Mathematical Programming Models for Influence Maximization on Social Networks

S. Raghavan and Rui Zhang

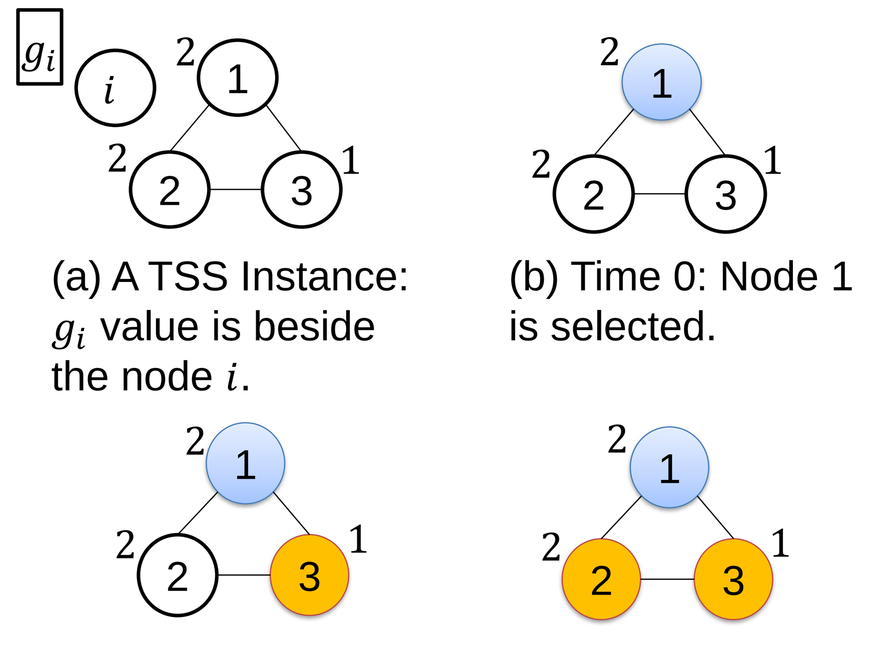
History of the Research

Fundamental problem in social network analysis: Whom do you target to maximize the adoption of a product/innovation? Also has applications in epidemiology.

Chen [2009] proposed the **Target Set Selection (TSS)** problem:

- Given a connected undirected graph $G = (V; E)$. For each $i \in V$, there is a threshold, g_i, which is between 1 and degree(i). All nodes are inactive initially.
- Select a subset of nodes, the target set, and they become active immediately.
- After that, in each step, an inactive node i becomes active if at least g_i of its neighbors are active in the previous step.
- ***Goal: Find the minimum target set while ensuring that all nodes are active at the end of this diffusion process.***

All previous work has focused on heuristics and approximation algorithms.



(a) A TSS Instance: g_i value is beside the node i.

(b) Time 0: Node 1 is selected.

(c) Time 1: Node 3 is active.

(d) Time 2: Node 2 is active.

We want to apply mathematical programming techniques to develop EXACT approaches.

Future of the Research

First researchers to study exact solution approaches to problem and to solve large real world size instances to optimality! Focus of future research:

- Proportion requirements---fraction α of population is influenced (instead of 100% adoption).
- Latency constraints---time constraints on diffusion process.
- Time varying networks---connections change over time.
- Robust target sets/variants under stochastic influence factors.
- Applications in epidemiology.
- Voting models and opinion formation on social networks.

Current State of the Research

We study three fundamental problems.
All three problems are NP-hard (APX-hard).

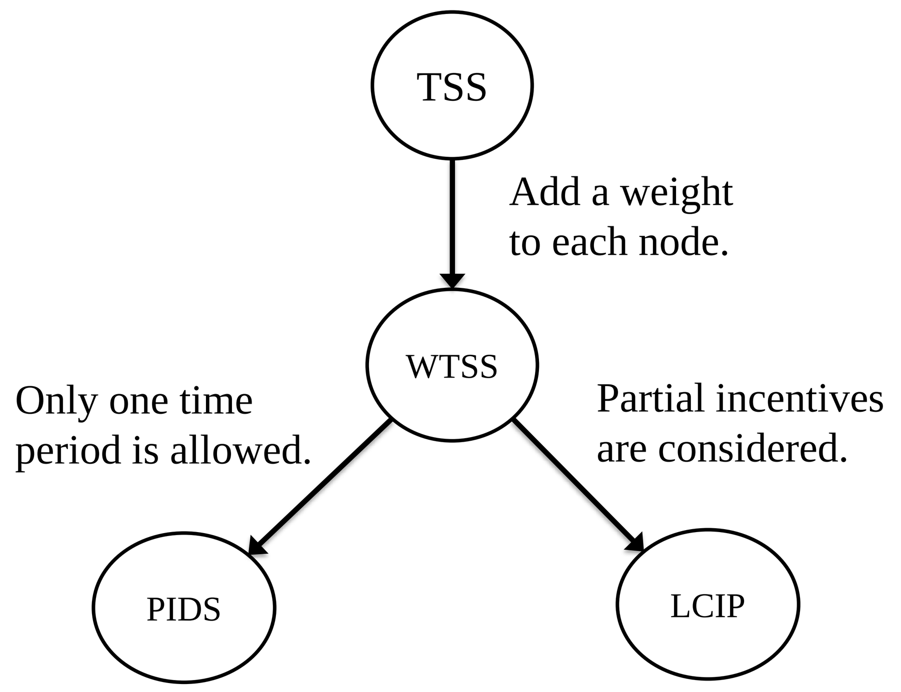
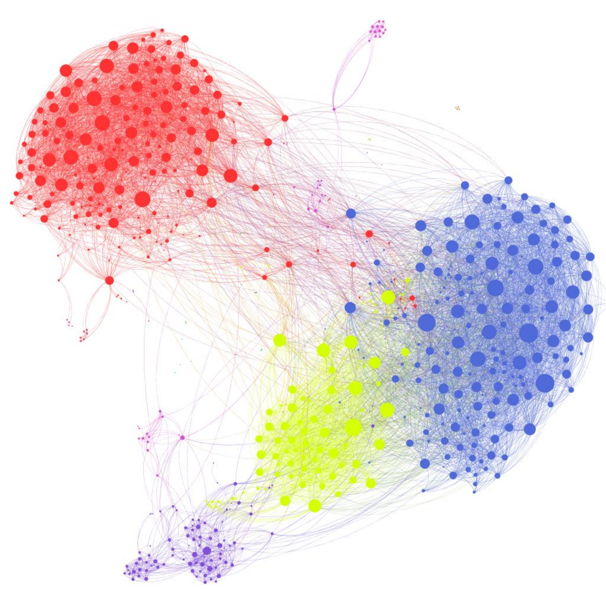


Fig: Relationships among these three problems.

Our Research Scheme

Start with tree graphs:

- Can we solve it?
- Can we find the strongest integer programming (IP) formulation for it?
- Can we study its polytope?

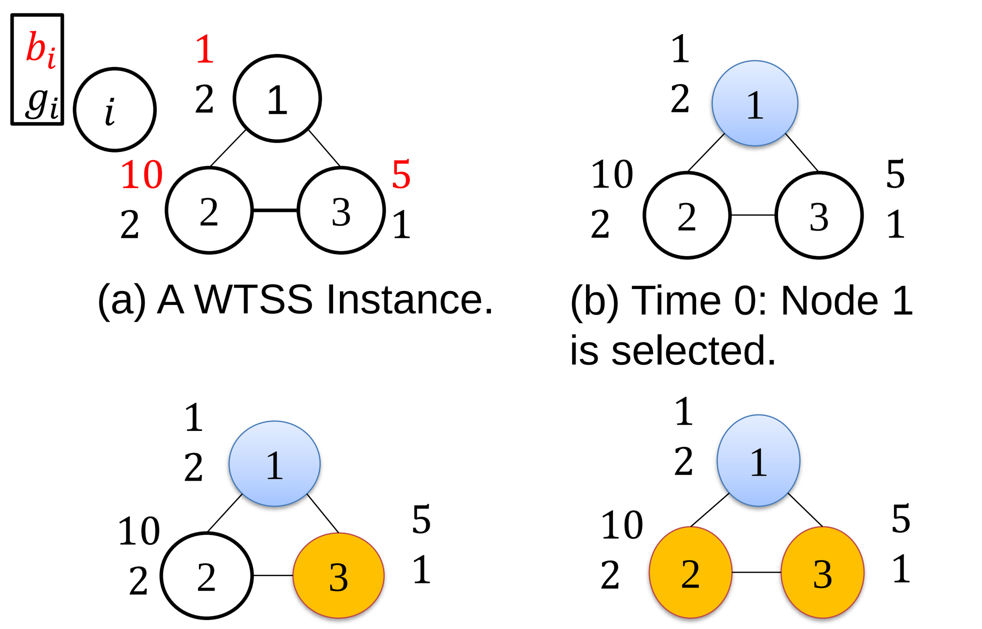


Turn to general graphs:

- How to apply the "good" tree formulation here?
- Can we develop Branch-and-Cut approaches?

The Weighted Target Set Selection (WTSS) Problem

A weight (cost) b_i for each node $i \in V$ (different nodes require different levels of effort).



(a) A WTSS Instance.

(b) Time 0: Node 1 is selected.

(c) Time 1: Node 3 is active.

(d) Time 2: Node 2 is active.

Our results for the WTSS and PIDS problems:

On trees:

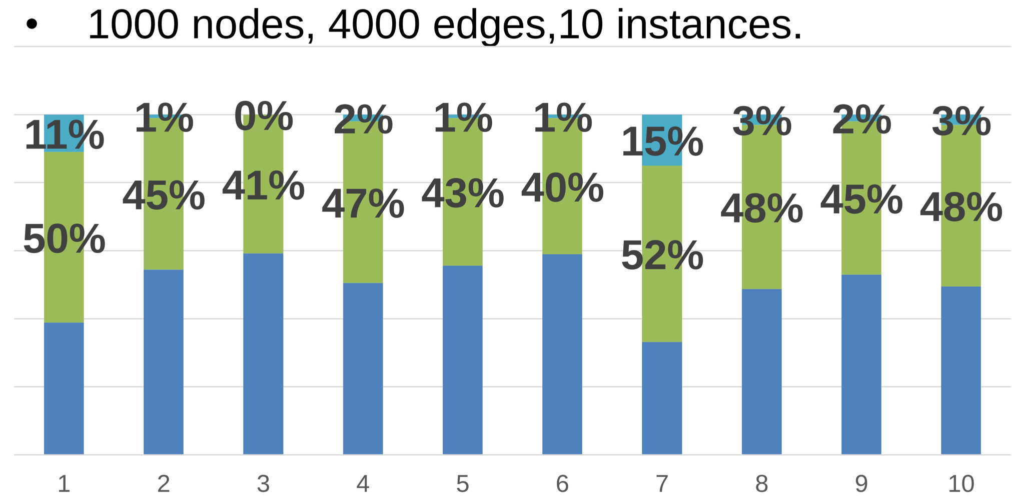
- A linear time algorithm.
- A tight and compact extended IP formulation.
- A complete description of the polytope.

On general graphs:

- A strong IP formulation.
- A specialized Branch-and-Cut approach.

Some computational results for the WTSS:

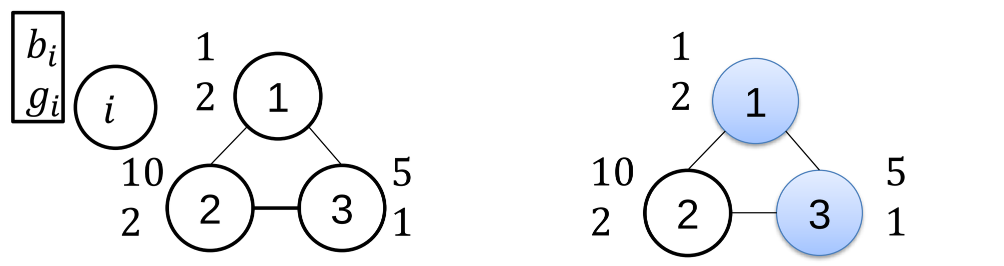
- 1000 nodes, 4000 edges,10 instances.



Blue: previous formulation. Green: gap we closed. Aqua: the remaining gap.

The Positive Influence Dominating Set (PIDS) Problem

Only one time period is allowed for diffusion.



(a) A PIDS Instance.

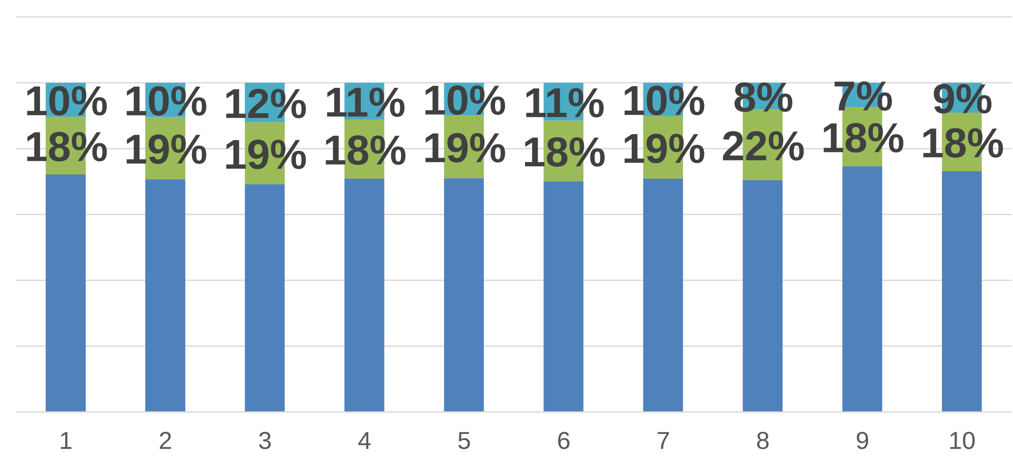
(b) Time 0: Node 1 and node2 are selected.

Instances and Environment:

- Generate networks à la Watts and Strogatz [1998] model.
- Randomly generate b_i and g_i for each node i.
- CPLEX 12.6, Python API, Intel i5 3.40GHz, 24 GB ram, Ubuntu.

Some computational results for the PIDS problem:

- 200 nodes, 800 edges,10 instances.



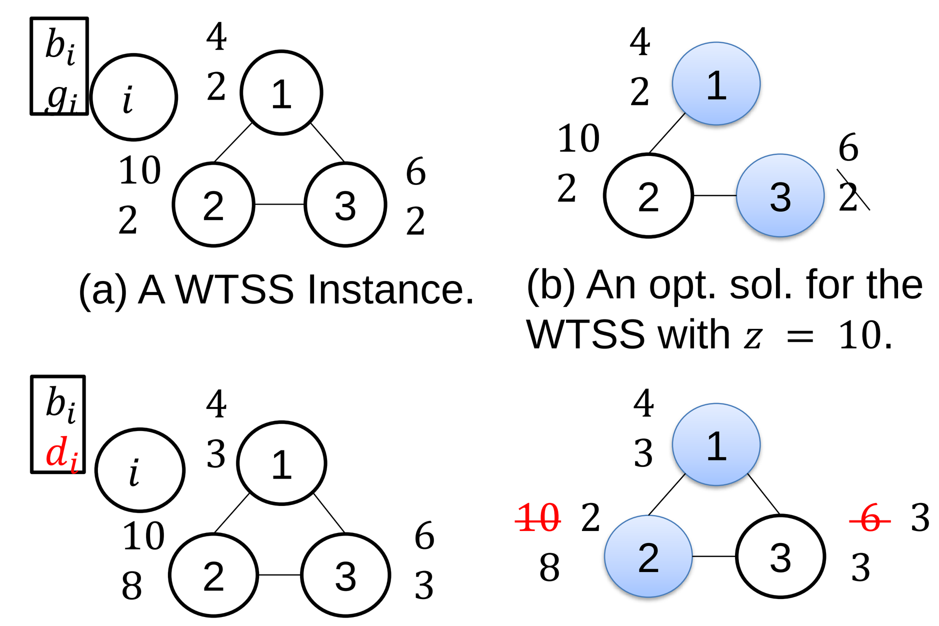
Blue: previous formulation. Green: gap we closed. Aqua: the remaining gap.

Our results for the LCIP:

- A totally unimodualr matrix (TUM) formulation on trees.
- A Branch-and-Cut approach for general graphs based on the TUM formulation and the observation that the influence propagation graph must be a directed acyclic graph.

The Least Cost Influence Problem (LCIP)

- Influence factor d_{ij} represents the influence from node j to node i such that its threshold reduced by d_{ij} after node j becomes active.
- In the WTSS, $d_{ij} = d_i$ and $g_i = \lceil \frac{b_i}{g_i} \rceil$.
- In the LCIP, partial incentives are considered.



(a) A WTSS Instance.

(b) An opt. sol. for the WTSS with $z = 10$.

(c) A LCIP instance

(d) An opt. sol. for the LCIP with $z = 6$.

Some computational results for the LCIP:

Tricks of the Branch-and-Cut approach:

- **BC**: Branch-and-Cut with cycle separation.
- **H**: heuristics for initial solutions.
- **C**: prioritize branching over separation.
- **B**: a specialized branching rule.
- **P**: perturbations for symmetry elimination.

10 ***Facebook*** graph instances with 4039 nodes and 88234 edges run for 1 hour limit.

	Opt. #	Opt. Run. Time (S)	Avg. Gap
BC-H	0	NA	36.7%
BC-H-C	2	1537	5.8%
BC-H-C-B	2	2535	3.8%
BC-H-C-B-P	5	2271	0.4%