On positive influence dominating sets in social networks

Feng Wang, Hongwei Du, Erika Camacho, Kuai Xu, Wonjun Lee, Yan Shi, and Shan Shan


Speaker: Joseph, Chuang-Chieh Lin
Supervisor: Professor Maw-Shang Chang

Computation Theory Laboratory
Department of Computer Science and Information Engineering
National Chung Cheng University, Taiwan

March 15, 2011
1 Introduction

2 Complexity of PIDS

3 An $H(\Delta)$-approximation algorithm for PIDS
1 Introduction

2 Complexity of PIDS

3 An $H(\Delta)$-approximation algorithm for PIDS
The dominating set problem

Input: Given a graph $G = (V, E)$ and an integer $k$.

Task: Find a subset $D \subseteq V$ of size $\leq k$ such that each vertex in $V \setminus D$ is adjacent to (i.e., dominated by) at least one vertex in $D$. 

Recent issues in social networks related to dominating sets

- A \((1 + O(1))\)-approximation algorithm to the dominating set in a power-law graph [SODA’2004].

- Another optimization problem:

**The Positive Influence Dominating Set problem (PIDS)**

- **Input**: Given a graph \(G = (V, E)\)
- **Task**: Find a subset \(D \subseteq V\) such that any \(v \in V\) is dominated by at least \(\left\lceil \frac{d(v)}{2} \right\rceil\) vertices.
Applications of PIDS

- It’s helpful for the success of intervention programs for a certain type of social problem (e.g., drinking, smoking, drug related issues...)

- For example, a binge drinker can be converted to an abstainer through intervention programs and have positive impact on his direct friends.

- However, he (she) might turn back into a binge drinker and have negative impact on his (her) direct friends if many of his (her) direct friends are binge drinkers.
Applications of PIDS

- It’s helpful for the success of intervention programs for a certain type of social problem (e.g., drinking, smoking, drug related issues...)

- For example, a binge drinker can be converted to an abstainer through intervention programs and have positive impact on his direct friends.

- However, he (she) might turn back into a binge drinker and have negative impact on his (her) direct friends if many of his (her) direct friends are binge drinkers.
Applications of PIDS

- It’s helpful for the success of intervention programs for a certain type of social problem (e.g., drinking, smoking, drug related issues...)

- For example, a binge drinker can be converted to an abstainer through intervention programs and have positive impact on his direct friends.

- However, he (she) might turn back into a binge drinker and have negative impact on his (her) direct friends if many of his (her) direct friends are binge drinkers.
Contribution of this paper

- Explaining the application of PIDS in social networks.
- Showing that PIDS is APX-hard.
  - Though it is still open that whether PIDS is in APX or not.
- Providing a greedy $H(Δ)$-approximation algorithm for PIDS.
- Conjecture that PIDS is easier in a power-law graph
  - Most social networks follow the power-law.
1. Introduction

2. Complexity of PIDS

3. An $H(\Delta)$-approximation algorithm for PIDS
Theorem 2.1

*PIDS is APX-hard.*

Theorem 2.2 (Du & Ko 2000)

*The vertex cover problem in cubic graph (VC-cubic) is APX-complete.*

We construct an $L$-reduction from VC-cubic to PIDS.
Theorem 2.3

*PIDS is APX-hard.*

Theorem 2.4 (Du & Ko 2000)

*The vertex cover problem in cubic graph (VC-cubic) is APX-complete.*

We construct an $L$-reduction from VC-cubic to PIDS.

- Wait a minute... What is “APX” and what is an $L$-reduction?
Theorem 2.3

**PIDS is APX-hard.**

Theorem 2.4 (Du & Ko 2000)

*The vertex cover problem in cubic graph (VC-cubic) is APX-complete.*

We construct an $L$-reduction from VC-cubic to PIDS.

- Wait a minute... What is “APX” and what is an $L$-reduction?
APX & $L$-reduction

**APX, APX-hard, & APX-complete**

- APX (an abbreviation of "approximable") is the set of NP optimization problems that allow constant-factor polynomial-time approximation algorithms.
- A problem $\mathcal{P}$ is APX-complete: $\mathcal{P} \in \text{APX} \&$ there exists a PTAS-reduction from every problem in APX to $\mathcal{P}$ (APX-hard).

- Note that PTAS = polynomial-time approximation scheme
- Whenever an APX-hard problem is shown to possess a PTAS, every problem in APX has a PTAS.
**L-reduction (a kind of PTAS-reduction)**

- **A and B**: two optimization problems;
- **$c_A$ and $c_B$**: the cost functions of $A$ and $B$ respectively.
- A pair of functions $f$ and $g$ is an $L$-reduction if all of the following conditions hold:
  - $f$ and $g$ are computable in polynomial time;
  - if $x$ is an instance of problem $A$, then $f(x)$ is an instance of problem $B$;
  - if $y$ is a solution to $f(x)$, then $g(y)$ is a solution to $x$;
  - there exists $\alpha > 0$ such that
    \[ |OPT_B(f(x))| \leq \alpha \cdot |OPT_A(x)|; \]
  - there exists $\beta > 0$ such that for every solution $y$ to $f(x)$
    \[ |OPT_A(x) - c_A(g(y))| \leq \beta \cdot |OPT_B(f(x)) - c_B(y)|. \]
The proof of the APX-hardness of PIDS

\begin{center}
\begin{tikzpicture}
    \node (v0) at (0,0) [circle,fill,inner sep=2pt] {$v_0$};
    \node (v1) at (1,0) [circle,fill,inner sep=2pt] {$v_1$};
    \node (v2) at (1,1) [circle,fill,inner sep=2pt] {$v_2$};
    \node (v3) at (0,1) [circle,fill,inner sep=2pt] {$v_3$};
    \node (v4) at (0,2) [circle,fill,inner sep=2pt] {$v_4$};
    \node (v5) at (1,2) [circle,fill,inner sep=2pt] {$v_5$};
    \node (v6) at (0.5,1.5) [circle,fill,inner sep=2pt] {$v_7$};
    \node (v7) at (1.5,1.5) [circle,fill,inner sep=2pt] {$v_8$};
    \node (v8) at (0.5,0.5) [circle,fill,inner sep=2pt] {$v_9$};
    \node (v9) at (1.5,0.5) [circle,fill,inner sep=2pt] {$v_{10}$};
    \node (v10) at (0,0.5) [circle,fill,inner sep=2pt] {$v_{11}$};
    \node (v11) at (1,0.5) [circle,fill,inner sep=2pt] {$v_{12}$};

    \draw (v0) -- (v1);
    \draw (v0) -- (v2);
    \draw (v1) -- (v2);
    \draw (v2) -- (v3);
    \draw (v3) -- (v1);
    \draw (v4) -- (v5);
    \draw (v5) -- (v6);
    \draw (v6) -- (v7);
    \draw (v7) -- (v8);
    \draw (v8) -- (v9);
    \draw (v9) -- (v10);
    \draw (v10) -- (v11);
    \draw (v11) -- (v12);
    \draw (v12) -- (v13);

    \foreach \i in {0,1,2,3,4,5,6,7,8,9,10,11,12}
    \foreach \j in {0,1,2}
    \draw (v\i) -- (v\j);
\end{tikzpicture}
\end{center}
Claim 1

$G$ has a vertex cover of size $\leq k \iff G'$ has a positive influence dominating set of size $\leq k + 9n$ (where $n = |V|$).

- Suppose that $G$ has a vertex cover $C$ of size $k$.
- Let $D = C \cup \{a_e, b_e \mid e \in E\} \cup \{p_v, q_v, p'_v, q'_v, p''_v, q''_v \mid v \in V\}$.
- Note that $|E| = 3|V|/2$ ($\because G$ is a cubic graph).
- $|D| = |C| + 2 \cdot |E| + 6 \cdot |V| = k + 9n$.
- $C$ is a vertex cover of $G \iff D$ is a PIDS of $G'$.
Claim 1

*G has a vertex cover of size \( \leq k \) \iff \( G' \) has a positive influence dominating set of size \( \leq k + 9n \) (where \( n = |V| \)).*

- Suppose that \( G \) has a vertex cover \( C \) of size \( k \).
- Let \( D = C \cup \{ a_e, b_e \mid e \in E \} \cup \{ p_v, q_v, p'_v, q'_v, p''_v, q''_v \mid v \in V \} \).
- Note that \( |E| = 3|V|/2 \) (\( \because \) \( G \) is a cubic graph).
- \( |D| = |C| + 2 \cdot |E| + 6 \cdot |V| = k + 9n \).
- \( C \) is a vertex cover of \( G \) \iff \( D \) is a PIDS of \( G' \).
The proof of the APX-hardness of PIDS (contd.)

Claim 1

\[ G \text{ has a vertex cover of size } \leq k \iff G' \text{ has a positive influence dominating set of size } \leq k + 9n \text{ (where } n = |V|). \]

- Suppose that \( G \) has a vertex cover \( C \) of size \( k \).
- Let \( D = C \cup \{a_e, b_e | e \in E\} \cup \{p_v, q_v, p'_v, q'_v, p''_v, q''_v | v \in V\} \).
- Note that \( |E| = 3|V|/2 \) (\( \because G \) is a cubic graph).
- \( |D| = |C| + 2 \cdot |E| + 6 \cdot |V| = k + 9n \).
- \( C \) is a vertex cover of \( G \) \iff \( D \) is a PIDS of \( G' \).
opt_{VC-cubic}(G): the size of the minimum vertex cover of $G$;
opt_{PIDS}(G')$: the size of the minimum PIDS of $G'$.

- $opt_{VC-cubic}(G) \iff opt_{PIDS}(G') = opt_{VC-cubic}(G) + 9n$.
- $n/2 = |E|/3 \leq opt_{VC-cubic}(G)$ ($\because \forall v \in V(G), \deg(v) = 3$).
- Plugging $n = (opt_{PIDS}(G') - opt_{VC-cubic}(G))/9$ into the inequality, we have

$$opt_{PIDS}(G') \leq 19 \cdot opt_{VC-cubic}(G).$$
From the proof of Claim 2, we see that if \( G' \) has a PIDS, then we can construct in polynomial time, a vertex cover \( C = D \cap V \) of \( G \) with size \( |D| - 9n \).

Therefore,

\[
||C| - \text{opt}_{VC-cubic}(G)| = ||D| - \text{opt}_{PIDS}(G')|.
\]
1 Introduction

2 Complexity of PIDS

3 An $H(\Delta)$-approximation algorithm for PIDS
- $n_A(v)$: the number of neighbors of $v$ in $A$ for any vertex subset $A \subseteq V$.

- We denote $h(v) = \lceil \deg(v)/2 \rceil$.

- $f(A) = \sum_{v \in V} \min(h(v), n_A(v))$. 

Important properties of $f$

- $n_A(v)$: the number of neighbors of $v$ in $A$ for any vertex subset $A \subseteq V$.
- We denote $h(v) = \lceil \text{deg}(v)/2 \rceil$.

$\star f(A) = \sum_{v \in V} \min(h(v), n_A(v))$.

**Lemma 3.1**

- $f(\emptyset) = 0$;
- $f(A) = \sum_{v \in V} h(v)$ if and only if $A$ is a positive influence dominating set;
- If $f(A) < \sum_{v \in V} h(v)$, then there exists a vertex $u \in V \setminus A$ such that $f(A \cup \{u\}) > f(A)$. 
The greedy algorithm

Greedy Algorithm

1: $A \leftarrow \emptyset$;
2: while $f(A) < \sum_{v \in V} h(v)$ do
3: choose $u \in V \setminus A$ to maximize $f(A \cup \{u\})$;
   set $A \leftarrow A \cup \{u\}$;
4: end while
5: output $A$. 
Theorem 3.2 (Wolsey 1982)

Suppose that $f$ is a monotone increasing, submodular integer function with $f(\emptyset) = 0$. Then the above Greedy Algorithm produces an approximation solution with a factor of $H(\gamma)$ from optimal, where $\gamma = \max_{v \in V} f(\{v\})$ and $H(\gamma) = \sum_{i=1}^{\gamma} \frac{1}{i}$. 
Definition 3.3

(1) $f$ is monotone increasing if $A \subset B$ implies $f(A) \leq f(B)$.

(2) $f$ is submodular if for any two subsets $A$ and $B$,

$$f(A) + f(B) \geq f(A \cup B) + f(A \cap B).$$

(1) Suppose $A \subset B$, then $n_A(v) \leq n_B(v)$ for all $v \in V$. Hence

$$f(A) = \sum_{v \in V} \min(h(v), n_A(v))$$

$$\leq \sum_{v \in V} \min(h(v), n_B(v))$$

$$= f(B).$$

(2)
\( f \) is submodular iff for \( u \notin B, A \subset B \) implies \( \delta_u f(A) \geq \delta_u f(B) \) where 
\[
\delta_u f(A) = f(A \cup \{u\}) - f(A).
\]

[Ding-Zhu Du et al. SODA’2008]

We have
\[
\delta_u f(A) = \sum_{v \in V} \left[ \min(h(v), n_{A \cup \{u\}}(v)) - \min(h(v), n_A(v)) \right],
\]
\[
\delta_u f(B) = \sum_{v \in V} \left[ \min(h(v), n_{B \cup \{u\}}(v)) - \min(h(v), n_B(v)) \right].
\]

For \( u \notin B \) and \( A \subset B \), we have
\[
n_A(v) \leq n_B(v) \text{ and } n_{A \cup \{u\}}(v) \leq n_{B \cup \{u\}}(v).
\]
(2) $f$ is submodular iff for $u \notin B$, $A \subset B$ implies $\delta_u f(A) \geq \delta_u f(B)$ where $\delta_u f(A) = f(A \cup \{u\}) - f(A)$. [Ding-Zhu Du et al. SODA'2008]

- We have

$$\delta_u f(A) = \sum_{v \in V} \left[ \min(h(v), n_{A \cup \{u\}}(v)) - \min(h(v), n_A(v)) \right],$$

$$\delta_u f(B) = \sum_{v \in V} \left[ \min(h(v), n_{B \cup \{u\}}(v)) - \min(h(v), n_B(v)) \right].$$

- For $u \notin B$ and $A \subset B$, we have

$$n_A(v) \leq n_B(v) \text{ and } n_{A \cup \{u\}}(v) \leq n_{B \cup \{u\}}(v).$$
(2)

Case a. \( n_{A\cup\{u\}}(v) \leq h(v) \). In this case,

\[
\delta_u f(A) = \min(h(v), n_{A\cup\{u\}}(v)) - \min(h(v), n_A(v)) \\
= n_{A\cup\{u\}}(v) - n_A(v) \\
= n_{B\cup\{u\}}(v) - n_B(v) \\
\geq \min(h(v), n_{B\cup\{u\}}(v)) - \min(h(v), n_B(v)) \\
= \delta_u f(B).
\]

Case b. \( n_{A\cup\{u\}}(v) \leq h(v) \). In this case,

\[
\delta_u f(A) = \min(h(v), n_{A\cup\{u\}}(v)) - \min(h(v), n_A(v)) \\
= 0 \\
= \min(h(v), n_{B\cup\{u\}}(v)) - \min(h(v), n_B(v)) \\
= \delta_u f(B).
\]
Theorem 3.4

The Greedy Algorithm for PIDS produces an approximation solution within a factor of $H(\Delta)$ from optimal where $\Delta$ is the maximum vertex degree of the input graph.

Note that $\gamma = \max_{v \in V} f(\{v\}) = \Delta$. 
Thank you.