The Containment Problem

Michael Capalbo

mcapalbo@dimacs.rutgers.edu

(Joint work with James Abello)

Let $G = (V, E)$ be a network (e.g., social, computer network, ect.), and let S_0 be any subset of V.

$\triangleright$ Every node in S_0 is infected with a virus that spreads from each infected node to all of its nonvaccinated neighbors in one time-step.

$\triangleright$ Our allowed response: vaccinate a limited number (about a_l) of nodes during each time step $l = 1, 2, 3, \ldots, t$

$\triangleright$ Our goal: find what nodes to vaccinate each step to minimize the total number m of nodes that eventually become infected.

We call this the $\mathrm{Containment\ Problem}$ (input is G, S_0, and the a_l's).

Why would we care about solving the Containment Problem?

$\triangleright$ Limited supply of vaccine available initially to stop an infection.

$\triangleright$ Containment of computer virus spreading through a network.

$\triangleright$ Blocking off suspects from escaping the scene of a crime where only a limited number of policeman are available initially.

Note that it is always at least as effective to vaccinate a node earlier rather than later.

Lemma 1 Let C be the set of nodes that we vaccinate at some point or another. If each node v in C is vaccinated before the infection reaches it, then the number of nodes that eventually become infected is the number of nodes that share a component in $G \setminus C$ with a node in S_0.

Unfortunately, the Containment Problem (CP) is NP-hard: It is probably impossible to devise a tractable algorithm that returns an optimum strategy.

So we devise a tractable approximation algorithm for CP that returns a (slightly) inferior vaccination strategy:

- ▷ have to vaccinate slightly more ($O(\log |V(G)|) \times a_l$ nodes instead of only a_l nodes as before) at each time-step l, and
- ▷ the total number of nodes that become infected is no more than $3m$ (instead of the minimum m as before).

And we also require that the network G satisfy the following expansion property: Every subset S of no more than a quarter of the nodes of G has at least $|S|$ neighbors outside S.

▷ Empirical evidence: social networks resemble random graphs

▷ Random graphs have expansion property

▷ So our assumption about the expansion properties of G is reasonable.

Overview of the approximation algorithm for the CP

Lemma 2 If G is an expander, then it is possible to vaccinate only twice as many nodes per time-step as before for the first $t = \log |V(G)|$ time-steps, and then none after, so that no more than nodes than before become infected.

▷ Formulate CP as an integer program (IP), where we vaccinate $2a_l$ nodes per time-step $l \leq t$ (and none thereafter) instead of only a_l.

▷ State and solve an appropriate linear relaxation (LP) of (IP).

▷ Use combinatorial techniques (and that t in Lemma 2 is small) to convert the solution for (LP) into a vaccination strategy.

- $\triangleright$ Formulate the CP as an integer program (IP). $\leftarrow$
- $\triangleright$ State and solve an appropriate linear relaxation (LP) of (IP).
- $\triangleright$ Use combinatorial techniques...

(IP) Minimize

$$\sum_{v \in V} \sum_{i=0}^{|V|} |x_{v,i}|$$

subject to

$\triangleright$ $\sum_{i=1}^{l+1} y_{v,i} + \sum_{i=1}^{l+1} x_{v,i} \geq \sum_{i=1}^{l} x_{u,i}$, $\forall v \in V$, $\forall \{u,v\} \in E$, and $\forall j = 1, 2, \ldots$

$\triangleright$ $\sum_{v \in V} y_{v,l} \leq 2a_l$, $\forall l = 1, 2, \ldots, t$, and $y_{v,l} = 0$ for all $l > t$.

$\triangleright$ $x_{s,0} = 1$, $\forall s \in S_0$, and

$\triangleright$ $x_{v,i}, y_{v,i} \in \{0,1\}$, $\forall v \in V$; $\forall i = 1, 2, \ldots, |V|$.

▷ Formulate the CP as an integer program (IP). $\surd$

▷ State and solve an appropriate linear relaxation (LP) of (IP). $\leftarrow$

▷ Use combinatorial techniques...

(LP) Minimize

$$\sum_{v\in V}\sum_{i=0}^{|V|}|x_{v,i}|$$

subject to

▷ $\sum_{i=1}^{l+1} y_{v,i} + \sum_{i=1}^{l+1} x_{v,i} \geq \sum_{i=1}^{l} x_{u,i}$, $\forall v \in V$, $\forall\{u,v\} \in E$, and $\forall l = 1, 2, \ldots$

▷ $\sum_{v\in V} y_{v,l} \leq 2a_l$, $\forall l = 1, 2, \ldots, t$, and $y_{v,l} = 0$ for all $l > t$.

▷ $x_{s,0} = 1$, $\forall s \in S_0$, and

▷ $0 \leq x_{v,i}, y_{v,i} \leq 1$, $\forall v \in V$; $\forall i = 1, 2, \ldots, |V|$.

▷ Formulate the CP as an integer program (IP). $\surd$

▷ State and solve an appropriate linear relaxation (LP) of (IP). $\surd$

▷ Use combinatorial techniques (and the fact that t (number of time-steps) is small) to convert the solution of (LP) into a vaccination strategy$\leftarrow$

Sol'n of LP $\rightarrow$ vaccination strategy

▷ Let $\{x_{v,i}, y_{v,i} |\ v \in V; i = 0, 1, \ldots, |V|\}$ be the sol'n to (LP).

▷ Set S to be the nodes v s.t. $\sum_i x_{v,i} \geq \frac{2}{3}$, and T to be the nodes v s.t. $\sum_i x_{v,i} < \frac{1}{3}$, and let C be a min-cut in G between S and T.

▷ Let C_1 be the vertices v in S such that $y_{v,1} \geq \frac{1}{3t}$. For general $l < t$, set C_{l+1} to be the vertices v in S of distance at least $l+1$ from a vertex in S_0 in $G \setminus (C_1 \cup \ldots \cup C_l)$, s.t. $\sum_{i=1}^{l} y_{v,i} \geq \frac{1}{3t}$.

▷ For each $l < t$, vaccinate the vertices in C_l at time step l, and the vertices in $C_t \cup C$ at time step t, and return.

Further research directions

- ▷ Remove the condition that G is an expander.
- ▷ Improve the approximation factors of the algorithm.
- ▷ Establish stronger hardness results.