IOWA STATE UNIVERSITY
Department of Computer Science

How to spread Information in Social Networks that contain Adversarial Users

Madhavan Padmanabhan

In Collaboration with Naresh Somisetty, Samik Basu, Pavan Aduri

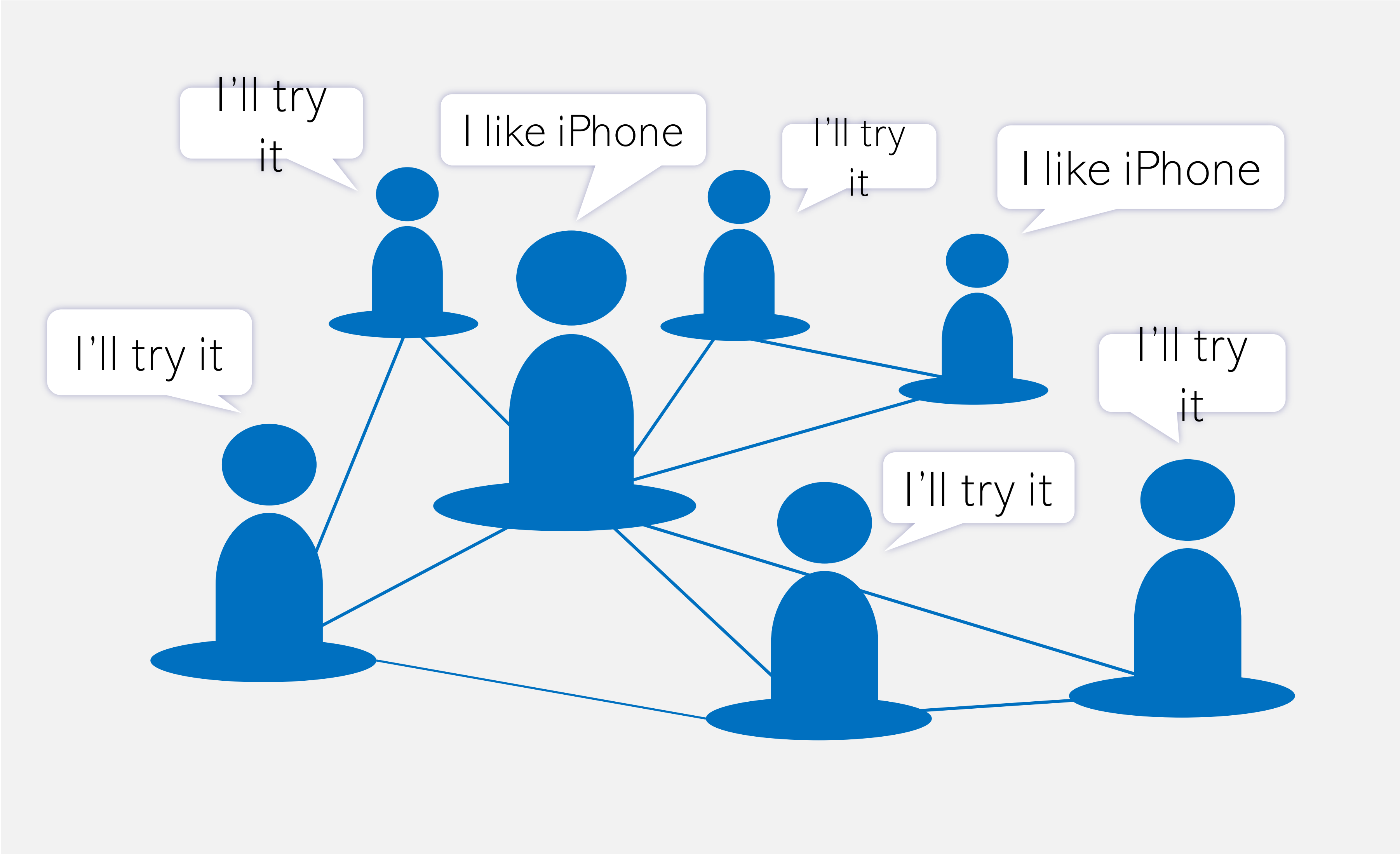
Supported in part by NSF grants CCF 1421163 and CCF 1555780

Outline

- Overview of Influence Maximization
- CIM Problem
- Greedy Algorithm
- MultiGreedy Algorithm
- Experimental Results

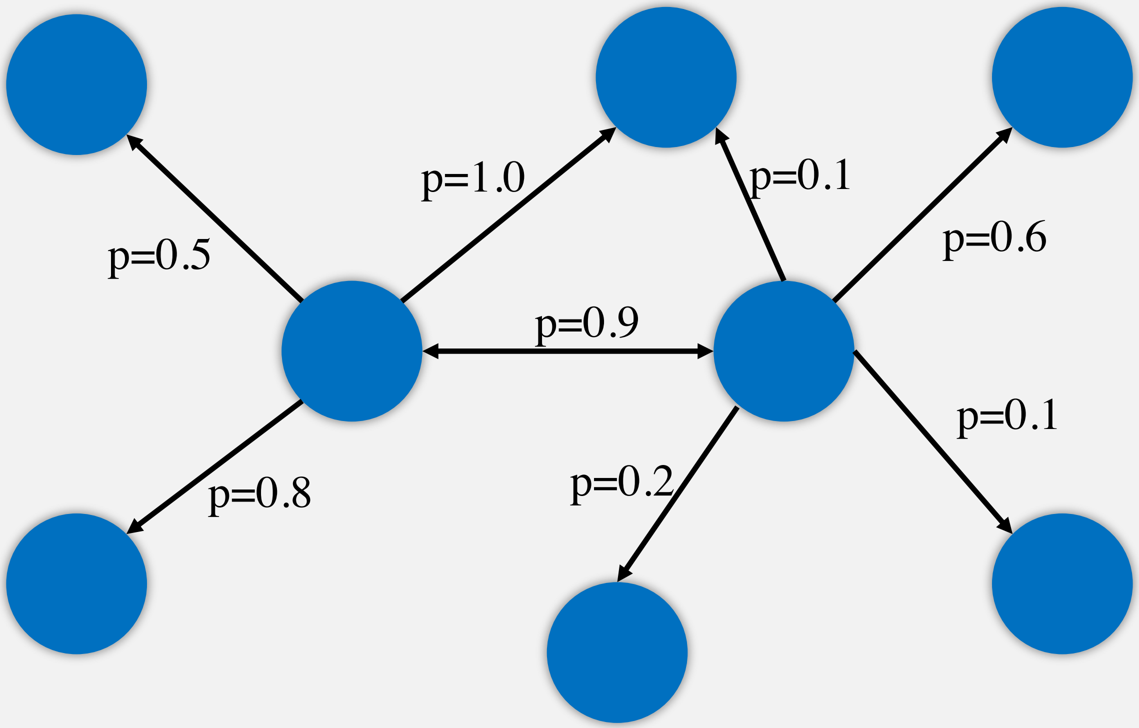
OVERVIEW OF INFLUENCE MAXIMIZATION

Social Networks



Influence Maximization Problem

- Find a set of highly influential users *(seed)* in the network
- Posed as an optimization problem by Kempe et. Al.



Applications

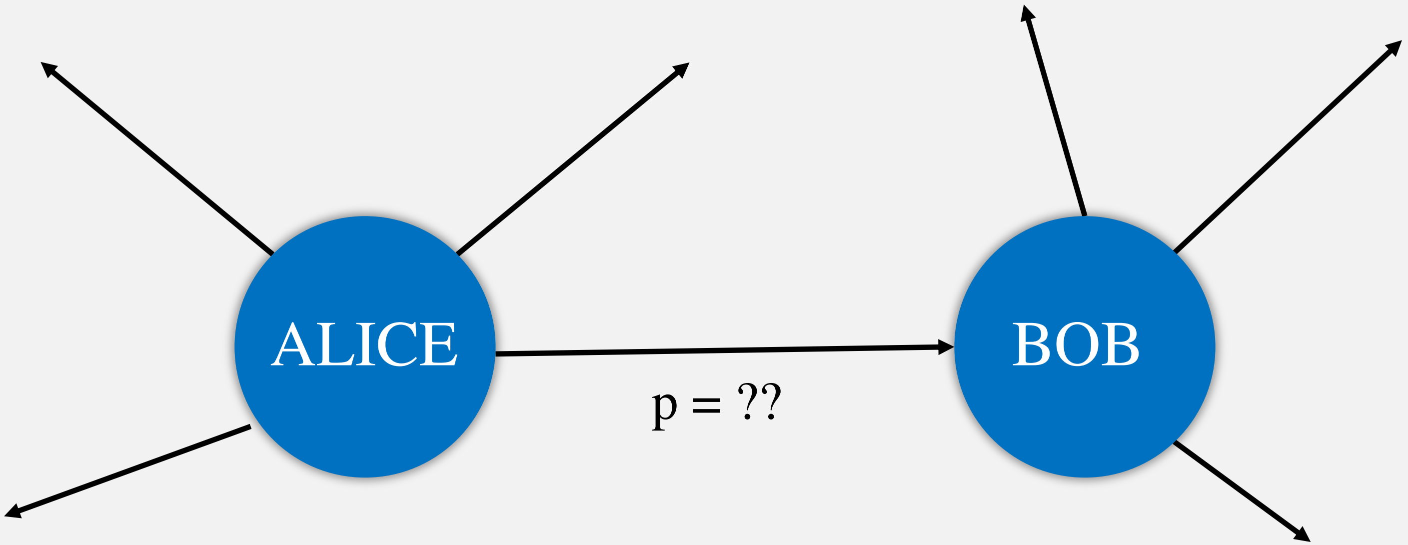
- Political Campaigning – How can I get people to vote for me?

- Viral Marketing – Who do I ask to advertise a product?

Information Diffusion

1. An “idea” originates from a user or a set of users in the network. These users are called the ***“Seed”*** users.
2. Users connected to the ***Seed*** are exposed to the idea.
3. The exposed users, if they choose to, further propagate the idea to users connected to them.

Diffusion – A Probabilistic Process



- Alice posts about something !
- What’s the chance that Alice influences Bob?
- What’s the chance Alice influences Bob’s friends?

Diffusion Models

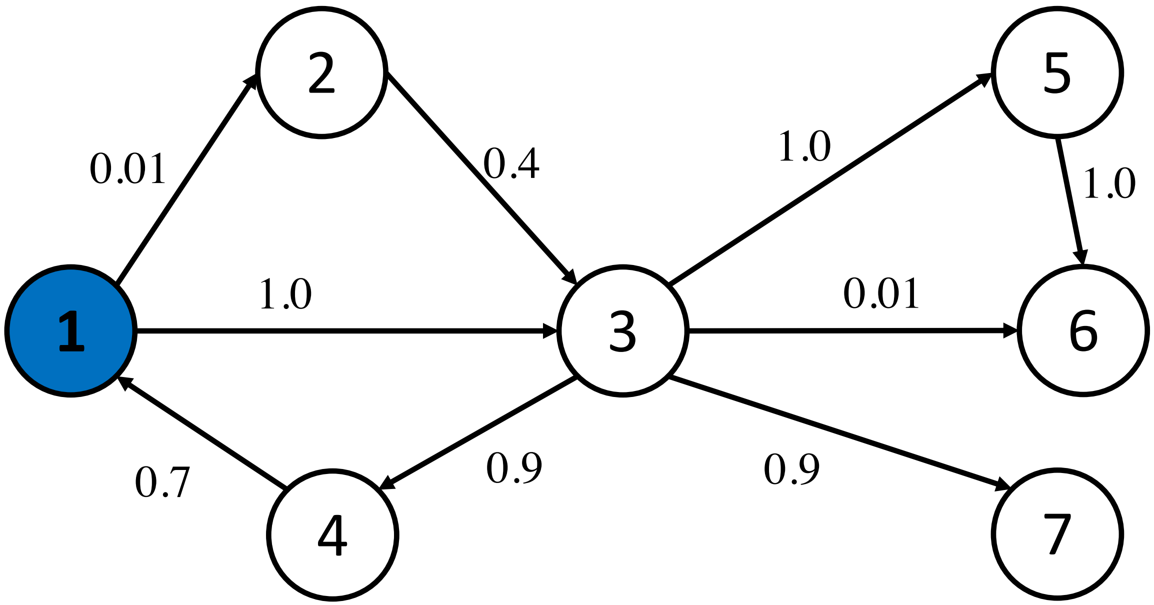
- Models of *Diffusion* – Characterizes the spread of information from one user to the next
- Models mirror the diffusion in real world social networks.
- 2 popular models:
  - Independent Cascade (IC) Model
  - Linear Threshold (LT) Model

Independent Cascade Model

- The network is modelled as a graph $G = (V, E)$.
- Every edge (u, v) has an associated probability - $p(uv)$.
- u has exactly one chance to convince v to adopt the idea.

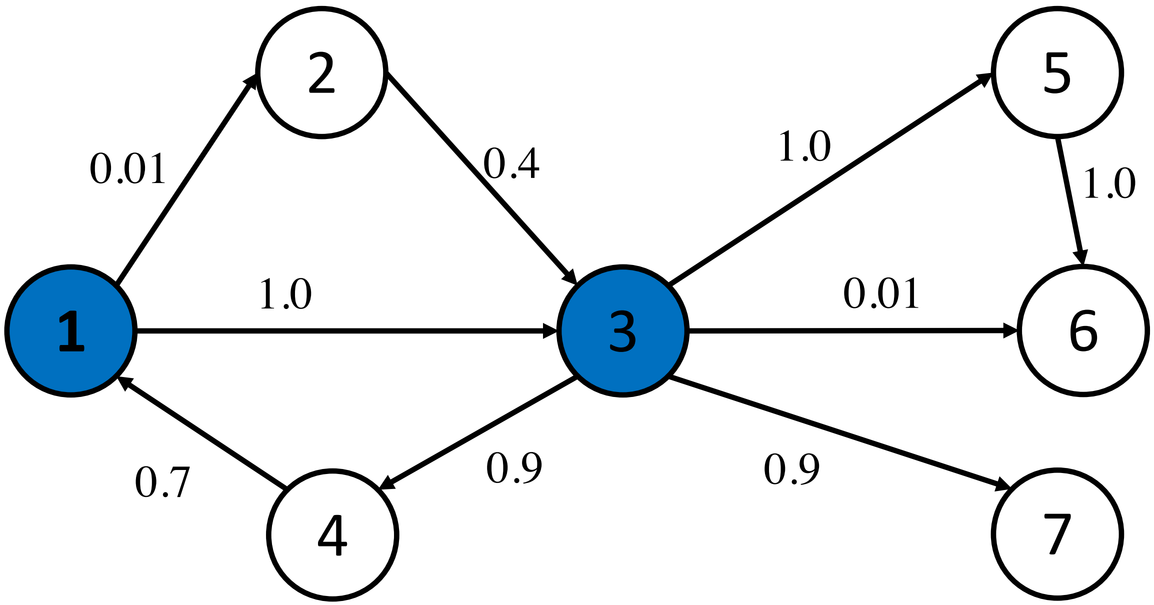
  u succeeds with probability $p(uv)$.
- If successful, v is considered to be ***"activated"***.

The Independent Cascade Model – an example



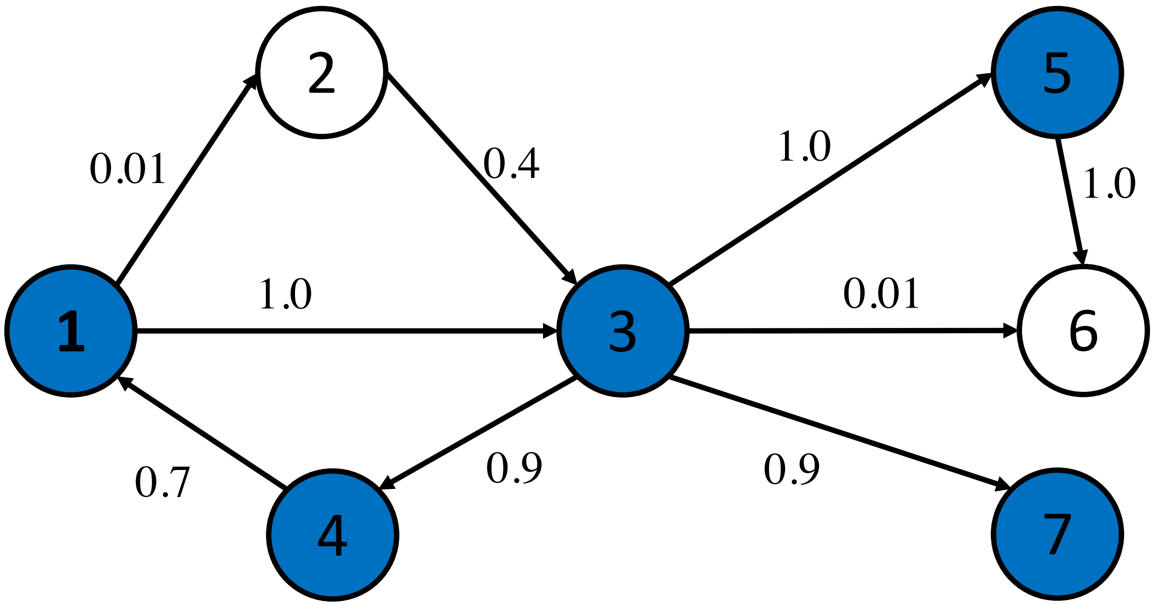
$t = 0$

The Independent Cascade Model – an example



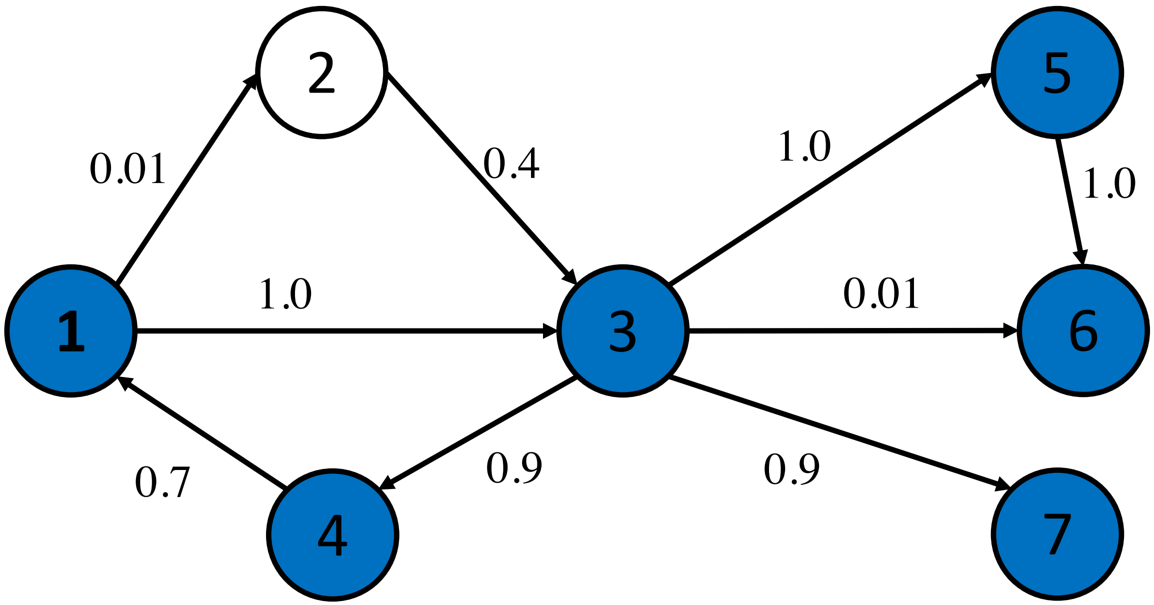
t = 1

The Independent Cascade Model – an example



t = 2

The Independent Cascade Model – an example

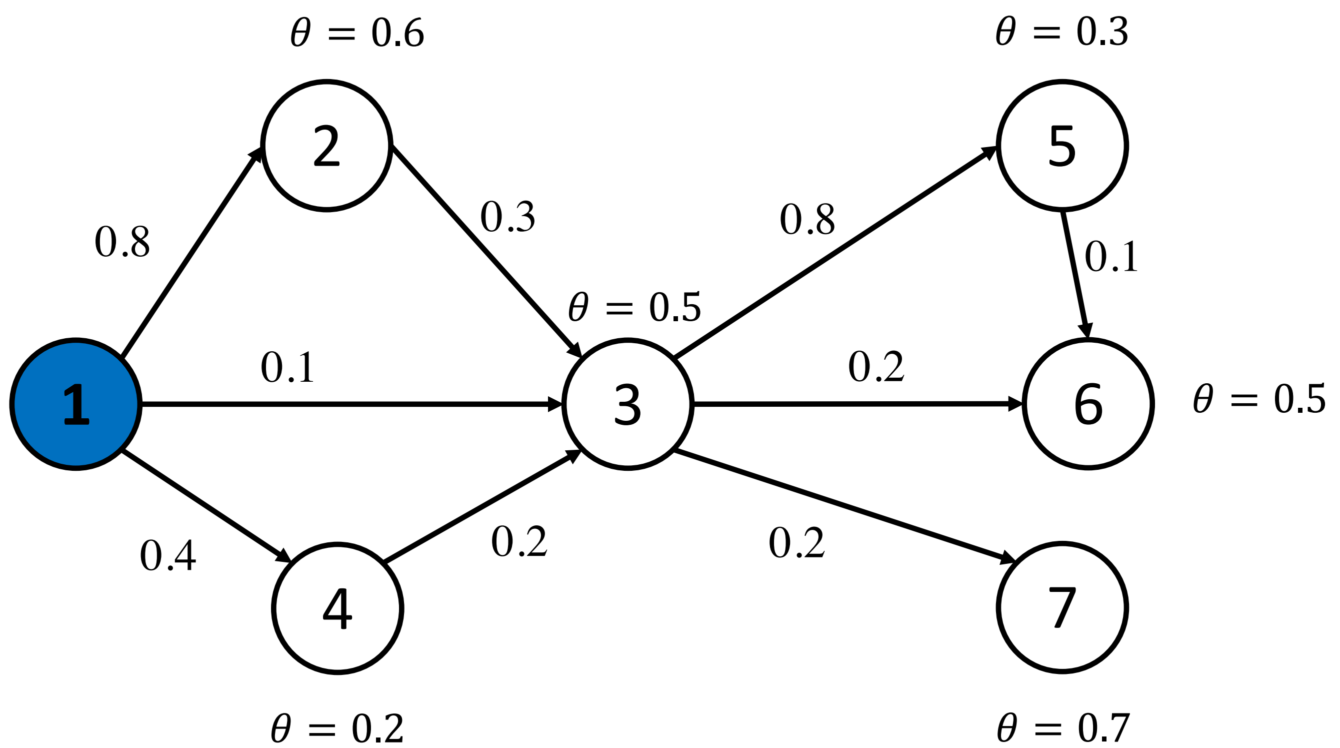


t = 3

Linear Threshold Model

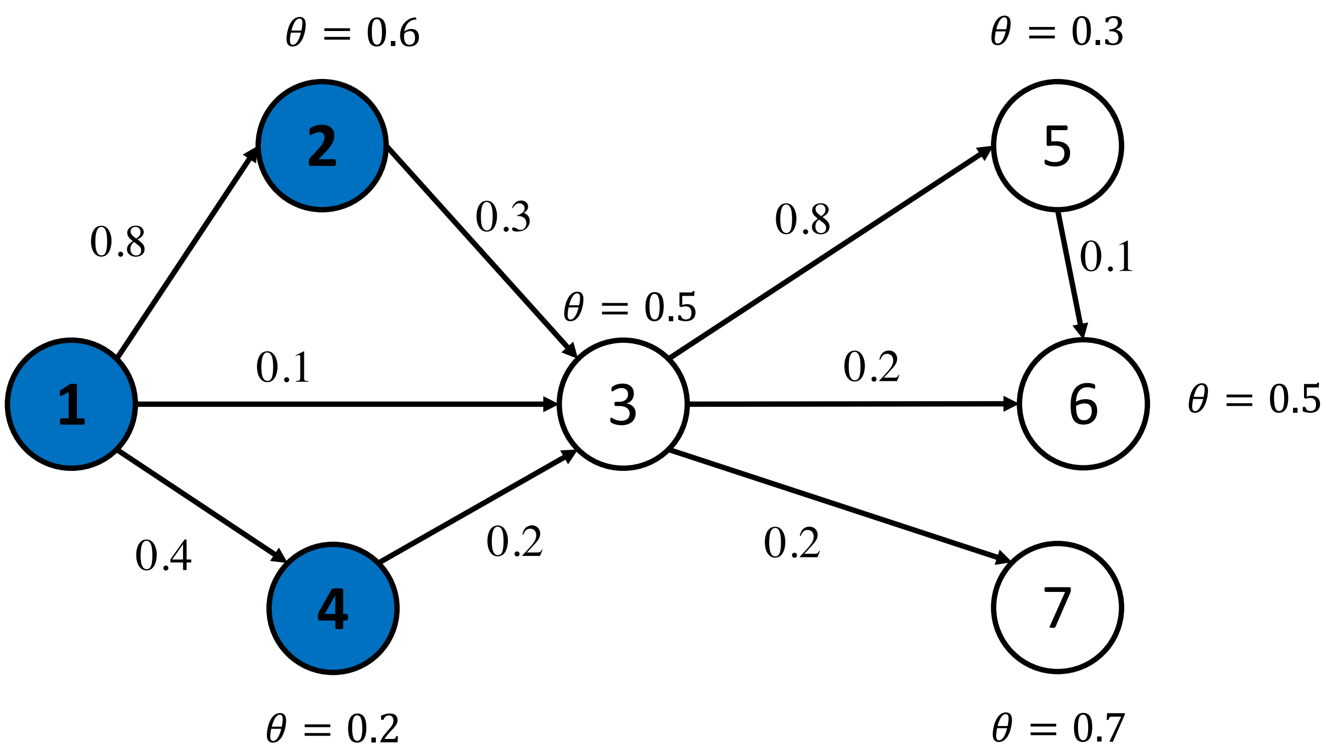
- The network is modelled as a graph $G = (V, E)$.
- Every edge (u, v) has an associated weight - w(u, v).
- Each node v is assigned a random *threshold* $\theta_v \in [0,1]$.
- v is considered to be ***"activated"*** if sufficient neighbors of v are activated:
  - $\boldsymbol{\sum_{Active\ u} w(u, v) \geq \theta_v}$

The Linear Threshold Model – an example



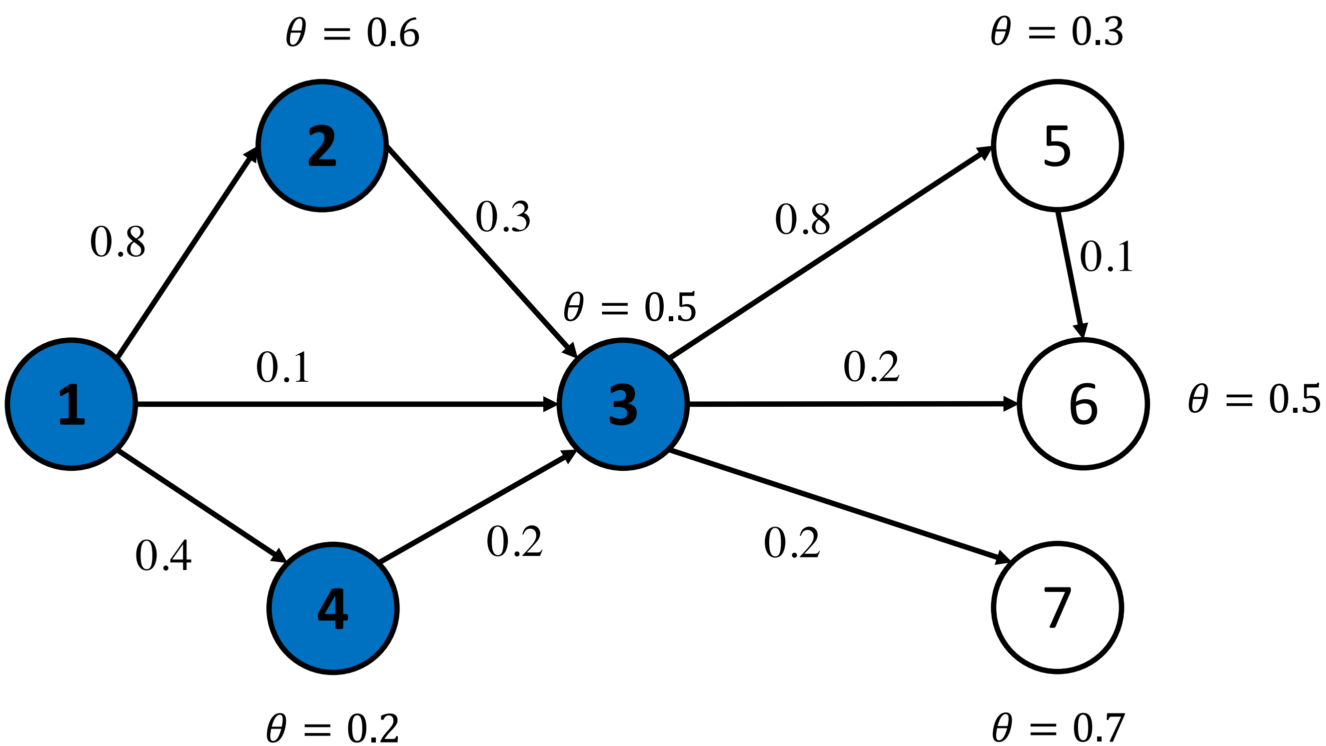
t = 0

The Linear Threshold Model – an example



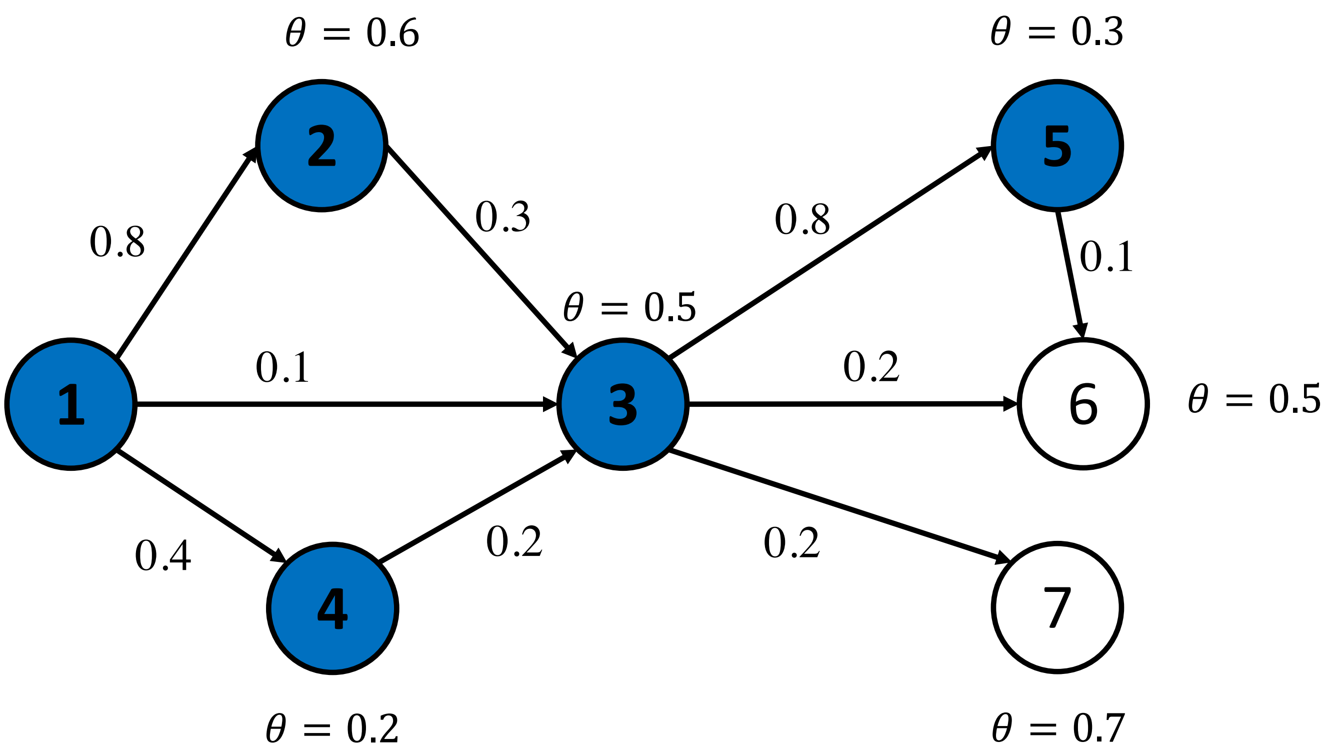
t = 1

The Linear Threshold Model – an example



t = 2

The Linear Threshold Model – an example



t = 3

Influence Function

- $\sigma(S)$ - *Expected number of users influenced by a set of users S*
- *Under the IC Model,* $\sigma(S)$ is *monotone, submodular*
- *Submodular:*

$$\forall X \subset Y \subseteq V \ and \ a \notin Y:$$

$$\sigma(X \cup \{a\}) - \sigma(X) \geq \sigma(Y \cup \{a\}) - \sigma(Y)$$

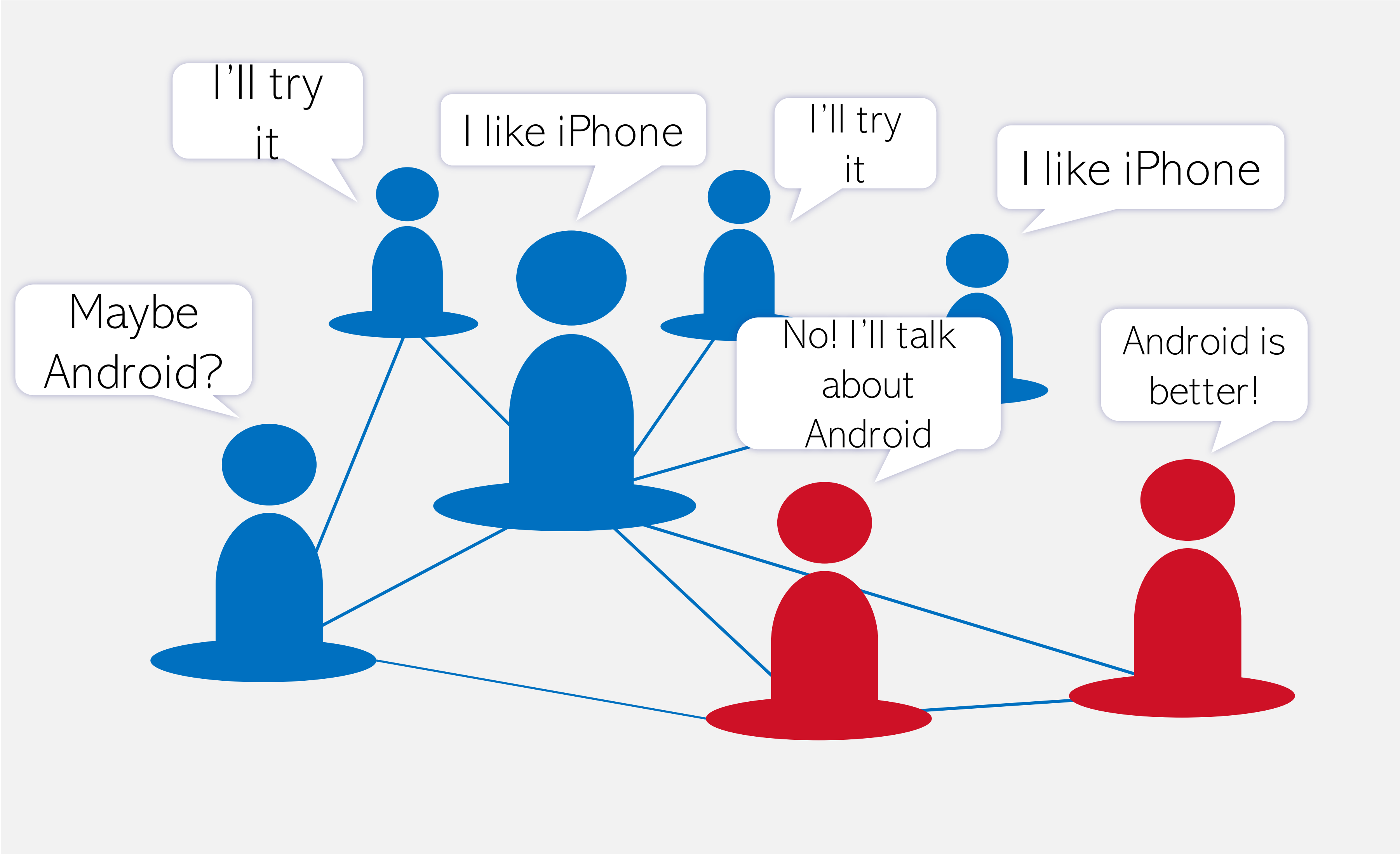
Influence Maximization

- **Input**: $G=(V,E)$, a budget k
- $\sigma(S)$ - *Expected number of users influenced by a set of users S*
- ***Objective***:

$$Find\ S\ of\ size\ k\ that\ maximizes\ \sigma(S)$$

- An NP-Hard problem
- *Greedy algorithm gives a 0.63-approximate solution*

What if there are adversaries?



What if there are adversaries?

- Political Campaigning – Can rally opposing candidate supporters
- Marketing: Advertisements for products such as alcohol, tobacco must not be shown to children
- Can cause a negative reaction to the information being spread
- How to approach this problem?

CIM PROBLEM

Constrained Influence Maximization(CIM)

- Label users as “Targets” or ”Non-Targets”
- Given: *G=(V,E, L),* budget *k,* threshold θ
- $\sigma_T(S)$ *- Expected number of “Target” users influenced by S*
- $\sigma_N(S)$ *- Expected number of “Non-Target” users influenced by S*
- $\sigma^\theta(S) = \begin{cases} \sigma_T(S), \; \sigma_N(S) \leq \theta \\ 0, \; otherwise \end{cases}$
- Objective:

$$Find\ S\ of\ size\ k\ that\ maximizes\ \sigma^\theta(S)$$

IM vs. CIM

Influence Maximization Problem

$$Maximize\ \sigma(S)$$
$$s.t.\ |S| \leq k$$

- $\sigma(S)$ is a monotone, submodular function.
- Greedy Algorithm gives a 0.63-approximate solution.

IM vs. CIM

Constrained Influence Maximization Problem

$$Maximize\ \sigma_T(S)$$
$$s.t.\ \sigma_N(S) \leq \theta$$
$$|S| \leq k$$

- $\sigma_T(S), \sigma_N(S)$ are monotone, submodular functions.
- Maximize under a submodular constraint and a cardinality constraint !

IM vs. CIM

IM Problem	CIM Problem
• NP-Hard	• NP-Hard
• Submodular Maximization	• Submodular Maximization
• Cardinality Constraint	• Submodular Constraint
• Greedy algorithm gives a 0.63-approximate solution.	• Cardinality Constraint

Theoretical Challenges of CIM

- **Theorem**: For every 0$\leq c \leq 1$, if there is a polynomial time c-approximation algorithm for the CIM problem under the IC model, then every problem in NP can be solved in $O\left(n^{(\log n)^k}\right)$ time, for some k≥ 1.

Obtaining a constant factor approximation algorithm is quasi NP-hard!

NATURAL GREEDY ALGORITHM

Natural Greedy Algorithm

Start with an empty Set

↓

Find the best vertex v that when added, the set influences at most θ Non Targets

↓

Repeat until a set of size k is obtained

Theoretical Analysis of Greedy

Theorem:

$$\sigma_T^{\theta}(S) \geq 0.63\ OPT - Additive\ Loss$$

The Greedy solution has an approximation guarantee that depends on an additive error!

Runtime: $O(k \times |V| \times Time\ taken\ to\ compute\ \sigma)$

Theoretical Analysis of Greedy

Proof Idea:

Let S^* be the set that has the optimum value $OPT_{k,\theta}$.

$gain_{S_i}(\{e\},\ \theta)$– The gain achieved by adding $\{e\}$ to S_i such that at most θ Non-Targets are influenced.

$$OPT_{k,\theta} \leq \sigma^{2\theta}(S^* \cup S_i)$$

$$OPT_{k,\theta} \leq \sigma^{\theta}(S_i) + \sum_{e \in S^*} gain_{S_i}(\{e\}, 2\,\theta)$$

$$OPT_{k,\theta} \leq \sigma^{\theta}(S_i) + k \times \sigma^{2\theta}(S'_{i+1}) - k \times \sigma^{\theta}(S_i)$$

Theoretical Analysis of Greedy

$BG(S_i, \theta)$ – The maximum gain achieved by adding an element to S_i such that at most θ Non-Targets are influenced

$$OPT_{k,\theta} - \sigma^{\theta}(S_{i+1}) \leq \left(1 - \frac{1}{k}\right)(OPT_{k,\theta} - \sigma^{\theta}(S_i)) + BG(S_i, 2\theta) - BG(S_i, 2\theta)$$

$$\sigma_T^{\theta}(S_k) \geq 0.63\, OPT_{k,\theta} - \left(\sum_{i=0}^{k-1} BG(S_i, 2\theta) - \sigma^{\theta}(S_k)\right)$$

Theoretical Analysis of Greedy

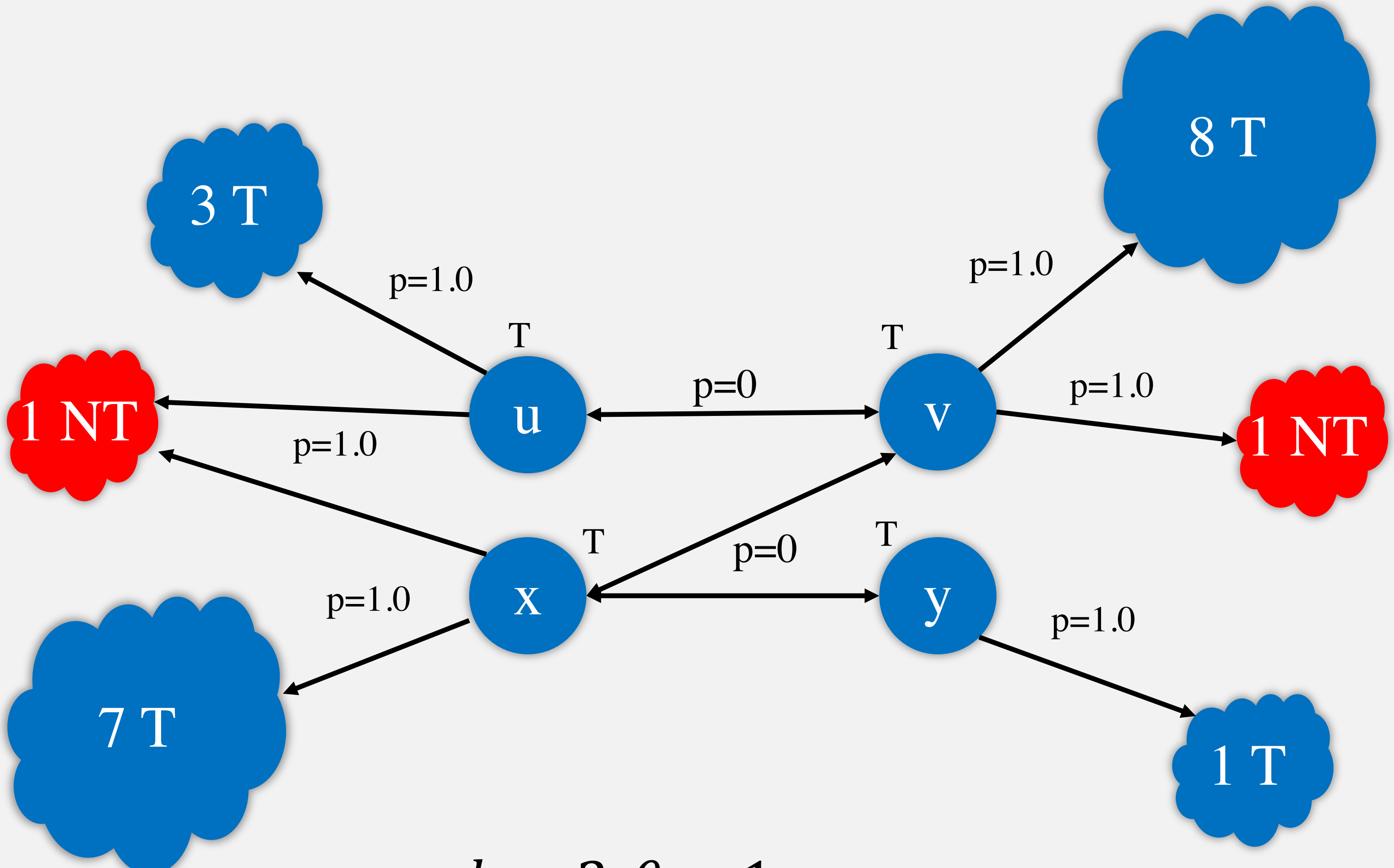
Additive Loss:

$$\sum_{i=0}^{k-1} BESTG(S_i, 2\theta) - \sigma_T^{\theta}(S_k)$$

Approximately difference of targets influenced between by the greedy solution with threshold θ and threshold 2θ

Runtime: $O(k \times |V| \times Time\ taken\ to\ compute\ \sigma)$

Can we improve on Greedy?

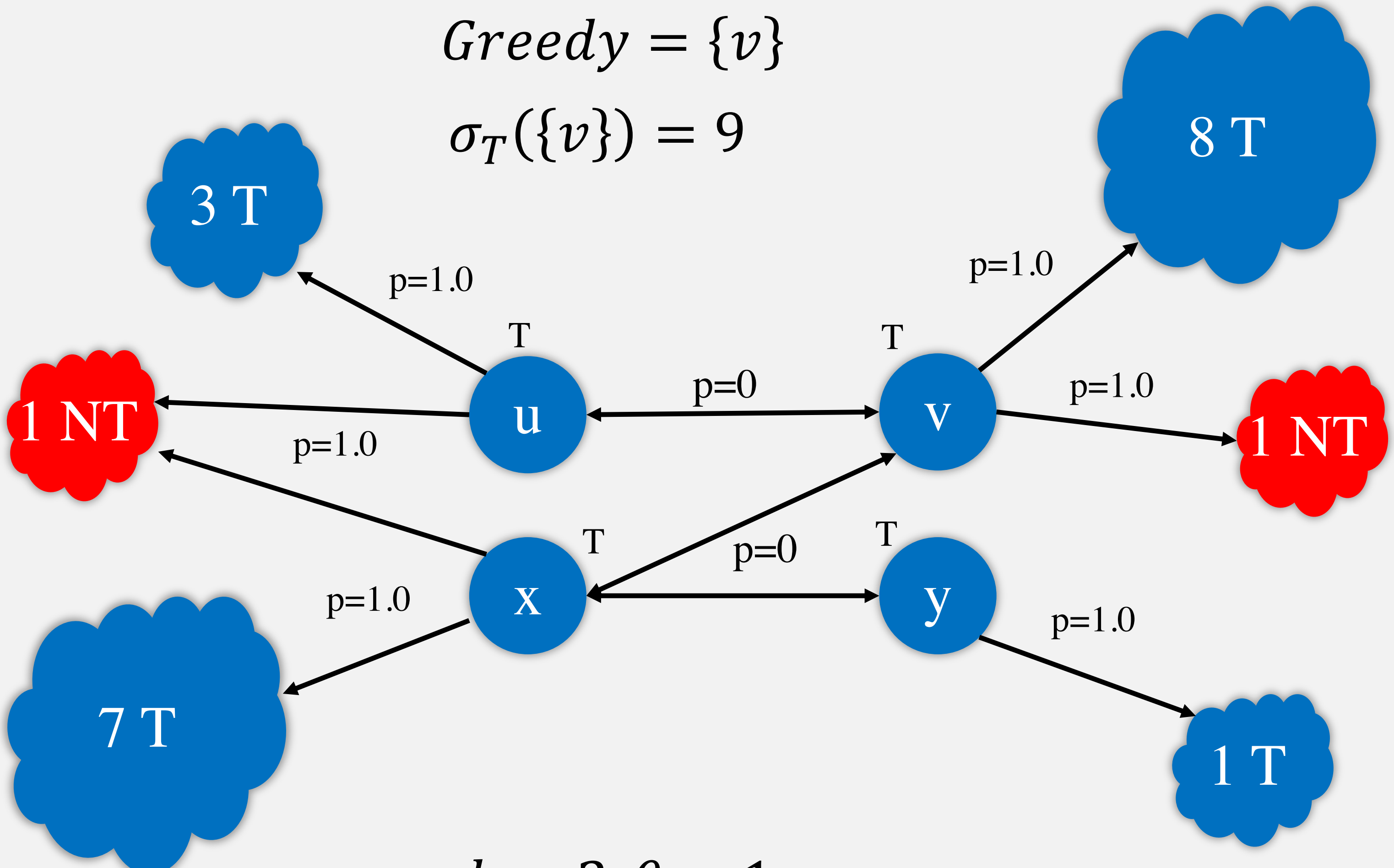


$$k = 2, \theta = 1$$

Can we improve on Greedy?

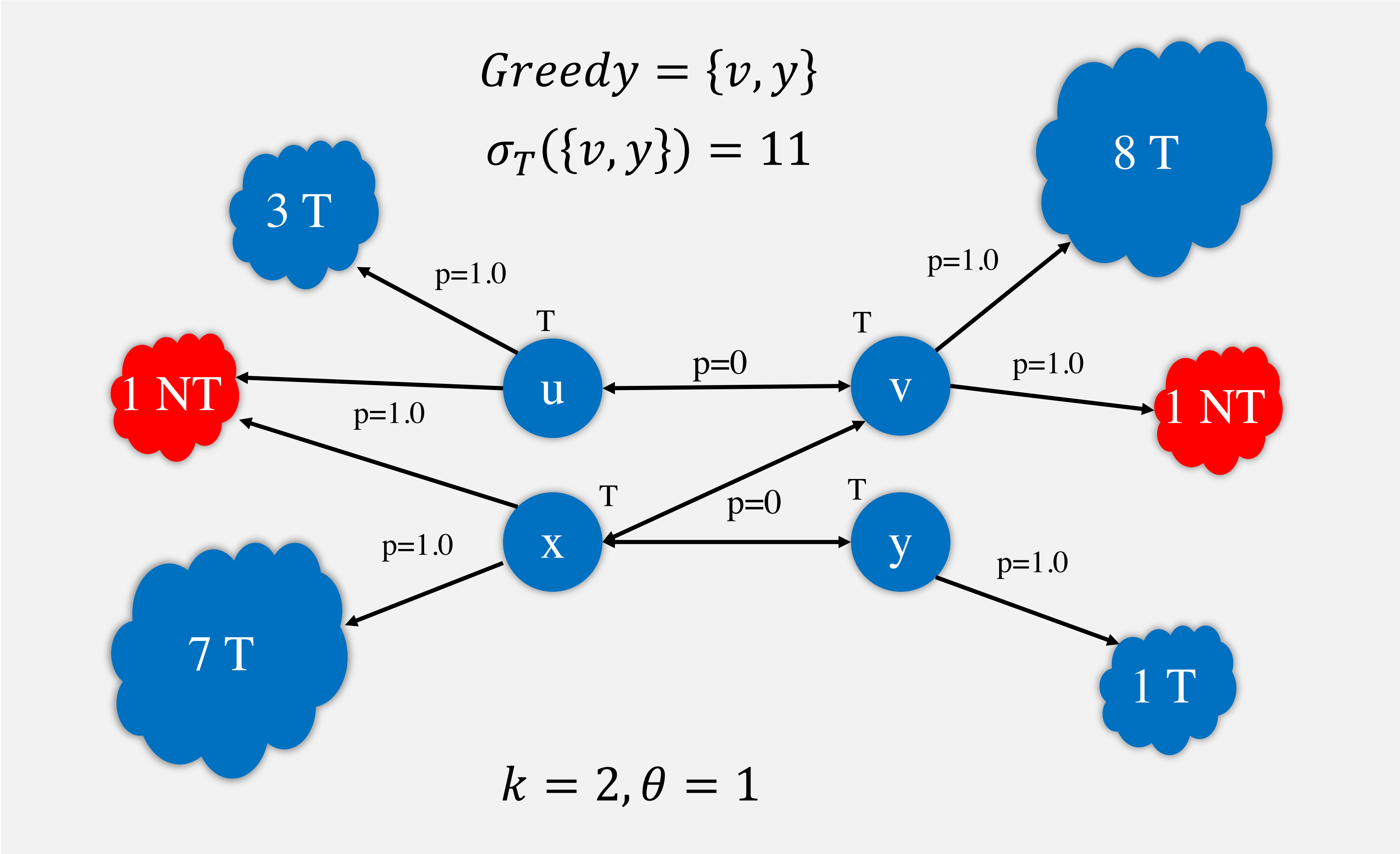
$$Greedy = \{v\}$$

$$\sigma_T(\{v\}) = 9$$

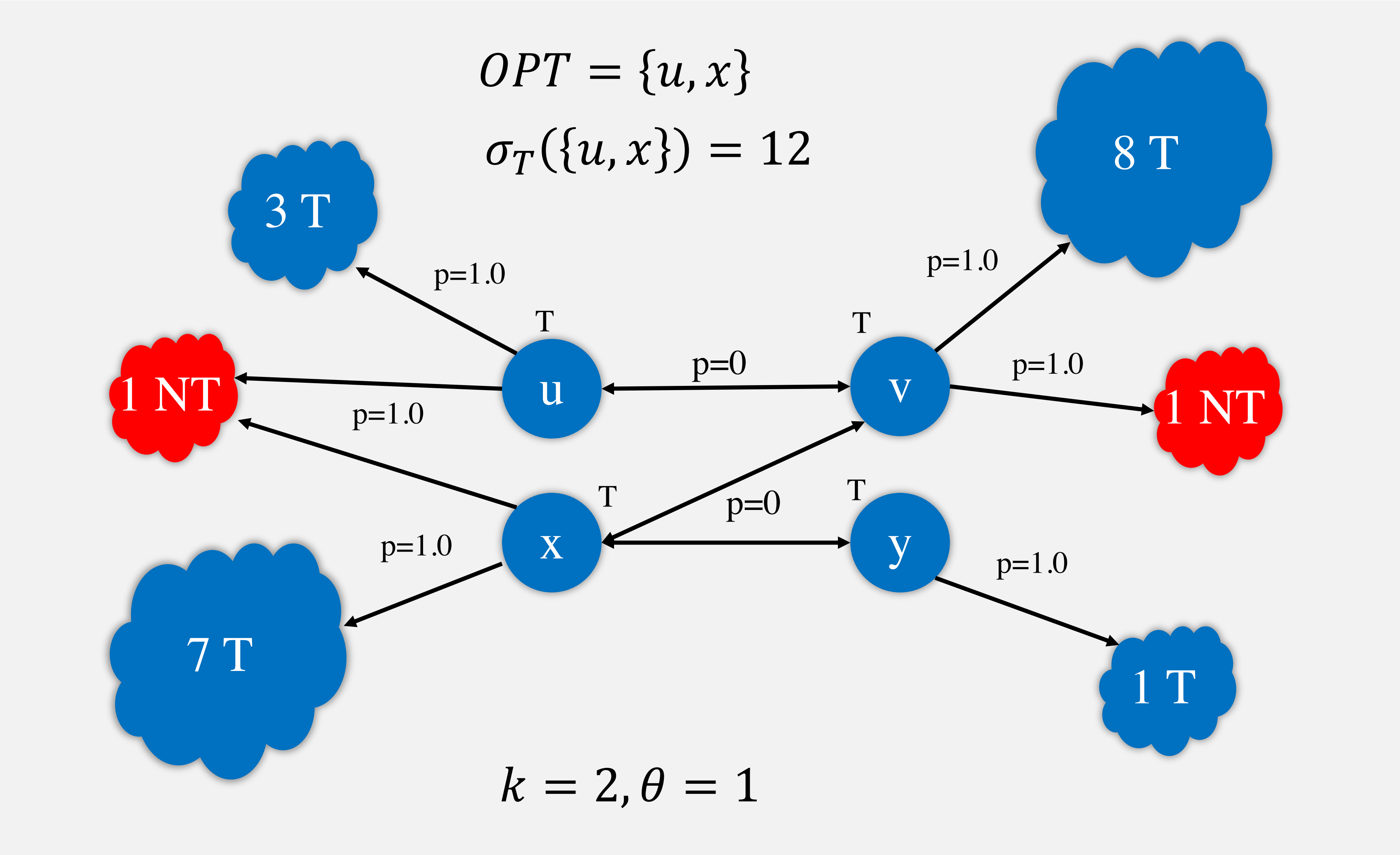


$$k = 2, \theta = 1$$

Can we improve on Greedy?



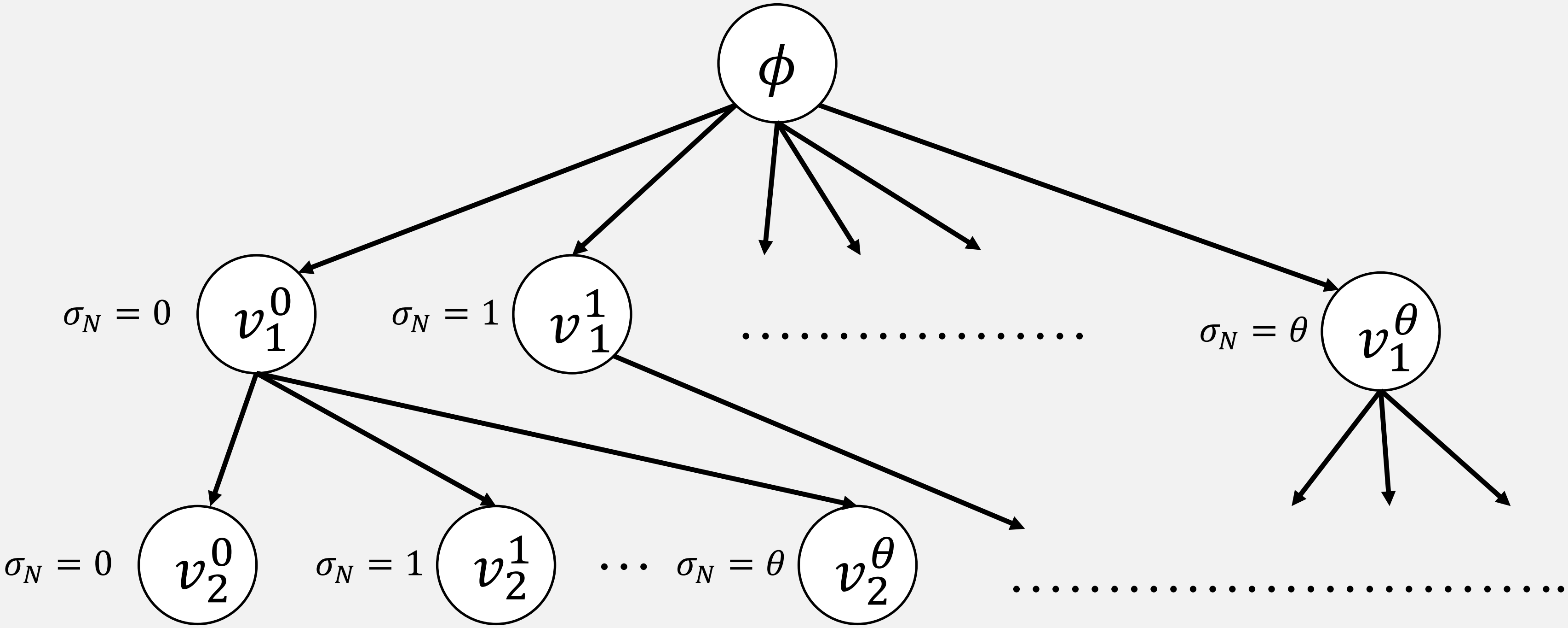
Can we improve on Greedy?



MULTIGREEDY ALGORITHM

MultiGreedy Algorithm

- Keeps track of multiple seed sets



Proceed till depth k and return the best path from root to leaf

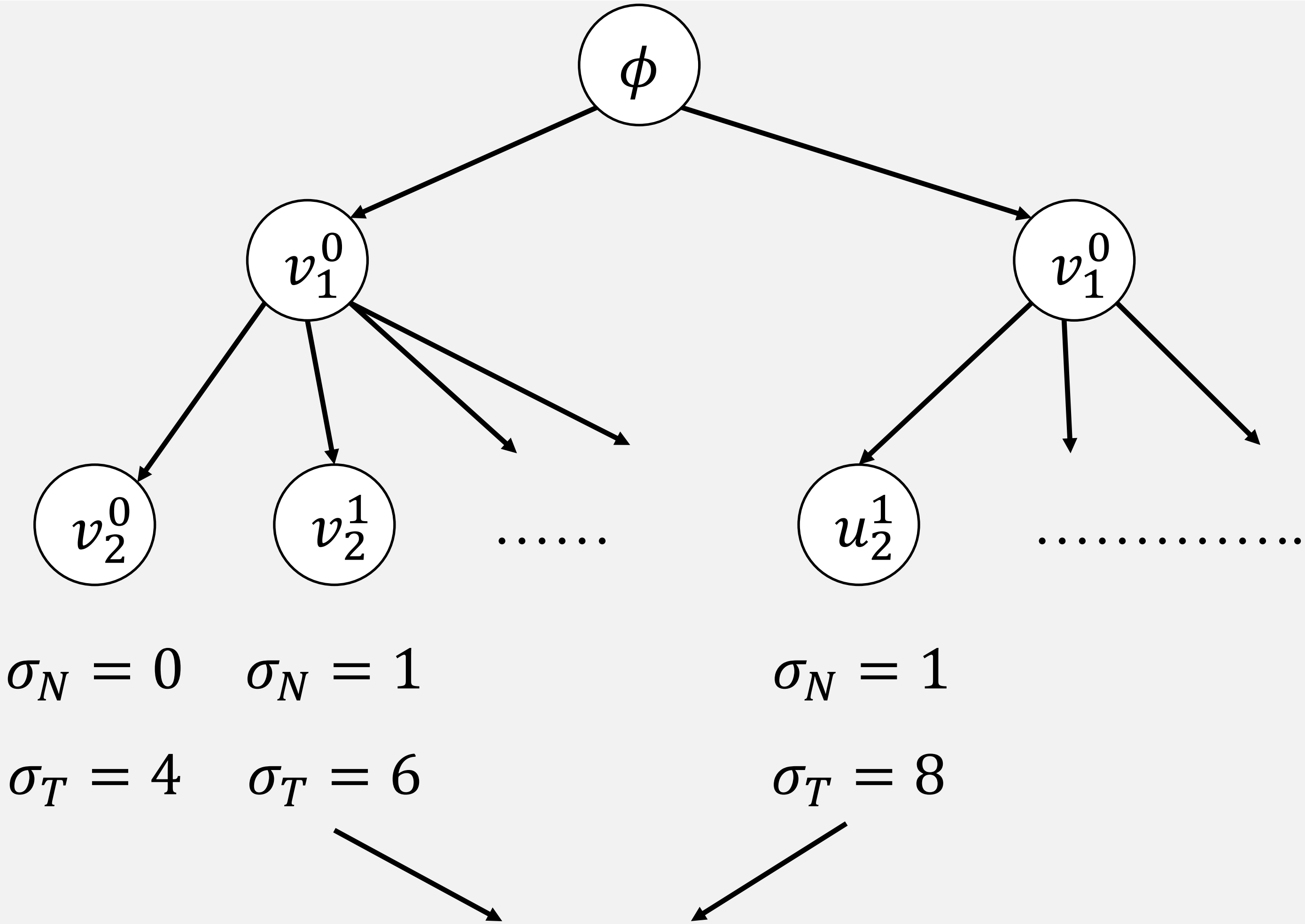
MultiGreedy Algorithm

- The greedy solution will be in one of the branches
- **Theorem:**

$$\sigma_T^\theta(MultiGreedy) \geq \sigma_T^\theta(Greedy)$$

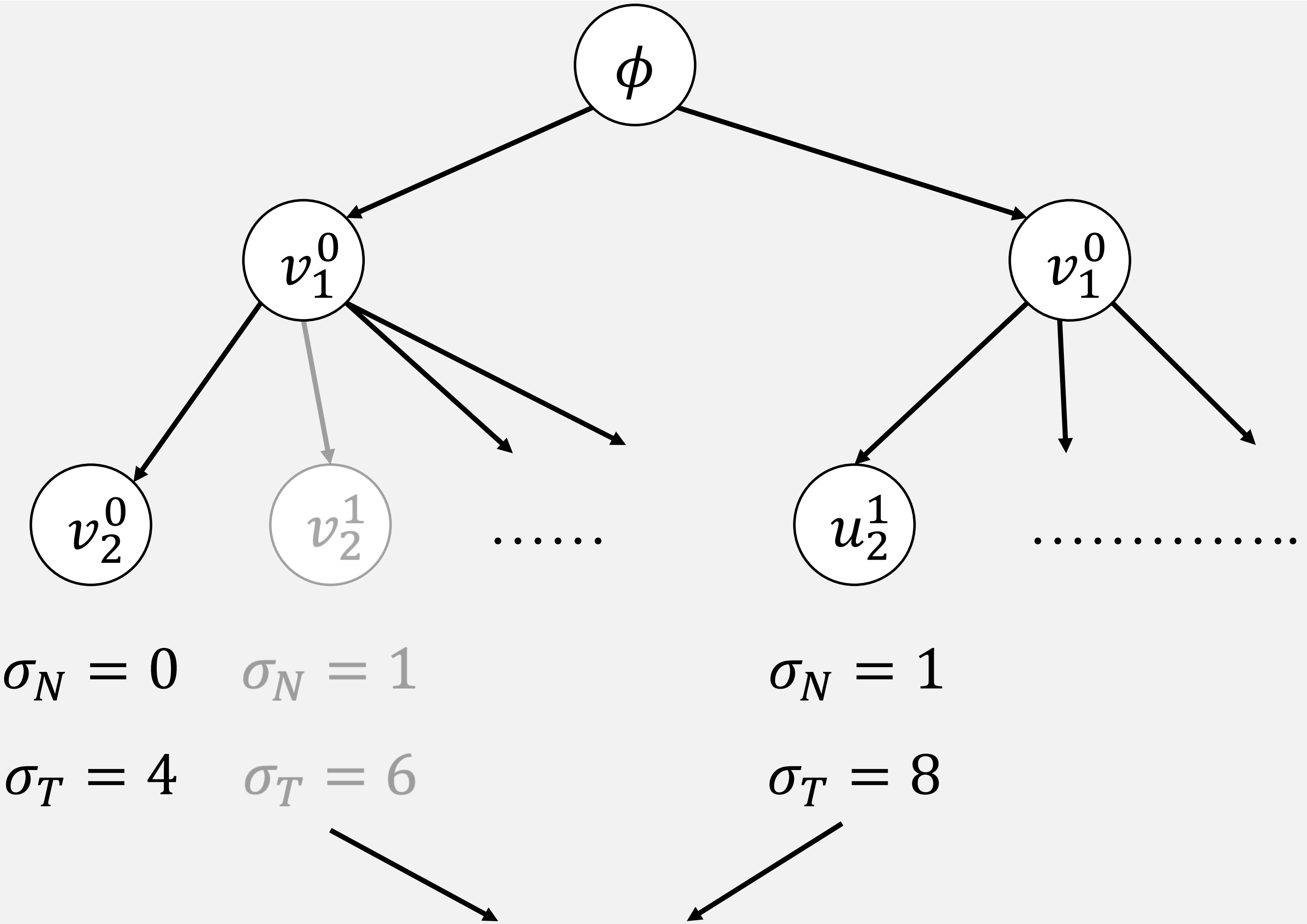
- Runtime: At least $\mathrm{O}(\theta^k)$
- Computationally infeasible!
- Let's prune the tree!

Efficient MultiGreedy with IMTree



Both hit 1 Non Target, but u_2^1 hits 8 Targets !

Efficient MultiGreedy with IMTree



Let's prune the branch that contains v_2^1

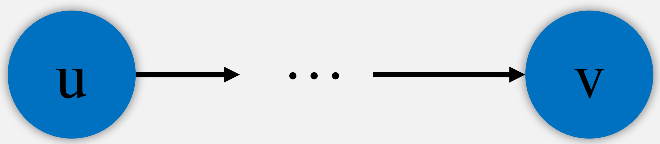
ESTIMATING INFLUENCE FUNCTION $\sigma(S)$

Estimating Influence Function

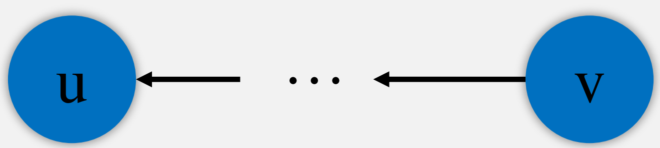
- Exact computation of $\sigma_T(S), \sigma_N(S)$ is #P-Hard
- Several techniques exists: Monte Carlo Simulations, Forward Influence Sketching, Reverse Influence Sketching(RIS)
- We've used RIS based estimation.
- Our algorithms can be adapted to different methodologies of estimating the influence function

Reverse Influence Sampling

- Let $g \sim G$ be a graph sampled from the random graph distribution
- $P[\, u\ influencing\ v] = P[\exists\ path\ from\ u\ to\ v\ in\ g]$



- Look at transpose g^T!
- $P[\, u\ influencing\ v] = P[\exists\ path\ from\ v\ to\ u\ in\ g^T]$



Random Reverse Reachable Set

- Randomly Select a vertex u.
- Generate a set R by performing a Random Reverse BFS starting from u
- $\sigma(S) = n \times P[S \cap R \neq \phi]$
- This observation was made by Borgs et. al.
- If sufficient samples are generated, $\sigma(S)$ can be accurately estimated with high probability.
- To estimate $\sigma_T(S), \sigma_N(S)$, we randomly select a Target, Non-Target respectively.

EXPERIMENTS

Experiment Results

- Exact computation of $\sigma_T(S), \sigma_N(S)$ is #P-Hard

- Several techniques exists: Monte Carlo Simulations, Forward Influence Sketching, Reverse Influence Sketching(RIS)

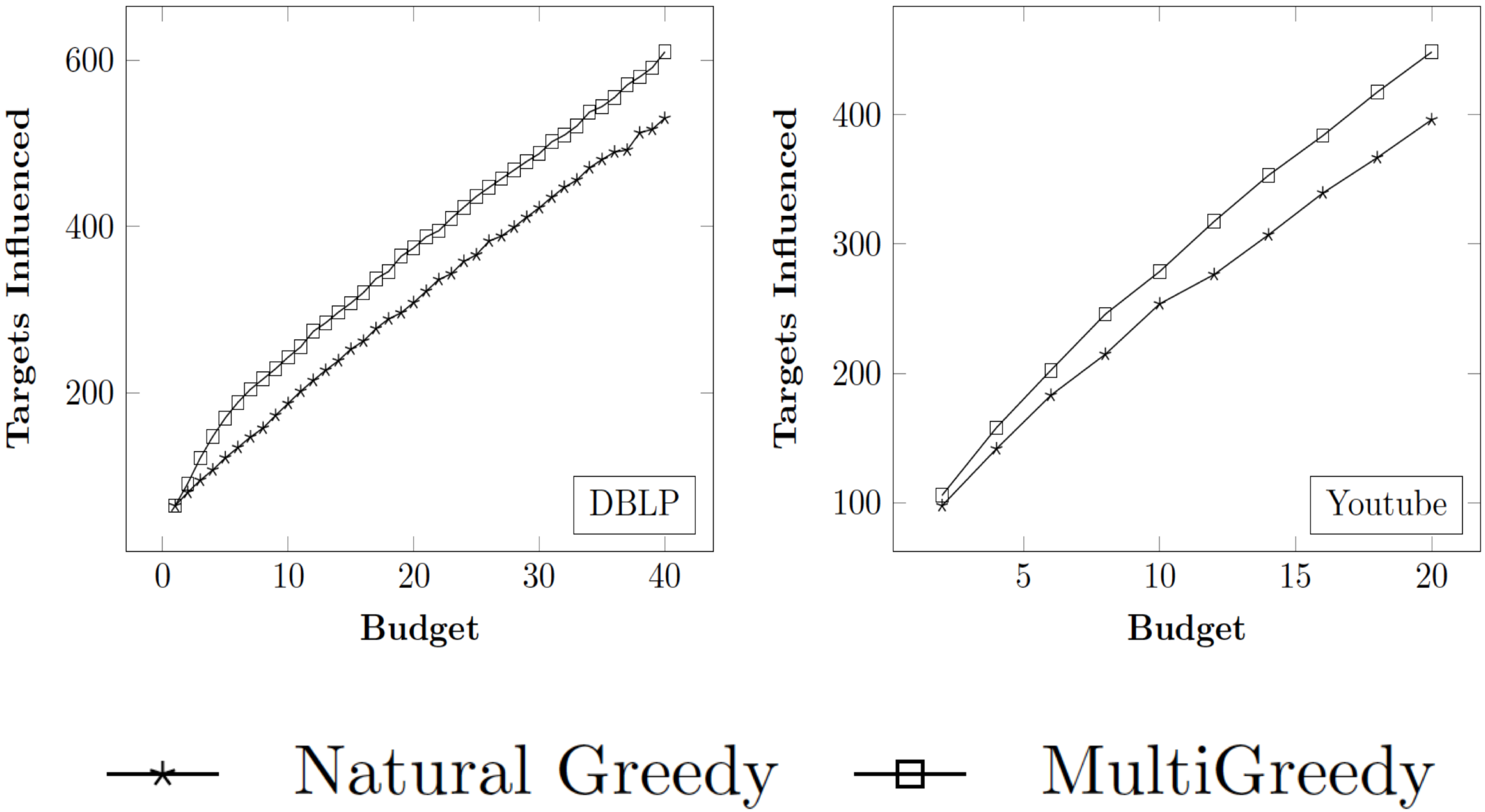
- We've used RIS based estimation.

- Our algorithms can be adapted to different methodologies of estimating the influence function

Datasets

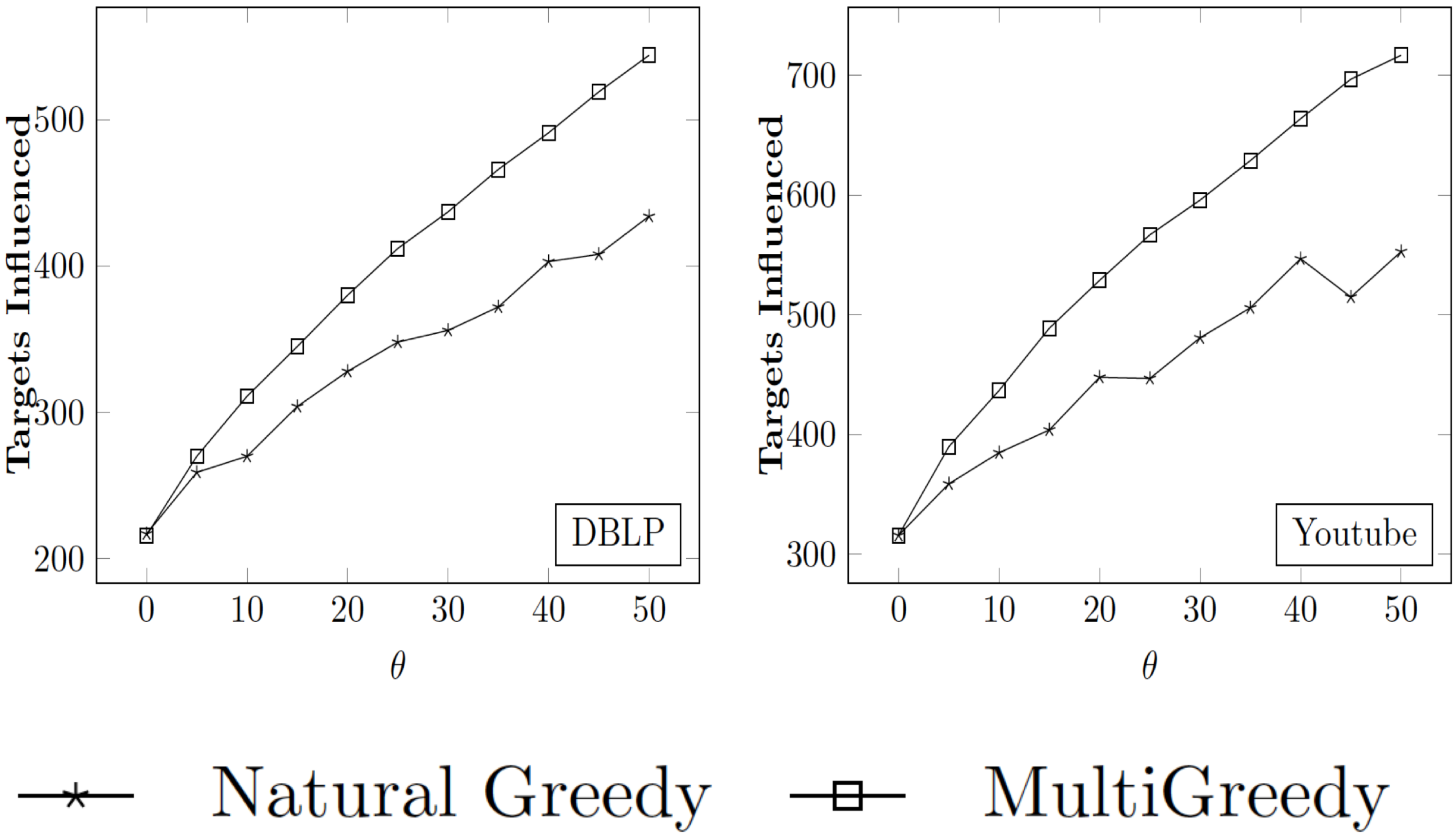
Network Name	**# Nodes**	**# Edges**
NetHept	15 k	62 k
Epinions	75 k	508 k
Amazon	334 k	925 k
DBLP	613 k	1.99 M
Youtube	1.13 M	2.98 M
Pokec	1.63 M	30.62 M

Budget Vs. Influence



$\theta = 10$

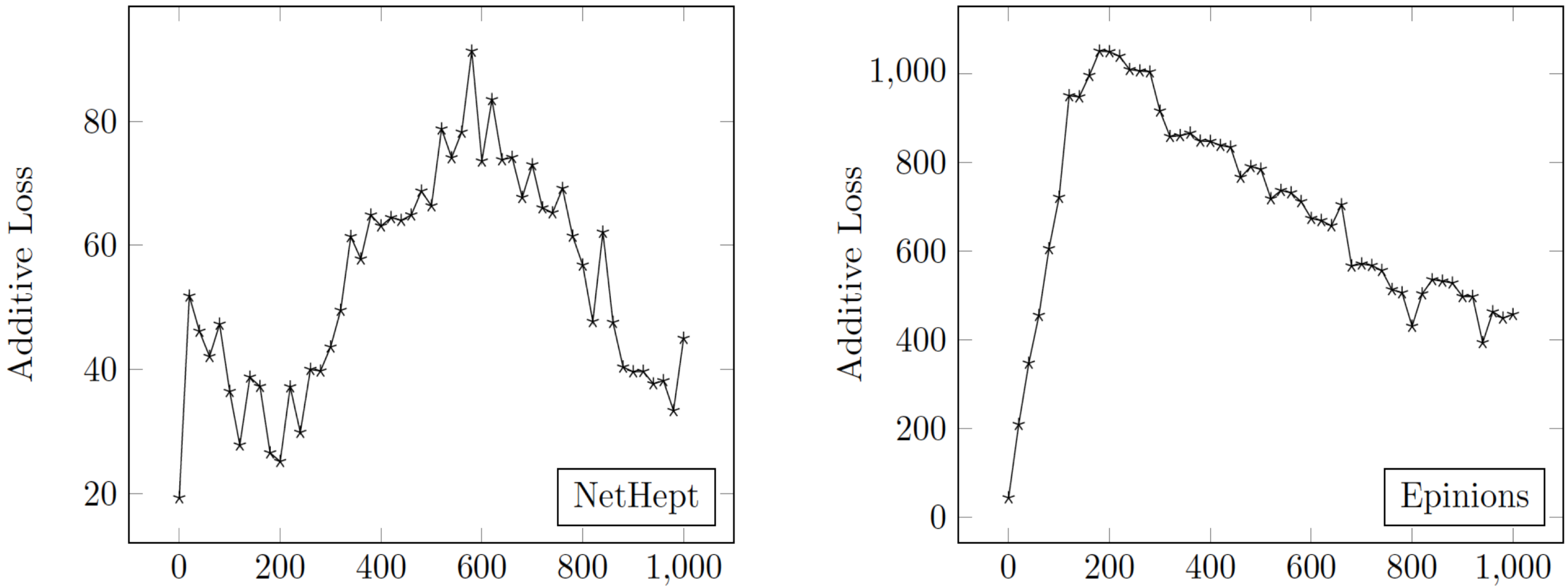
Threshold Vs. Influence



$k = 20$

Additive Loss in Natural Greedy

$$\sigma_T^{\theta}(S) \geq 0.63\ OPT - Additive\ Loss$$



$$\sigma_T^{\theta}(S) \geq 0.63\ OPT - Additive\ Loss$$

$$k = 20$$

Our Contributions

- Formulated the Constrained Influence Maximization (CIM) Problem
- Provided a theoretical analysis on hardness of CIM
- Studied the Greedy algorithm and proved its approximation guarantee involving an additive error
- Designed a novel MultiGreedy algorithm with an efficient implementation
- Experimentally evaluated Greedy, MultiGreedy algorithms on real world datasets

Future Work

- Can we design an algorithm that can tighten the additive error?
- Study how the additive error depends on the structure of the graph.

Thank you!

Presenter: Madhavan R.P

Email: madhavrp@iastate.edu

Research Group:

Pavan Aduri
Samik Basu
Xiaoyun Fu
Madhavan R.P