On the Approximability of Influence in Social Networks

Yilin Shen

January 27, 2010

Contents

1. Problem Definition and Threshold Model

Contents

1. Problem Definition and Threshold Model
2. Inapproximability Results

Contents

1. Problem Definition and Threshold Model
2. Inapproximability Results
   1. Inapproximability Result on General Threshold Model
   2. Addition Inapproximability Result on Different Threshold Models
      1. Majority Thresholds
      2. Small Thresholds
      3. Unanimous Thresholds
      4. Tree Structure

Problem Definition and Threshold Model

Definition (Threshold Model)

Given a connected undirected graph $G = (V, E)$, let $d(v)$ be the degree of $v \in V$. For each $v \in V$, there is a threshold value $t(v) \in N$, where $1 \leq t(v) \leq d(v)$.

Problem Definition and Threshold Model

Definition (Threshold Model)

Given a connected undirected graph $G = (V, E)$, let $d(v)$ be the degree of $v \in V$. For each $v \in V$, there is a threshold value $t(v) \in N$, where $1 \leq t(v) \leq d(v)$.

Definition (Target Set Selection Problem)

Given a threshold model, initially the states of all vertices are inactive. The Target Set Selection problem is to pick the minimum subset of vertices, the target set, and set their state to be active. After that, in each discrete time step, the states of vertices are updated according to following rule: An inactive vertex v becomes active if at least $t(v)$ of its neighbors are active. The process runs until either all vertices are active or no additional vertices can update states from inactive to active.

Inapproximability Result on General Threshold Model

Theorem (2.1)

The Target Set Selection problem can not be approximated within the ratio of $O\left(2^{\log^{1-\epsilon} n}\right)$, *for any fixed constant* $\epsilon > 0$, *unless* $NP \subseteq DTIME(n^{poly \log(n)})$.

Inapproximability Result on General Threshold Model

Theorem (2.1)

The Target Set Selection problem can not be approximated within the ratio of $O\left(2^{\log^{1-\epsilon} n}\right)$, *for any fixed constant* $\epsilon > 0$, *unless* $NP \subseteq DTIME(n^{poly \log(n)})$.

Proof.

We will prove the theorem by a reduction from the Minimum Representative (MinRep) problem. □

Minimum Representative (MinRep) problem

Definition

Given a bipartite graph $G = (A, B; E)$, where A and B are disjoint sets of vertices, there are explicit partitions of A and B into equal-sized subsets. That is, $A = \cup_{i=1}^{\alpha} A_i$ and $B = \cup_{j=1}^{\beta} B_j$, where all sets A_i have the same size $|A|/\alpha$ and all sets B_j have the same size $|B|/\beta$. The partition of G induces a super-graph H as follows: There are $\alpha + \beta$ super-vertices, corresponding to each A_i and B_j respectively, and there is a super-edge between A_i and B_j if there exist some $a \in A_i$ and $b \in B_j$ that are adjacent in G.

Minimum Representative (MinRep) problem

Definition

Given a bipartite graph $G = (A, B; E)$, where A and B are disjoint sets of vertices, there are explicit partitions of A and B into equal-sized subsets. That is, $A = \cup_{i=1}^{\alpha} A_i$ and $B = \cup_{j=1}^{\beta} B_j$, where all sets A_i have the same size $|A|/\alpha$ and all sets B_j have the same size $|B|/\beta$. The partition of G induces a super-graph H as follows: There are $\alpha + \beta$ super-vertices, corresponding to each A_i and B_j respectively, and there is a super-edge between A_i and B_j if there exist some $a \in A_i$ and $b \in B_j$ that are adjacent in G.

The goal of the MinRep problem is to select the minimum number of representatives from each set A_i and B_j such that all super-edges are covered. That is, we wish to find subsets $A' \subseteq A$ and $B' \subseteq B$ with the minimum total size $A' + B'$ such that, for every super-edge (A_i, B_j), there exist representatives $a \in A' \cap A_i$ and $a \in B' \cap B_j$ that are adjacent in G.

Minimum Representative (MinRep) problem (Cont.)

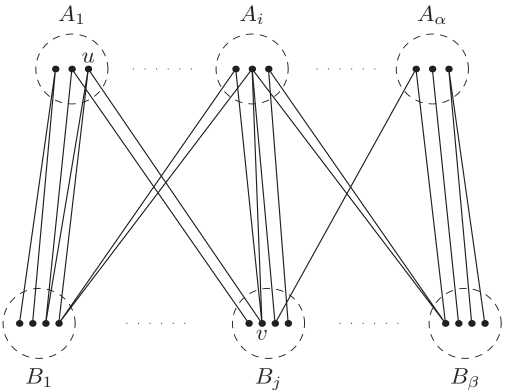


Figure: An instance of the MinRep problem

Minimum Representative (MinRep) problem (Cont.)

Theorem (2.2. R. Raz)

For any fixed $\epsilon > 0$, the MinRep problem can not be approximated within the ratio of $O\left(2^{\log^{1-\epsilon} n}\right)$, unless $NP \subseteq DTIME(n^{poly \log(n)})$.

Proof of Theorem 2.1

Definition (Basic Gadget Γ_l)

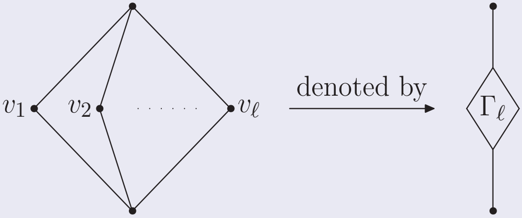


Figure: The basic gadget Γ_l

Proof of Theorem 2.1 (Cont.)

The Construction of Graph G' for the Target Set Selection Problem

For any given MinRep instance $G = (A, B; E)$, let M be the number of super-edges and N be the total input size.

Basically, G' consists of four different groups of vertices V_1, V_2, V_3, V_4, where the vertices between two groups are connected by the basic gadgets described above.

Proof of Theorem 2.1 (Cont.)

The Construction of Graph G' for the Target Set Selection Problem

For any given MinRep instance $G = (A, B; E)$, let M be the number of super-edges and N be the total input size.

Basically, G' consists of four different groups of vertices V_1, V_2, V_3, V_4, where the vertices between two groups are connected by the basic gadgets described above.

- $V_1 = \{a|a \in A\} \cup \{b|b \in B\}$ and each vertex has threshold N^2.

Proof of Theorem 2.1 (Cont.)

The Construction of Graph G' for the Target Set Selection Problem

For any given MinRep instance $G = (A, B; E)$, let M be the number of super-edges and N be the total input size.

Basically, G' consists of four different groups of vertices V_1, V_2, V_3, V_4, where the vertices between two groups are connected by the basic gadgets described above.

- $V_1 = \{a | a \in A\} \cup \{b | b \in B\}$ and each vertex has threshold N^2.
- $V_2 = \{u_{a,b} | (a, b) \in E\}$ and each vertex has threshold $2N^5$. Vertex $u_{a,b} \in V_2$ is connected to each of $a, b \in V_1$ by a basic gadget Γ_{N^5}.

Proof of Theorem 2.1 (Cont.)

The Construction of Graph G' for the Target Set Selection Problem (Cont.)

- $V_3 = \{v_{i,j} | A_i, B_j$ is connected by a super-edge$\}$ and each vertex has threshold N^4. Vertex $u_{a,b} \in V_2$ is connected to $v_{i,j} \in V_3$ by a basic gadget Γ_{N^4} if $a \in A_i$ and $b \in B_j$.

Proof of Theorem 2.1 (Cont.)

The Construction of Graph G' for the Target Set Selection Problem (Cont.)

- $V_3 = \{v_{i,j} | A_i, B_j \text{ is connected by a super-edge}\}$ and each vertex has threshold N^4. Vertex $u_{a,b} \in V_2$ is connected to $v_{i,j} \in V_3$ by a basic gadget Γ_{N^4} if $a \in A_i$ and $b \in B_j$.
- $V_4 = \{w_1, \ldots, w_N\}$ and each vertex has threshold $M \cdot N^2$. Each vertex $v_{i,j} \in V_3$ is connected to each $w_k \in V_4$ by a basic gadget Γ_{N^2}, and each vertex $a, b \in V_1$ is connected to each $w_k \in V_4$ by a basic gadget Γ_N.

Proof of Theorem 2.1 (Cont.)

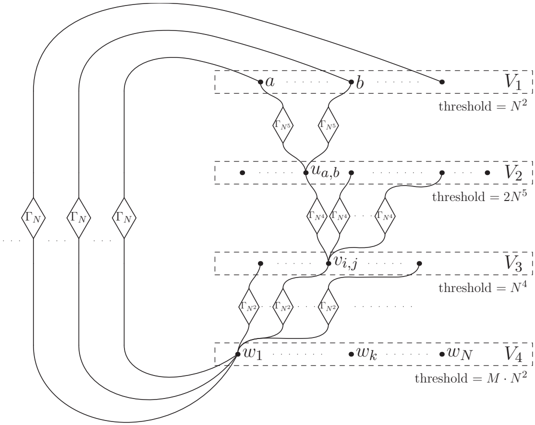


Figure: The structure of graph G'

Proof of Theorem 2.1 (Cont.)

Claim 2.3

The size of the optimal MinRep solution of G is within a factor of two of the size of the optimal Target Set Selection solution of G'.

Proof.

Proof of Theorem 2.1 (Cont.)

Claim 2.3

The size of the optimal MinRep solution of G is within a factor of two of the size of the optimal Target Set Selection solution of G'.

Proof.

($\Rightarrow$) Assume $A' \subseteq A$ and $B' \subseteq B$ is an optimal MinRep solution of G. We claim that $A' \cup B' \subseteq V_1$ is a Target Set Selection solution of G'.

Proof of Theorem 2.1 (Cont.)

Claim 2.3

The size of the optimal MinRep solution of G is within a factor of two of the size of the optimal Target Set Selection solution of G'.

Proof.

($\Rightarrow$) Assume $A' \subseteq A$ and $B' \subseteq B$ is an optimal MinRep solution of G. We claim that $A' \cup B' \subseteq V_1$ is a Target Set Selection solution of G'.

Since $A' \cup B'$ is a MinRep solution, for any super-edge (A_i, B_j), there exist $a \in A' \cap A_i$ and $b \in B' \cap B_j$ such that $(a, b) \in E$. Thus, vertex $u_{a,b} \in V_2$ can be active, which implies that $v_{i,j} \in V_3$ can be active as well. This is true for all super-edges, and thus all vertices in V_3 are active, which implies that all vertices in V_4 are active. Therefore, all vertices in V_1 can be active, which induces all vertices in G' to be active at the end. $\square$

Proof of Theorem 2.1 (Cont.)

Proof.

($\Leftarrow$) let S be an optimal Target Set Selection solution of G', we claim that the MinRep solution is at most $2|S|$.

Proof of Theorem 2.1 (Cont.)

Proof.

($\Leftarrow$) let S be an optimal Target Set Selection solution of G', we claim that the MinRep solution is at most $2|S|$.

- It is safe to assume that no middle vertices $v1, \ldots, v_l$ from any basic gadget Γ_l are in S.

Proof of Theorem 2.1 (Cont.)

Proof.

($\Leftarrow$) let S be an optimal Target Set Selection solution of G', we claim that the MinRep solution is at most $2|S|$.

- It is safe to assume that no middle vertices $v1, \ldots, v_l$ from any basic gadget Γ_l are in S.
- w.l.o.g Assume that no vertices in V_3 are in S because if a vertex $v_{i,j} \in S \cap V_3$, then we can remove $v_{i,j}$ from S and include $u_{a,b} \in V_2$ to S, where $a \in A_i$ and $b \in B_j$, which gives a solution of the same size.

Proof of Theorem 2.1 (Cont.)

Proof.

($\Leftarrow$) let S be an optimal Target Set Selection solution of G', we claim that the MinRep solution is at most $2|S|$.

- It is safe to assume that no middle vertices $v1, \dots, v_l$ from any basic gadget Γ_l are in S.
- w.l.o.g Assume that no vertices in V_3 are in S because if a vertex $v_{i,j} \in S \cap V_3$, then we can remove $v_{i,j}$ from S and include $u_{a,b} \in V_2$ to S, where $a \in A_i$ and $b \in B_j$, which gives a solution of the same size.
- If a vertex $u_{a,b} \in S \cap V_2$, we can remove $u_{a,b}$ from S and include $a, b \in V_1$ to S. By doing this, the size of S is increased by at most a factor of 2.

$\square$

Proof of Theorem 2.1 (Cont.)

Proof.

- Now $S \subseteq V_1 \cup V_4$. According to our construction, those vertices in $S \cap V_4$ can not affect any other vertices until all vertices in V_4 are active.

Proof of Theorem 2.1 (Cont.)

Proof.

- Now $S \subseteq V_1 \cup V_4$. According to our construction, those vertices in $S \cap V_4$ can not affect any other vertices until all vertices in V_4 are active.
- Therefore, the only direction for influence to flow in G' is through the channel $V_1 \rightarrow V_2 \rightarrow V_3 \rightarrow V_4$.

Proof of Theorem 2.1 (Cont.)

Proof.

- Now $S \subseteq V_1 \cup V_4$. According to our construction, those vertices in $S \cap V_4$ can not affect any other vertices until all vertices in V_4 are active.
- Therefore, the only direction for influence to flow in G' is through the channel $V_1 \rightarrow V_2 \rightarrow V_3 \rightarrow V_4$.
- To activate any vertex $w \in V_4 \setminus S$, all vertices in V_3 have to be activated. This implies that $S \cap V_1$ is a MinRep solution of G.

Proof of Theorem 2.1 (Cont.)

Proof.

- Now $S \subseteq V_1 \cup V_4$. According to our construction, those vertices in $S \cap V_4$ can not affect any other vertices until all vertices in V_4 are active.
- Therefore, the only direction for influence to flow in G' is through the channel $V_1 \rightarrow V_2 \rightarrow V_3 \rightarrow V_4$.
- To activate any vertex $w \in V_4 \setminus S$, all vertices in V_3 have to be activated. This implies that $S \cap V_1$ is a MinRep solution of G.

By Theorem 2.2, we have the same hardness of approximation result for the Target Set Selection problem, which completes the proof of Theorem 2.1. □

Extensions

We observe that Theorem 2.1 continues to hold for a few extensions:

Extensions

We observe that Theorem 2.1 continues to hold for a few extensions:

- The optimal solution influences each vertex in a constant number of rounds. This follows directly from the above construction.

Extensions

We observe that Theorem 2.1 continues to hold for a few extensions:

- The optimal solution influences each vertex in a constant number of rounds. This follows directly from the above construction.
- Instead of ensuring all vertices in the network are active, only a fixed fraction of vertices is needed to be activated. This can be done by the following simple reduction:

Extensions (Cont.)

The Construction of Graph G' for a Fixed Fraction of Vertices

- Replace each edge in E by a basic gadget Γ_n and define the new threshold of each $v \in V$ to be $t'(v) = n \cdot t(v)$.

Extensions (Cont.)

The Construction of Graph G' for a Fixed Fraction of Vertices

- Replace each edge in E by a basic gadget Γ_n and define the new threshold of each $v \in V$ to be $t'(v) = n \cdot t(v)$. (G and G' has the same optimal solution)

Extensions (Cont.)

The Construction of Graph G' for a Fixed Fraction of Vertices

- Replace each edge in E by a basic gadget Γ_n and define the new threshold of each $v \in V$ to be $t'(v) = n \cdot t(v)$. (G and G' has the same optimal solution)
- On graph G', by adding many dummy vertices (with thresholds being equal to their degrees) and connecting to all original vertices in V, it can be seen that to activate a fixed fraction of vertices, all vertices in V have to be activated.

Majority Thresholds

Definition (Majority Threshold Model)

In Threshold Model, for each $v \in V$, $t(v) = \lceil d(v)/2 \rceil$.

Majority Thresholds

Definition (Majority Threshold Model)

In Threshold Model, for each $v \in V$, $t(v) = \lceil d(v)/2 \rceil$.

Theorem (3.1)

Assume the Target Set Selection problem with arbitrary thresholds can not be approximated within the ratio of $f(n)$, for some polynomial time computable function $f(n)$. Then the problem with majority thresholds can not be approximated within the ratio of $O(f(n))$.

Proof of Theorem 3.1

Proof.

The basic idea is, for each $v \in V$ with $t(v) \neq \lceil d(v)/2 \rceil$, to add some dummy vertices incident to v (and change the threshold of v, if necessary) such that the threshold of v in the new setting is majority.

Proof of Theorem 3.1

Proof.

The basic idea is, for each $v \in V$ with $t(v) \neq \lceil d(v)/2 \rceil$, to add some dummy vertices incident to v (and change the threshold of v, if necessary) such that the threshold of v in the new setting is majority.

- $t(v) > \lceil d(v)/2 \rceil$. For this case, we add $2t(v) - d(v)$ isolated dummy vertices incident to v and with threshold 1 each.

Proof of Theorem 3.1

Proof.

The basic idea is, for each $v \in V$ with $t(v) \neq \lceil d(v)/2 \rceil$, to add some dummy vertices incident to v (and change the threshold of v, if necessary) such that the threshold of v in the new setting is majority.

- $t(v) > \lceil d(v)/2 \rceil$. For this case, we add $2t(v) - d(v)$ isolated dummy vertices incident to v and with threshold 1 each.
- $t(v) < \lceil d(v)/2 \rceil$. For this case, we add $d(v) - 2t(v)$ isolated dummy vertices incident to v and with threshold 1 each. Furthermore, let the new threshold of v be $d(v) - t(v)$.

Proof of Theorem 3.1

Proof.

The basic idea is, for each $v \in V$ with $t(v) \neq \lceil d(v)/2 \rceil$, to add some dummy vertices incident to v (and change the threshold of v, if necessary) such that the threshold of v in the new setting is majority.

- $t(v) > \lceil d(v)/2 \rceil$. For this case, we add $2t(v) - d(v)$ isolated dummy vertices incident to v and with threshold 1 each.
- $t(v) < \lceil d(v)/2 \rceil$. For this case, we add $d(v) - 2t(v)$ isolated dummy vertices incident to v and with threshold 1 each. Furthermore, let the new threshold of v be $d(v) - t(v)$.
- Add a "super" vertex u and connect u to all dummy vertices added in the above $t(v) < \lceil d(v)/2 \rceil$.

$\square$

Proof of Theorem 3.1 (Cont.)

Proof.

Claim 3.2

The size difference between the optimal Target Set Selection solution of G' and G is at most 1.

□

Small Thresholds

Definition (Small Threshold Model)

In Threshold Model, for each $v \in V$, $t(v) \leq 2$.

Small Thresholds

Definition (Small Threshold Model)

In Threshold Model, for each $v \in V$, $t(v) \leq 2$.

Notice that $t(v) = 1$ for any $v \in V$, the problem can be solved trivially: For each connected component of the graph, we target a vertex arbitrarily.

Small Thresholds

Definition (Small Threshold Model)

In Threshold Model, for each $v \in V$, $t(v) \leq 2$.

Notice that $t(v) = 1$ for any $v \in V$, the problem can be solved trivially: For each connected component of the graph, we target a vertex arbitrarily.

Theorem (4.1)

Assume the Target Set Selection problem with arbitrary thresholds can not be approximated within the ratio of $f(n)$, for some polynomial time computable function $f(n)$. Then the problem can not be approximated within the ratio of $O(f(n))$ when all thresholds are at most 2.

Small Thresholds (Cont.)

Corollary (4.1)

Given any graph where $t(v) = 2$ (or $t(v) \leq 2$) for any vertex v, the Target Set Selection problem can not be approximated within the ratio of $O\left(2^{\log^{1-\epsilon} n}\right)$, for any fixed constant $\epsilon > 0$, unless $NP \subseteq DTIME(n^{poly \log(n)})$.

Proof.

Small Thresholds (Cont.)

Corollary (4.1)

Given any graph where $t(v) = 2$ (or $t(v) \leq 2$) for any vertex v, the Target Set Selection problem can not be approximated within the ratio of $O\left(2^{\log^{1-\epsilon} n}\right)$, for any fixed constant $\epsilon > 0$, unless $NP \subseteq DTIME(n^{poly \log(n)})$.

Proof.

We will prove the $t(v) = 2$ case by a reduction from the $t(v) \leq 2$ case.

Small Thresholds (Cont.)

Corollary (4.1)

Given any graph where $t(v) = 2$ (or $t(v) \leq 2$) for any vertex v, the Target Set Selection problem can not be approximated within the ratio of $O\left(2^{\log^{1-\epsilon} n}\right)$, for any fixed constant $\epsilon > 0$, unless $NP \subseteq DTIME(n^{poly \log(n)})$.

Proof.

We will prove the $t(v) = 2$ case by a reduction from the $t(v) \leq 2$ case.

Given a graph $G = (V, E)$ where $t(v) \leq 2$ for any $v \in V$, we add a “super” vertex u and connect u to each $v \in V$ with $t(v) = 1$. Let the resulting graph be G' and all thresholds in G' be 2.

Small Thresholds (Cont.)

Corollary (4.1)

Given any graph where $t(v) = 2$ (or $t(v) \leq 2$) for any vertex v, the Target Set Selection problem can not be approximated within the ratio of $O\left(2^{\log^{1-\epsilon} n}\right)$, for any fixed constant $\epsilon > 0$, unless $NP \subseteq DTIME(n^{poly \log(n)})$.

Proof.

We will prove the $t(v) = 2$ case by a reduction from the $t(v) \leq 2$ case.

Given a graph $G = (V, E)$ where $t(v) \leq 2$ for any $v \in V$, we add a "super" vertex u and connect u to each $v \in V$ with $t(v) = 1$.

Let the resulting graph be G' and all thresholds in G' be 2.

The size difference between the optimal solution of G' and G is at most 1. $\square$

Simulating Majority Boolean Circuit

Majority Function f

A boolean function $f\colon \{0,1\}^n \to \{0,1\}$ is called a majority function if

$$f(x_1,\ldots,x_n) = \begin{cases} 1 & \text{if } x_1+\cdots+x_n \geq \lceil n/2 \rceil \\ 0 & \text{otherwise} \end{cases}$$

Simulating Majority Boolean Circuit

Majority Function f

A boolean function $f\colon \{0,1\}^n \rightarrow \{0,1\}$ is called a majority function if

$$f(x_1,\ldots,x_n) = \begin{cases} 1 & \text{if } x_1 + \cdots + x_n \geq \lceil n/2 \rceil \\ 0 & \text{otherwise} \end{cases}$$

Theorem (4.2 Ajtai, Komlós and Szemerédi)

There exist polynomial size monotone circuits to compute majority boolean functions, where monotone means only AND and OR gates are in the circuit.

Simulating Majority Boolean Circuit

Majority Function f

A boolean function $f\colon \{0,1\}^n \to \{0,1\}$ is called a majority function if

$$f(x_1,\dots,x_n) = \begin{cases} 1 & \text{if } x_1+\cdots+x_n \geq \lceil n/2 \rceil \\ 0 & \text{otherwise} \end{cases}$$

Theorem (4.2 Ajtai, Komlós and Szemerédi)

There exist polynomial size monotone circuits to compute majority boolean functions, where monotone means only AND and OR gates are in the circuit.

The basic idea is to construct small gadgets composed of vertices of thresholds at most 2 to simulate AND and OR gates in a circuit.

Gadget for AND gate

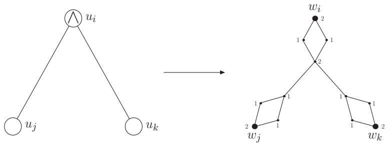


Figure: Gadget for AND gate (The value on each vertex is its threshold)

Gadget for AND gate

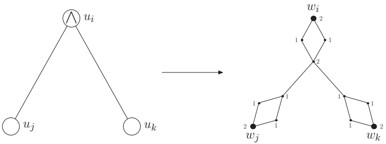


Figure: Gadget for AND gate (The value on each vertex is its threshold)

- "bottom-to-top": w_i is active (corresponding to the output u_i being 1) only if both w_j and w_k are active (corresponding to the inputs u_j and u_k being 1). In addition, if only one of w_j and w_k is active (say w_j), the center vertex of threshold 2 ensures that neither w_i nor w_k can get active due to influence from w_j.

Gadget for AND gate

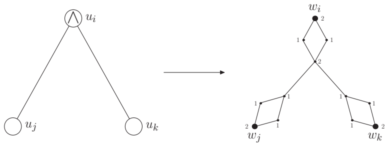


Figure: Gadget for AND gate (The value on each vertex is its threshold)

- "bottom-to-top": w_i is active (corresponding to the output u_i being 1) only if both w_j and w_k are active (corresponding to the inputs u_j and u_k being 1). In addition, if only one of w_j and w_k is active (say w_j), the center vertex of threshold 2 ensures that neither w_i nor w_k can get active due to influence from w_j.
- "top-to-bottom": Once w_i is active, both w_j and w_k become active as well.

Gadget for OR gate

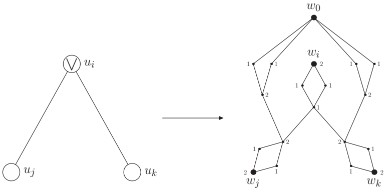


Figure: Gadget for OR gate (The value on each vertex is its threshold), w_0 is the vertex corresponding to the final output u_0 of the circuit

Gadget for OR gate

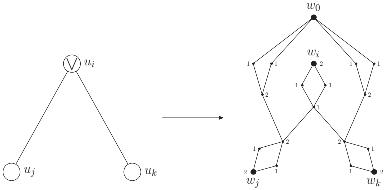


Figure: Gadget for OR gate (The value on each vertex is its threshold), w_0 is the vertex corresponding to the final output u_0 of the circuit

- "bottom-to-top": w_i is active (corresponding to the output u_i being 1) if at least one of w_j and w_k is active (corresponding to at least one of the inputs u_j and u_k being 1). In addition, if only one of w_j and w_k is active (say w_j), even though w_i can be activated, neither w_0 nor w_k can get active due to

Gadget for OR gate

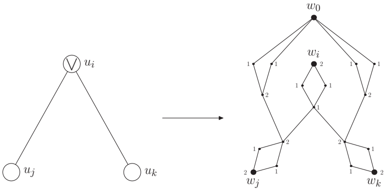


Figure: Gadget for OR gate (The value on each vertex is its threshold), w_0 is the vertex corresponding to the final output u_0 of the circuit

- "bottom-to-top": w_i is active (corresponding to the output u_i being 1) if at least one of w_j and w_k is active (corresponding to at least one of the inputs u_j and u_k being 1). In addition, if only one of w_j and w_k is active (say w_j), even though w_i can be activated, neither w_0 nor w_k can get active due to influence from w_i, w_j

The Properties of G_n

Let G_n correspond to the majority boolean function $f(x_1, \ldots, x_n)$. G_n has the following properties:

The Properties of G_n

Let G_n correspond to the majority boolean function $f(x_1, \ldots, x_n)$. G_n has the following properties:

- If w_0 is active, then all vertices in G_n can become active, donated as $r(G_n)$.

The Properties of G_n

Let G_n correspond to the majority boolean function $f(x_1, \ldots, x_n)$. G_n has the following properties:

- If w_0 is active, then all vertices in G_n can become active, donated as $r(G_n)$.
- If at least half of vertices in $\{w_1, \ldots, w_n\}$ are active, then w_0 can be active.

The Properties of G_n

Let G_n correspond to the majority boolean function $f(x_1, \ldots, x_n)$. G_n has the following properties:

- If w_0 is active, then all vertices in G_n can become active, donated as $r(G_n)$.
- If at least half of vertices in $\{w_1, \ldots, w_n\}$ are active, then w_0 can be active.
- If a vertex w_i is inactive, then all its neighbors are still inactive. In particular, this implies that if less than half of the input vertices are active, then the remaining inactive input vertices can not be activated due to influence from G_n.

Proof of Theorem 4.1

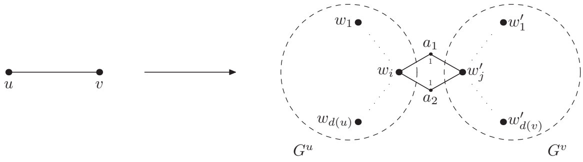


Figure: Gadget for edge (u, v)

Proof of Theorem 4.1 (Cont.)

For each $v \in V$, let $d(v)$ be the degree of v in G. We use a copy G_v of graph $G_{d(v)}$ to replace v and all its incident edges, where $G_{d(v)}$ is the graph constructed above to simulate majority function $f(\cdot)$ with $d(v)$ input variables. Each input vertex in G^v corresponds to an edge incident to v in E. For any edge $(u, v) \in E$, let w_i and w'_j be the two input vertices in G^u and G^v corresponding to (u, v), respectively. We connect w_i and w'_j by a basic gadget Γ_2 (i.e. we add two vertices a_1 and a_2 with threshold 1 each and connect (a_1, w_i), (a_1, w'_j), (a_2, w_i), (a_2, w'_j)). Denote the resulting graph by G'.

Proof of Theorem 4.1 (Cont.)

Claim 4.2

The size of the optimal Target Set Selection solution of G is equal to that of G'.

Proof of Theorem 4.1 (Cont.)

Proof.

($\Rightarrow$) For any Target Set Selection solution S of G, let $S' = \{r(G^v) | v \in S\}$, i.e. S' contains the output vertex of each G^v for $v \in S$. For any $v \in S$, we consider how its neighbor u could be influenced by v. In graph G, we know u can be influenced from v directly by one unit. In graph G', according to the properties of G^v established in the last subsection, we know all vertices in G^v are active. Thus, as u and v are connected by an edge, one of the input vertices of G^u becomes active. Since the threshold of u in G is majority, u becomes active when at least half of its neighbors are active, which is equivalent to at least half of the input vertices of G^u being active (and thus, all vertices in G^u are active). Hence, the influence propagation in G' follows exactly the same pattern as that in G, and hence S' is a Target Set Selection solution of G'. $\square$

Proof of Theorem 4.1 (Cont.)

Proof.

($\Leftarrow$) let S' be an optimal Target Set Selection solution of G'. According to the properties of simulation graph discussed above, we can assume without loss of generality that only output vertices are in S'. Define $S = \{v \in V | r(G^v) \in S'\}$. By a similar argument as above, it follows that S is a Target Set Selection solution of G. $\square$

Unanimous Thresholds

Definition (Unanimous Threshold Model)

In Threshold Model, $t(v) = d(v)$ for each $v \in V$.

Unanimous Thresholds

Definition (Unanimous Threshold Model)

In Threshold Model, $t(v) = d(v)$ for each $v \in V$.

Theorem (5.1)

If all thresholds in a graph are unanimous, it is NP-hard to compute the optimal Target Set Selection solution.

Proof of Theorem 5.1

Proof.

G has a vertex cover of size at most k if and only if G has Target Set Selection has a solution of size at most k.

Proof of Theorem 5.1

Proof.

G has a vertex cover of size at most k if and only if G has Target Set Selection has a solution of size at most k.

($\Rightarrow$) For any vertex cover solution V' of G, let the target set of G be V'. Then for each $v \notin V'$, all edges incident to v are covered by the corresponding vertices in V', which implies v can be active. Thus, by targeting V', all vertices are active at the end.

Proof of Theorem 5.1

Proof.

G has a vertex cover of size at most k if and only if G has Target Set Selection has a solution of size at most k.

($\Rightarrow$) For any vertex cover solution V' of G, let the target set of G be V'. Then for each $v \notin V'$, all edges incident to v are covered by the corresponding vertices in V', which implies v can be active. Thus, by targeting V', all vertices are active at the end.

($\Leftarrow$) For any Target Set Selection solution V', we argue that V' is a vertex cover as well. For any edge (u, v), if neither u nor v is in V', both u and v can not be activated, since their threshold is equal to their degree, which is a contradiction. □

Tree Structure

When the underlying graph $G = (V, E)$ is a tree, the Target Set Selection problem can be solved in polynomial time. The basic observation is that for any leaf $v \in V$, $t(v)$ is equal to 1. Thus, at most one of v and its parent u will be targeted in the optimal solution. Hence, we can assume without loss of generality that v is not targeted, otherwise, we can target u instead of v and get a solution of the same size. The algorithm is as follows:

ALG-TREE

ALG-TREE

1. Let $t'(v) = t(v)$, for $v \in V$
2. Let $x(v) = 0$, for each leaf $v \in V$
3. While there is $x(v)$ not defined yet
4.   for any vertex u where all $x(\cdot)$'s of its children have been defined
5.   let w be u's parent
6.   if $t'(u) \geq 2$
7.   let $x(u) = 1$
8.   let $t'(w) \leftarrow t'(w) - 1$
9.   else
10.   let $x(u) = 0$
11.   if $t'(u) \leq 0$
12.   let $t'(w) \leftarrow t'(w) - 1$
13. Output the target set $\{v \in V \mid x(v) = 1\}$

Figure: ALG-TREE

ALG-TREE

ALG-TREE

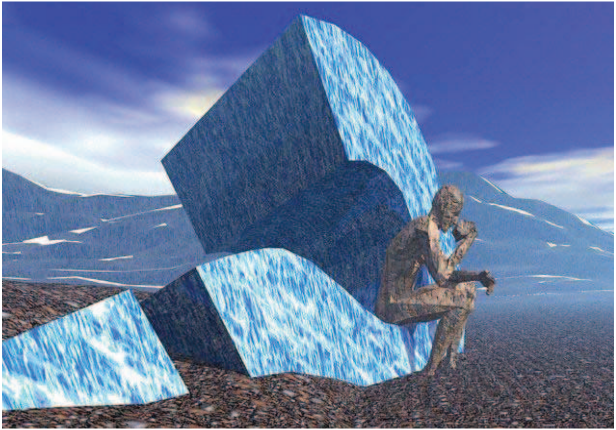
```
1. Let t'(v) = t(v), for v ∈ V
2. Let x(v) = 0, for each leaf v ∈ V
3. While there is x(v) not defined yet
4.    for any vertex u where all x(·)'s of its
      children have been defined
5.       let w be u's parent
6.       if t'(u) ≥ 2
7.          let x(u) = 1
8.          let t'(w) ← t'(w) − 1
9.       else
10.         let x(u) = 0
11.         if t'(u) ≤ 0
12.            let t'(w) ← t'(w) − 1
13. Output the target set {v ∈ V | x(v) = 1}
```

Figure: ALG-TREE

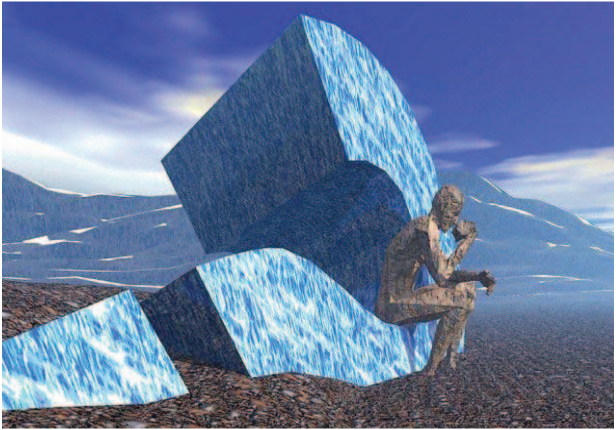
Theorem (6.1)

Alg-Tree computes an optimal solution for the Target Set Selection problem when the underlying graph $G = (V, E)$ is a tree.

Questions?



Questions?



Thank you !