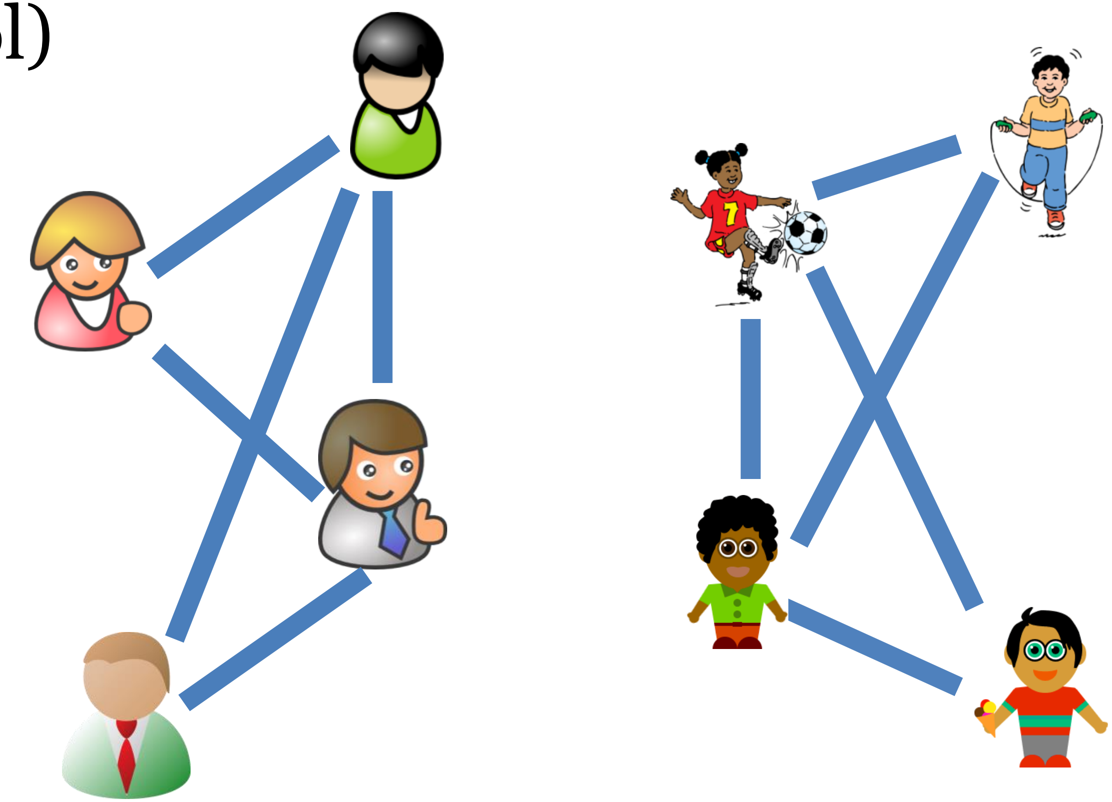
[PKDD 2010, Networking 2011]

DAY
(e.g., work, school)

A_1: adjacency matrix

8

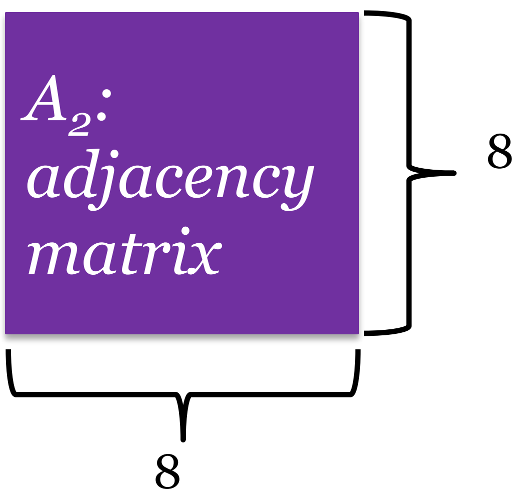
8

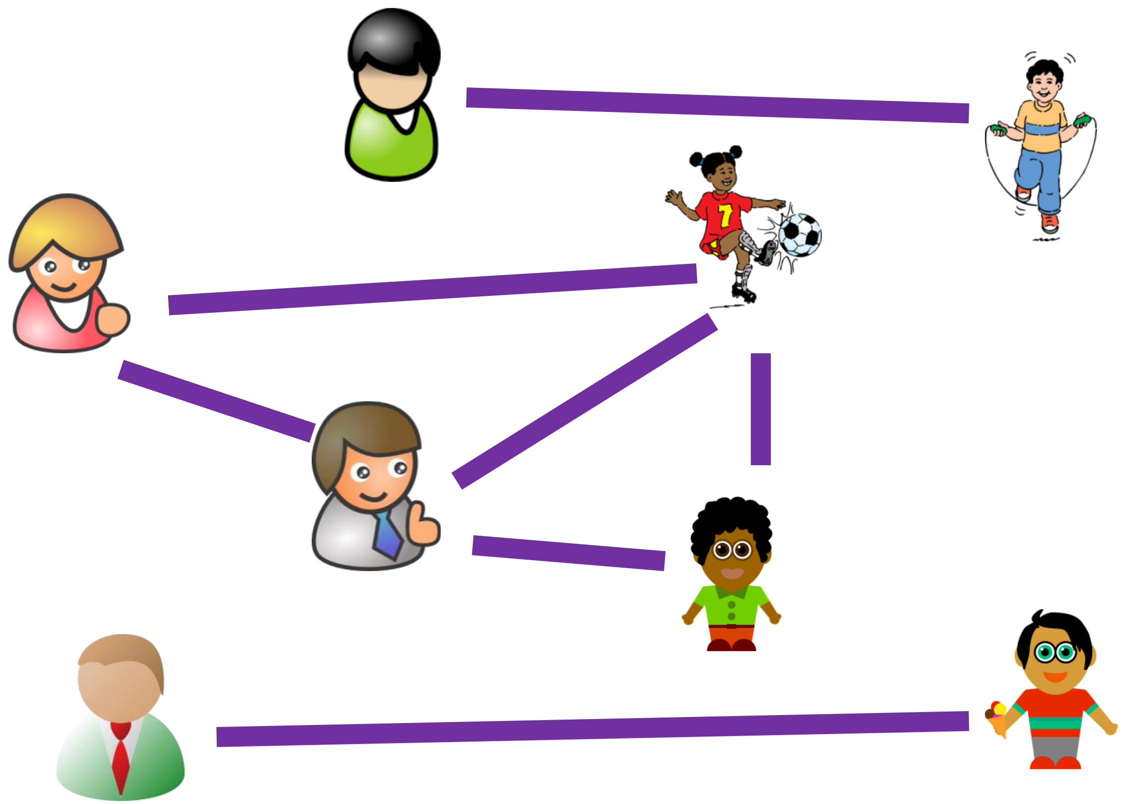


Beyond Static Graphs: Alternating Behavior

[PKDD 2010, Networking 2011]

NIGHT
(e.g., home)

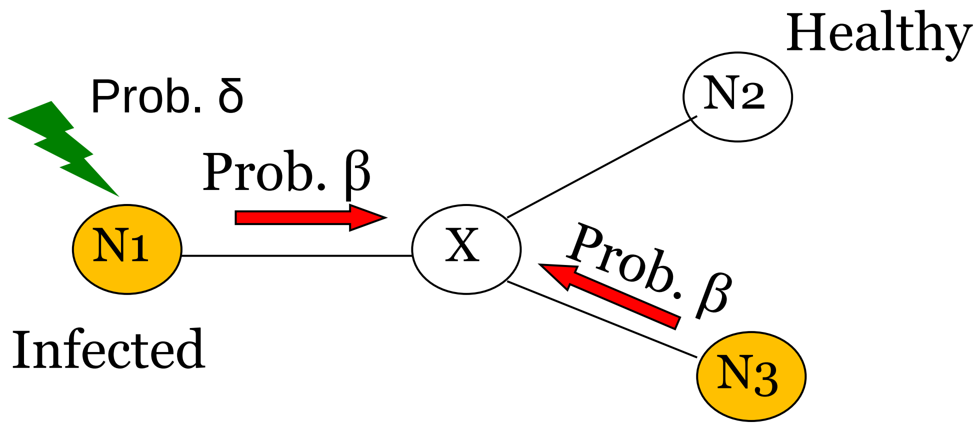




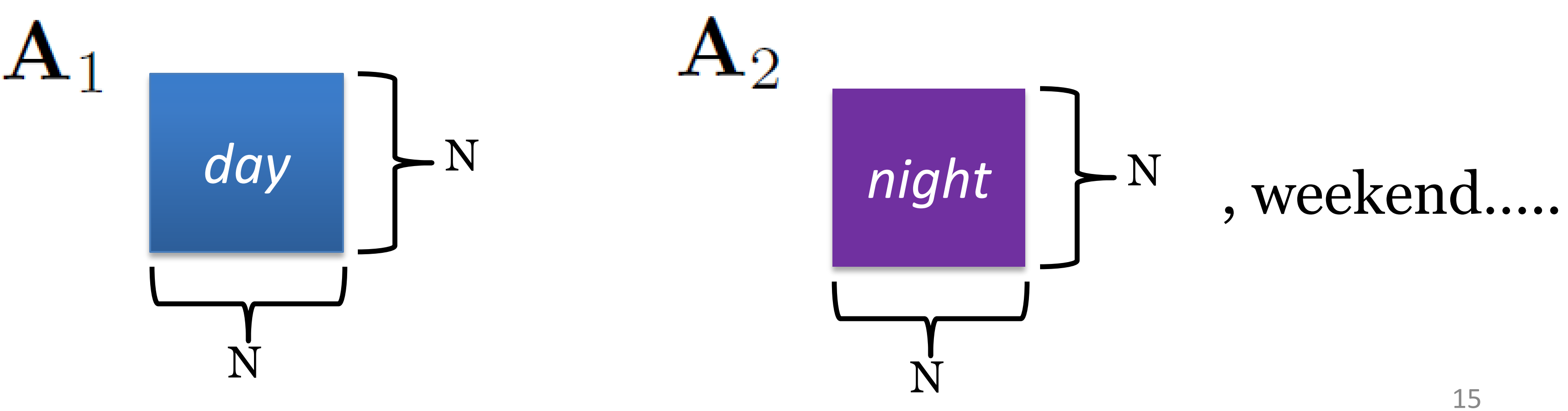
Formal Model Description

[PKDD 2010, Networking 2011]

- SIS model
  - recovery rate δ
  - infection rate β



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



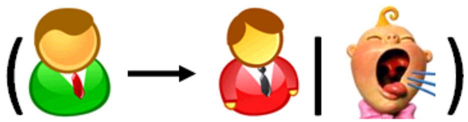
Epidemic Threshold for Alternating Behavior

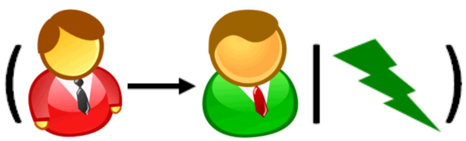
[PKDD 2010, Networking 2011]

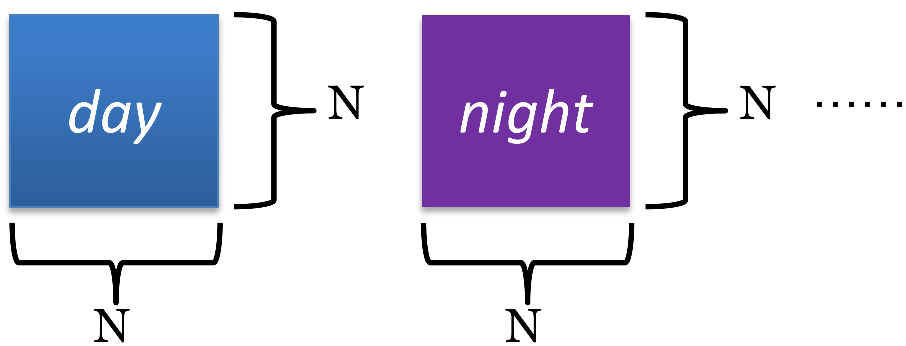
Theorem [PKDD 2010, Networking 2011]:
No epidemic If $\lambda(S) \leq 1$.

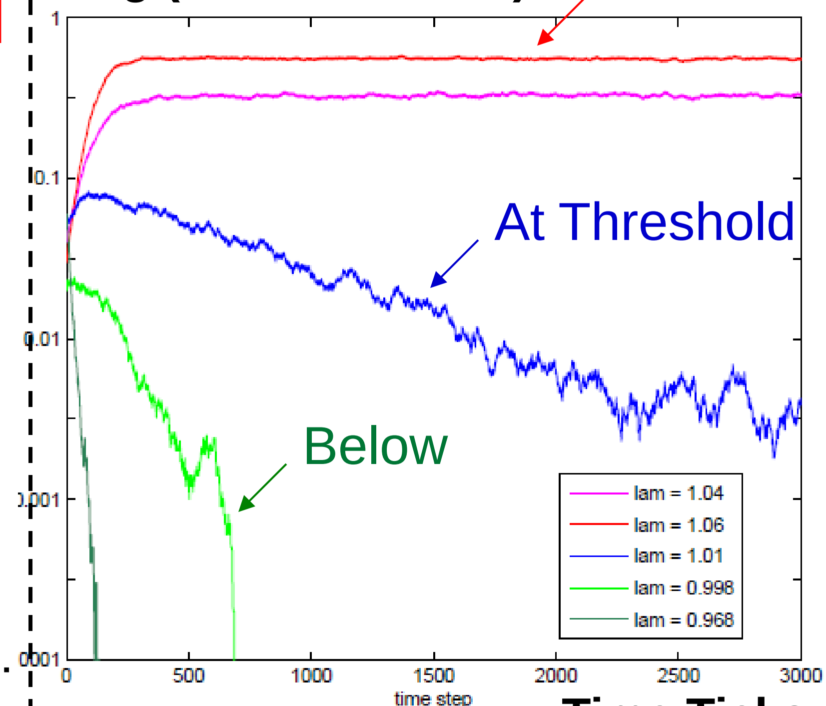
System matrix $S = \Pi_i S_i$

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

β: Prob 

δ: Prob 

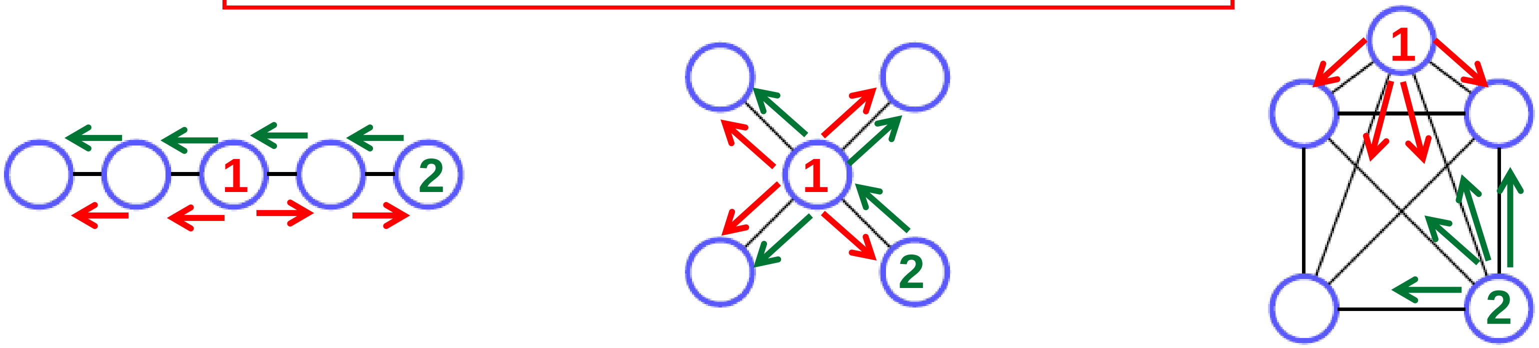
A_i 



Why is λ So Important?

- **λ** → Capacity of a Graph:

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



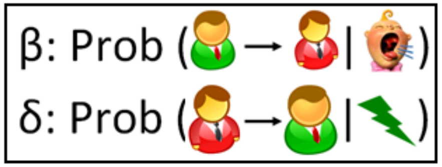
(a)Chain($\lambda_1 = 1.73$) (b)Star($\lambda_1 = 2$) (c)Clique($\lambda_1 = 4$)

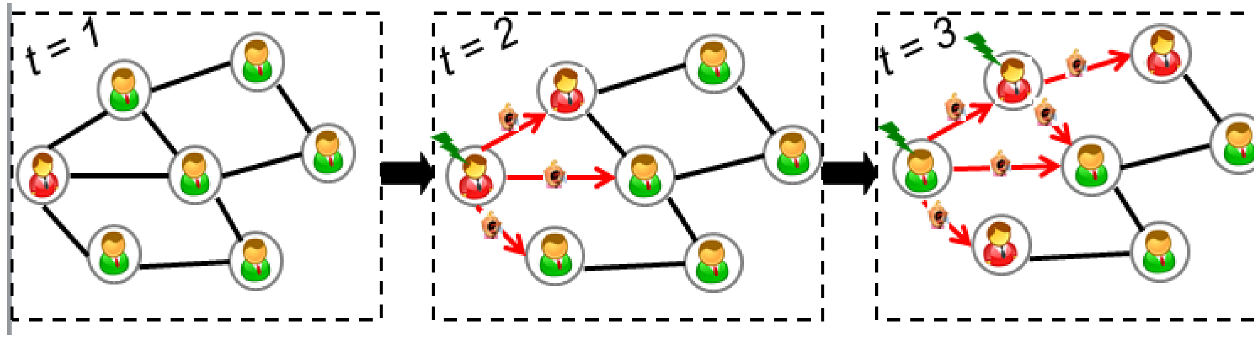
Larger *λ* → better connected

Why is λ So Important?

Details

- **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 [PKDD 2010]:**

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

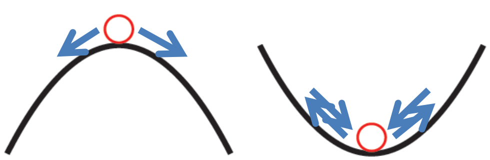
$$J_{k,l} = [\nabla g(\mathbf{p}^*)]_{k,l} = \frac{\partial p_{k,t+1}}{\partial p_{l,t}}\Big|_{\mathbf{p}_t=\mathbf{p}^*}$$

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

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

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

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



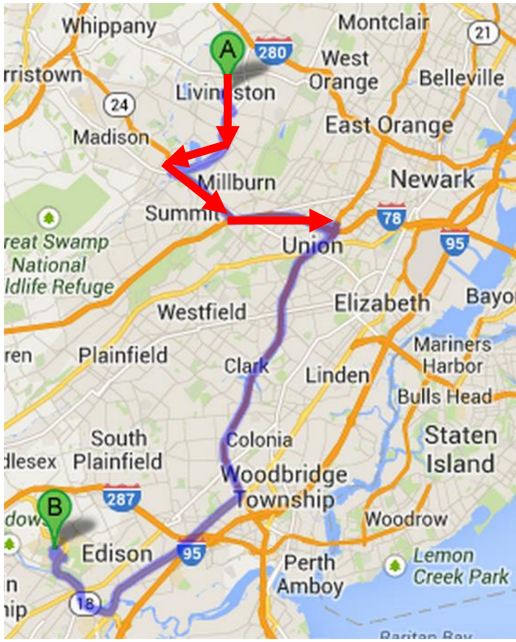
(A) Unstable

(B) Stable

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

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

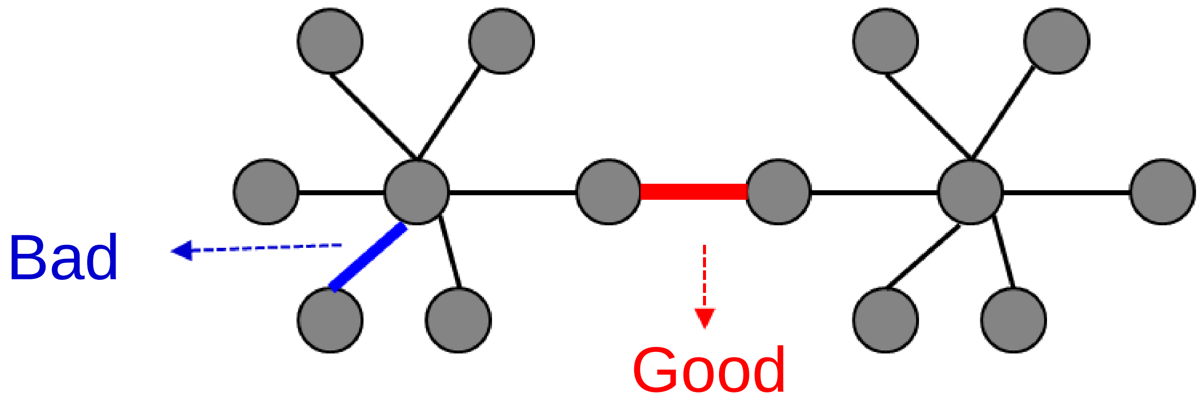
Roadmap



- ✔ Motivations
- ✔ Q1: Theory – Tipping Point
- ➡ Q2: Minimize the propagation
- Q3: Maximize the propagation
- Conclusions

Minimizing Propagation: Edge Deletion

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



Challenge: We need $O(\binom{m}{k}m)$ time for Naïve method!

Q: How to find k best edges to delete **efficiently**?

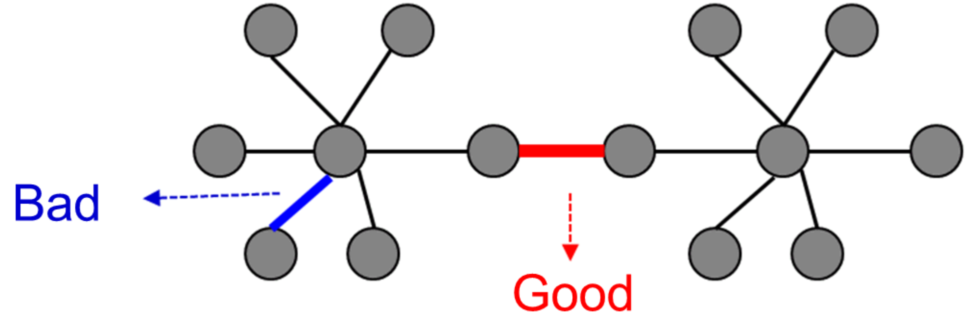
[CIKM12 a]

- Our Sol: By 1st order perturbation, we have

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

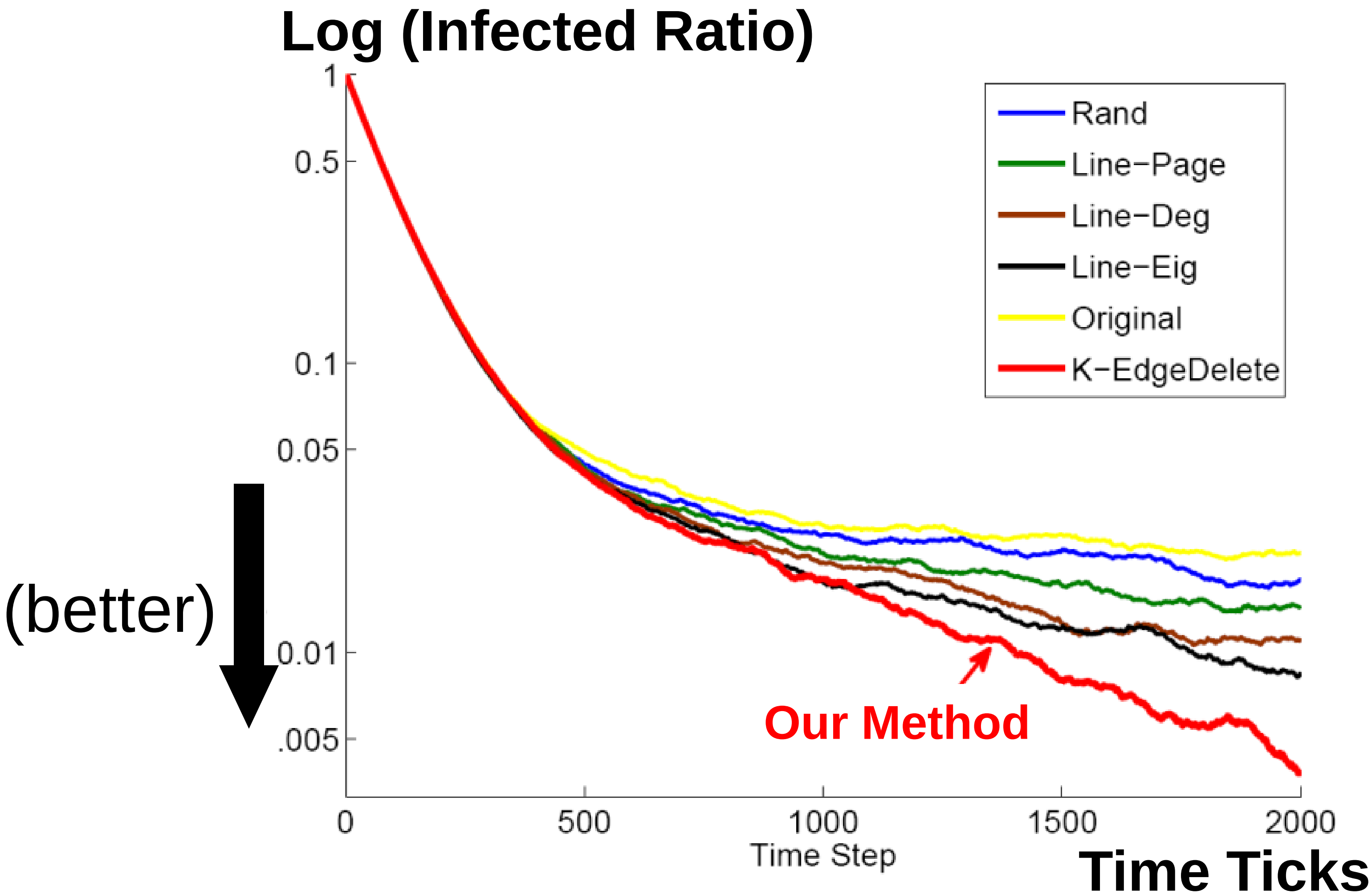
Left eigen-score of source

Right eigen-score of target



- Observations:
  - Only need eigen-computation once
  - Impact of different edges are de-coupled

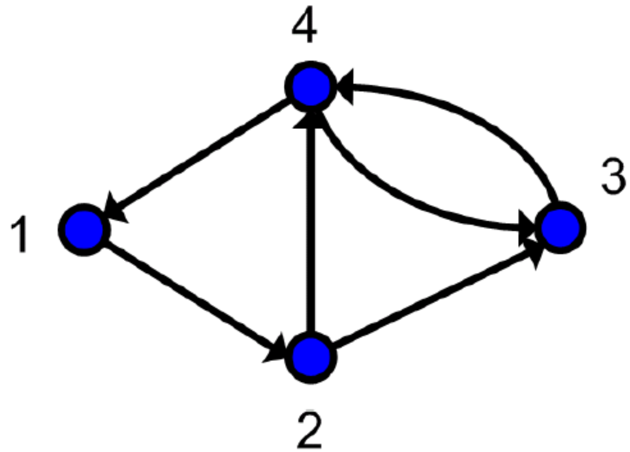
Minimizing Propagation: Evaluations [CIKM12 a]



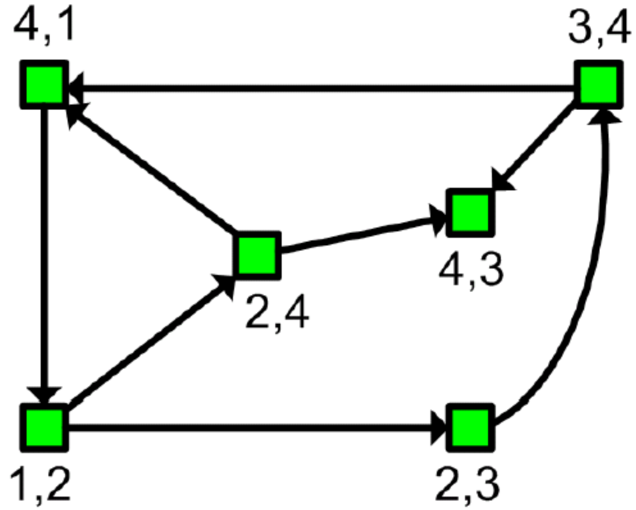
Data set: Oregon Autonomous System Graph (14K node, 61K edges)

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)*

- But, Node Deletion itself is not easy:

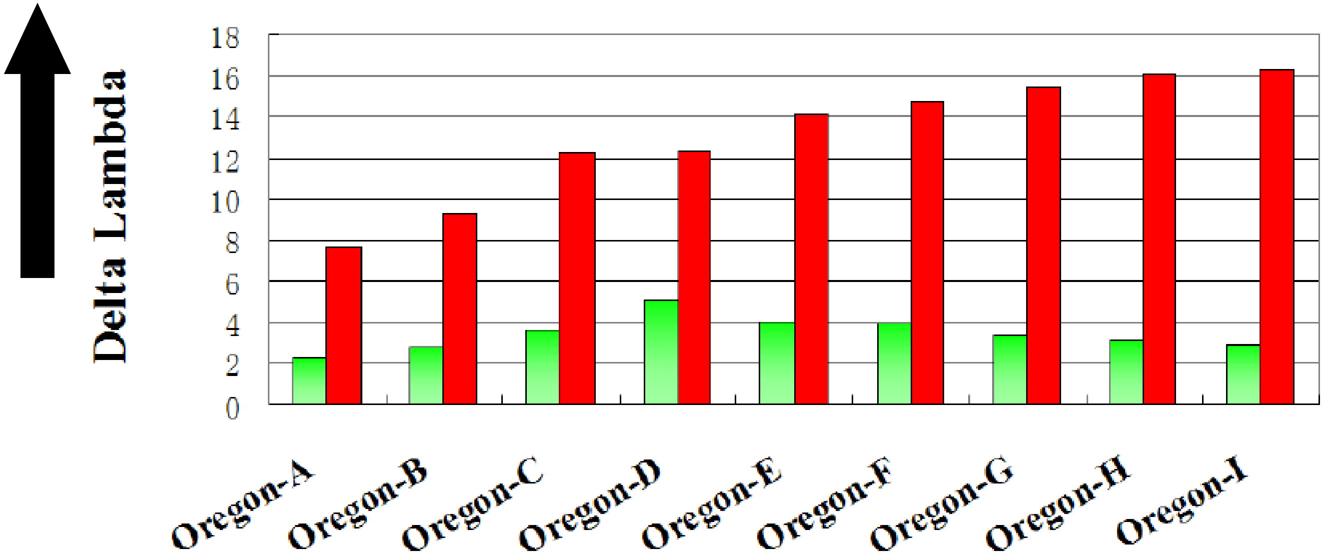
Theorem: Hardness of Node Deletion.

Find Optimal k-node Immunization is NP-Hard

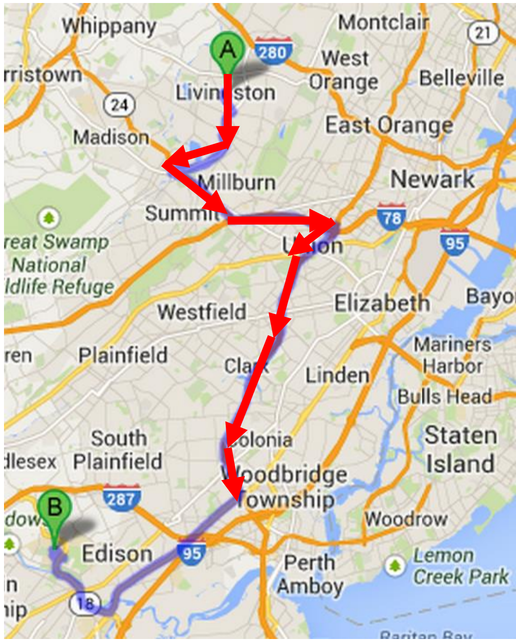
Discussions: Node Deletion vs. Edge Deletion

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

(better)



Green: Node Deletion [ICDM 2010](e.g., shutdown a twitter account)
Red: Edge Deletion (e.g., un-friend two users)



Roadmap

- ✔ Motivations
- ✔ Q1: Theory – Tipping Point
- ✔ Q2: Minimize the propagation
- ➡ Q3: Maximize the propagation
- Conclusions

Maximizing Dissemination: Edge Addition

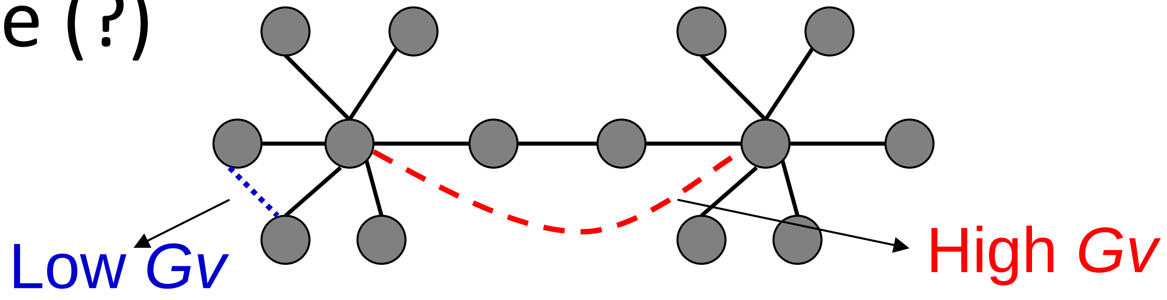
- **Given**: a graph *A*, virus prop model and budget *k*;
- **Find**: add *k* 'best' new edges into *A*.
  - By 1st order perturbation, we have

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

Left eigen-score of source

Right eigen-score of target

  - So, we are done (?)

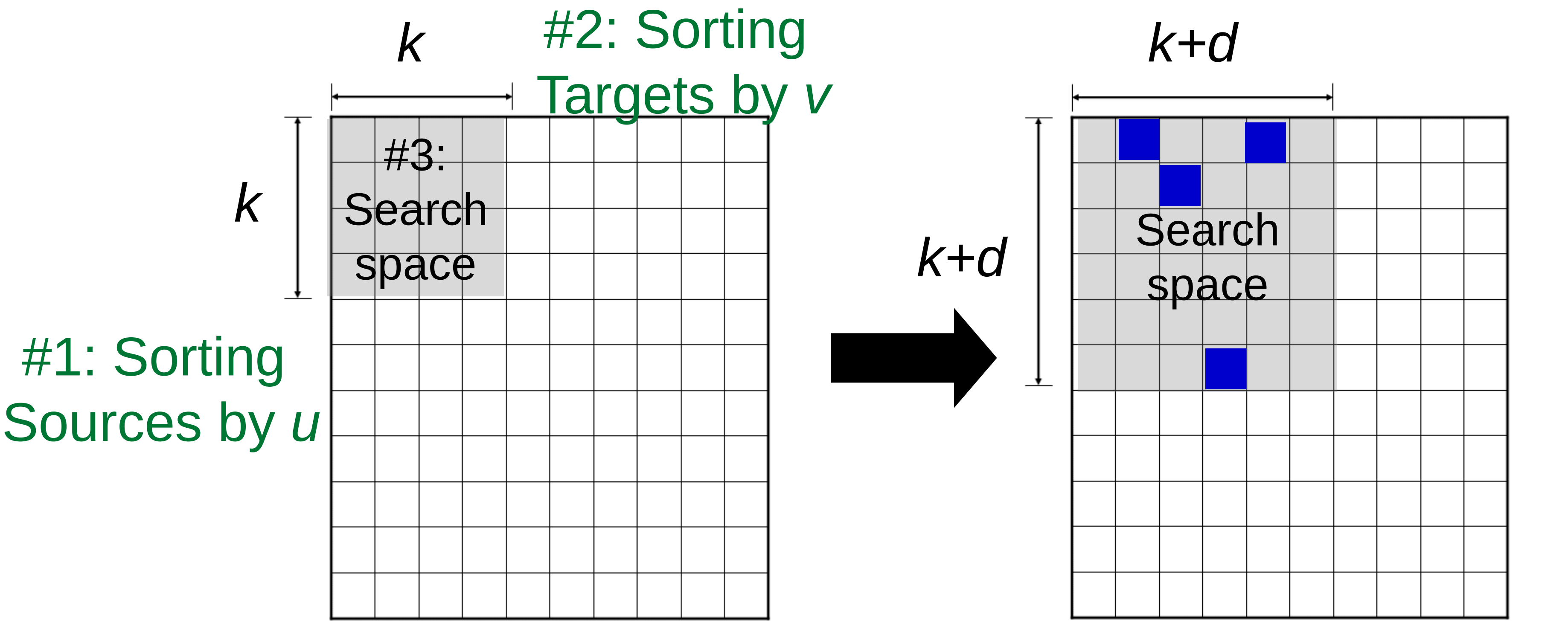


But … it has O(n^2-m) complexity

Maximizing Dissemination: Edge Addition

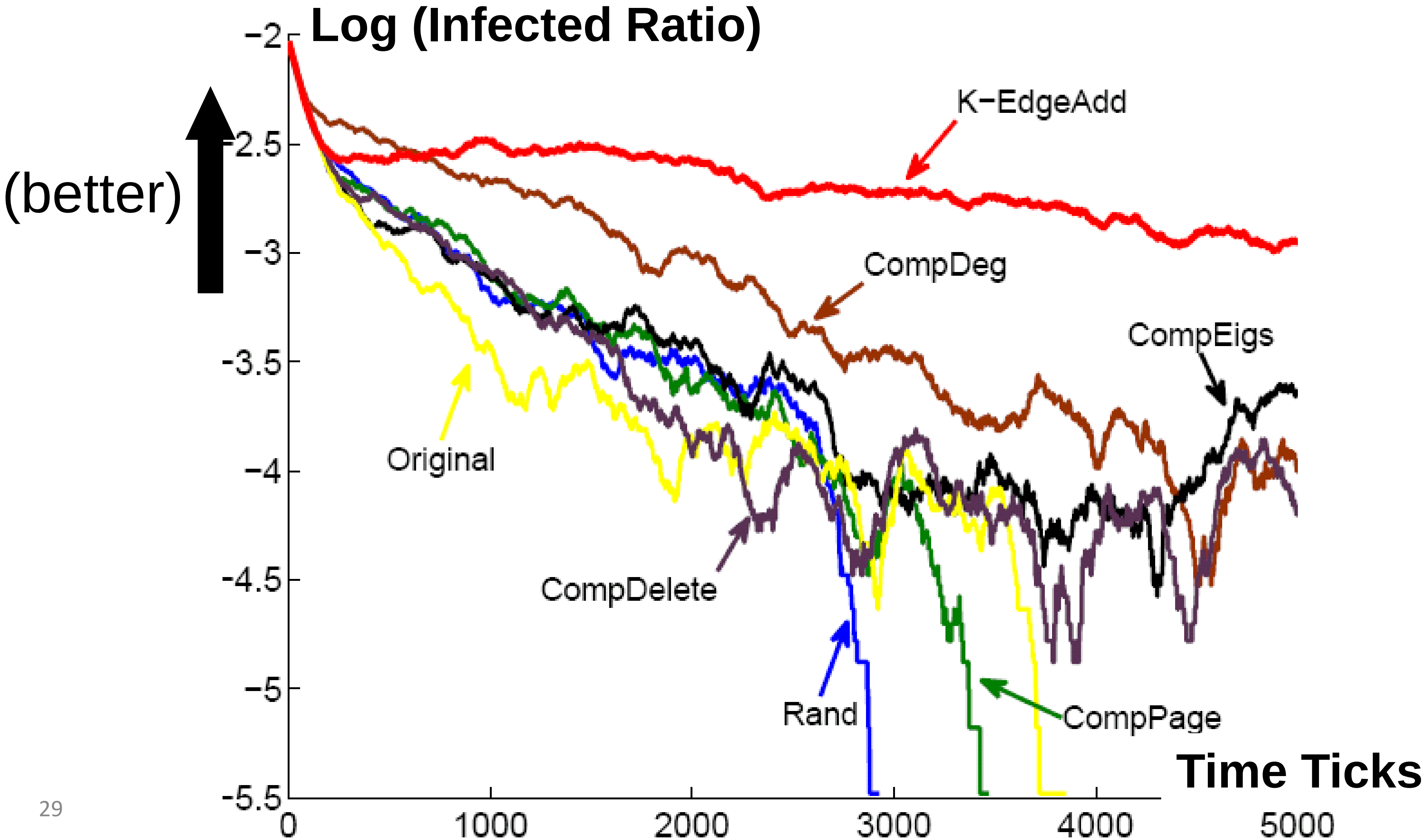
$$\lambda_s - \lambda \approx Gv(S) = c \sum_{e \epsilon S} u(i_e) v(j_e)$$

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

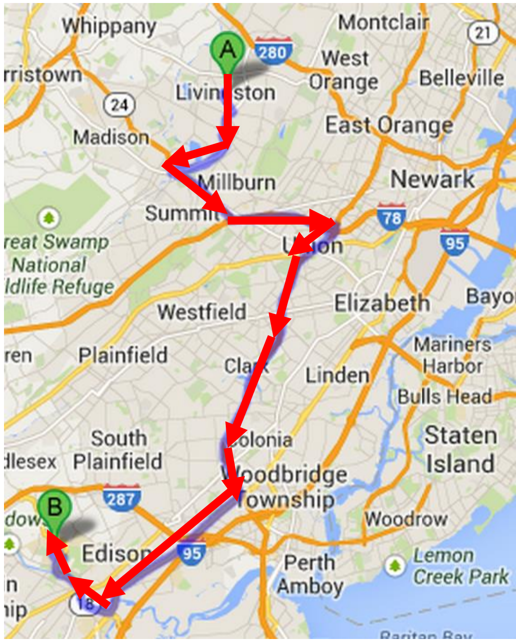


Time Complexity: $O(m+nt+kt^2)$, $t = \max(k,d)$ ■ :existing edge

Maximizing Dissemination: Evaluation



Conclusions



- **Goal**: Guild Dissemination by Opt. G
- **Theory**: Opt. Dissemination = Opt. λ
- **Algorithms**:
  - NetMel to Minimize Dissemination
  - NetGel to Maximize Dissemination
- **More on This Topic**
  - Beyond Link Structure (content, attribute) [WWW11]
  - Beyond Full Immunity [SDM13b]
  - Node Deletion [ICDM2010]
  - Higher Order Variants [CIKM12a]
  - Immunization on Dynamic Graphs [PKDD10]

Acknowledgement: Lada A. Adamic, Albert-László Barabási, Tina Eliassi-Rad, Christos Faloutsos, Michalis Faloutsos, Theodore J. Iwashyna, B. Aditya Prakash, Chaoming Song, Spiros Papadimitriou, Dashun Wang.