Approximability of the Firefighter Problem: Computing Cuts over Time[*]

Elliot Anshelevich[†] Deeparnab Chakrabarty[‡] Ameya Hate[†] Chaitanya Swamy[‡§]

Abstract

We provide approximation algorithms for several variants of the FIREFIGHTER problem on general graphs. The Firefighter problem models the case where a diffusive process such as an infection (or an idea, a computer virus, a fire) is spreading through a network, and our goal is to contain this infection by using targeted vaccinations. Specifically, we are allowed to vaccinate at most a fixed number (called the budget) of nodes per time-step, with the goal of minimizing the effect of the infection. The difficulty of this problem comes from its temporal component, since we must choose nodes to vaccinate at every time-step while the infection is spreading through the network, leading to notions of "cuts over time".

We consider two versions of the Firefighter problem: a "non-spreading" model, where vaccinating a node means only that this node cannot be infected; and a "spreading" model where the vaccination itself is an infectious process, such as in the case where the infection is a harmful idea, and the vaccine to it is another infectious beneficial idea. We look at two measures: the MAXSAVE measure in which we want to maximize the number of nodes which are not infected given a fixed budget, and the MINBUDGET measure, in which we are given a set of nodes which we have to save and the goal is to minimize the budget. We study the approximability of these problems in both models.

1 Introduction

Faced with an epidemic that is spreading through a population, and a limited supply of vaccine (or simply a lack of time to administer it), it is necessary to decide whom to vaccinate. Questions about the spread of disease and epidemics in a social network have often been modeled using graph theory (e.g. [2, 12]), and correspond to fundamental graph-theoretic concepts [24]. Moreover, these graph theoretic principles can be applied to many diffusive network processes, including epidemics in computer networks, the spread of innovations and ideas, and viral marketing [25]. In this paper, we focus specifically on inhibiting the spread of an epidemic or an idea by using vaccination.

Questions about epidemic propagation have been studied in several fields (e.g., [4, 33]), although most of this research does not consider the structure of the corresponding network (partly because this structure is only recently becoming available). Instead, building on the work of [24] and others [2, 13], this paper assumes that we know the network's topology, and provides worst-case guarantees over all possible networks. Earlier works consider *prophylactic vaccinations,* where the goal is to vaccinate parts of the graph so that once the epidemic begins, the destruction caused by it will be limited. The problem we consider, however, considers the case where the infection has already begun, and we must attempt to minimize its effect.

[*] A preliminary version of this paper appeared in ISAAC 2009.

[†] Dept. of Computer Science, Rensselaer Polytechnic Institute, Troy, NY. Supported in part by NSF CCF-0914782.

[‡] deepc@uwaterloo.ca, cswamy@math.uwaterloo.ca. Dept. of Combinatorics & Optimization, University of Waterloo, Waterloo, ON N2L 3G1, CANADA.

[§] Supported in part by NSERC grant 327620-09 and an Ontario Early Researcher Award.

The model and the Firefighter problem. We model our network of agents as a graph $G = (V, E)$ where vertices correspond to agents, and an edge $e = (u, v)$ represents contact between the two agents. The above is arguably a simplistic model, however, as we see below, even this model leads to many interesting questions and will be the main focus of our paper.

Such a model of spread of infection has been studied in the literature as the *Firefighter problem* [17, 22]. In this problem the infection/fire starts at a given node node s (or a set of nodes) at time $\tau = 0$. At every subsequent time step, the infection/fire deterministically spreads to all nodes that have an already infected neighbor. To stop the infection, we are allowed to vaccinate/defend at most B nodes per time-step, where B is a budget representing how much we are able to affect the network in a single time-step. A vaccinated node can no longer contract the infection, and therefore cannot pass it on to others. Once infected or vaccinated the vertex remains so for the rest of the time. The process comes to an end when the infection can no longer spread.

We consider two separate objectives in this paper. The first objective, which we call MaxSave, is to maximize the number of non-infected nodes in the end, when we are given a fixed budget B. The second objective, which we call MinBudget, is to minimize the budget B needed per time instant in order to save a given set of nodes, $T \subseteq V$.

It is not hard to see that in the end, the set of vaccinated notes form a vertex cut between the set of infected nodes and set of non-infected nodes. However, unlike previous works, such as [24], which examine the *static* problem of vaccinating a 'cut' before the infection has started spreading, we need to find the "best" *cut over time* (where best depends on the considered objective). This temporal nature of our optimization problem makes it significantly different and more challenging than the non-temporal versions.

In this paper, we also consider a different variant of this model, where the vaccination is also a process that spreads through the network. In this "spreading vaccination" model, if a node v is adjacent to a vaccinated node, then v itself becomes vaccinated during the next time-step (unless it is already infected). In the case of ideas propagating through a social network, this represents the fact that an antidote to a harmful idea is often another idea, which can be just as infectious. In disease propagation, this represents the fact that vaccines can be infectious as well, since they are often an attenuated version of the actual disease. We consider the above two objectives in this model as well.

Example 1. To gain some intuition about this problem, consider the example shown in Figure 1 using the non-spreading model of vaccination.

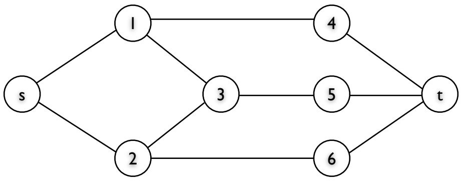

Figure 1: This example shows that sometimes vaccinating nodes far away from the infection is the only way to save all the required nodes.

Consider the MinBudget objective for this example. The infection begins at node s, and the goal is to find the smallest number B of nodes that need to be vaccinated at every time step so that we can save the node t, which we assume cannot itself be vaccinated. If we were only allowed to cut nodes during the first time-step, this would be equivalent to the minimum s-t node-cut problem.

The temporal nature of the problem, however, complicates matters: intuitively, the tradeoff is between vaccinating a small set of nodes close to the infection source early, or spreading out (over time) the vaccination of a larger set of nodes which are farther away from the source.

For instance, in the above example, a minimum s-t node-cut is $\{1, 2\}$, which requires $B = 2$. However, there *is* a solution to the above problem with $B = 1$, but the final set of vaccinated nodes does not form a minimum s-t node-cut. One such solution is to vaccinate vertices 4, 6, and 5 at time steps 1, 2, and 3 respectively, leading to the final set of vaccinated nodes being $\{4, 5, 6\}$ which is not a minimum cut. In fact, it is not hard to come up with examples where the optimal value of B is much smaller than the size of a minimum node s-t cut *and* the final set of vaccinated nodes is much larger than the size of a minimum node s-t cut (e.g., take a graph where s has k neighbours, each of which is connected to t via k long internally node-disjoint paths). Thus, this "cuts over time" problem is quite different from the classical min-cut problem, and in fact is known to be NP-hard (even when the graph is a tree!) [16].

Our Results. In Section 3, we consider the model of spreading vaccinations. In general, our results show that this model is more tractable than the model with non-spreading vaccinations. For MaxSave we show that this problem reduces to maximizing a submodular function with a matroid constraint. Therefore a simple greedy algorithm provides a 2-approximation, and a recent result of [7] lets us prove a $e/(e-1)$ factor approximation. Using this, for MinBudget we give a $\ln n$ approximation algorithm, and show that both of our approximation ratios are tight, by showing a set-cover hardness.

The non-spreading model, on the other hand, does not yield itself to good approximation algorithms. In fact, we show in Section 4 that it is NP-hard to approximate MaxSave in general graphs to a factor of $n^{(1-\epsilon)}$, for any $\epsilon > 0$. On the other hand, for MinBudget, we give a simple $O(\sqrt{n})$ factor approximation algorithm. For the special class of directed layered networks, we get an improved approximation factor of $H(\ell) = 1 + 1/2 + \cdots + 1/\ell$, where ℓ is the number of layers in the network. In fact, our result shows that the integrality gap of a natural *linear-programming* (LP) relaxation of the problem is $O\big(H(\ell)\big)$. Complementing this, we give an example of a directed ℓ-layered graph showing that the integrality gap is of our LP is also $\Omega(\log n)$. Both our algorithms are combinatorial, and the LP is used only in the analysis. We remark here that the hardness of MinBudget in the spreading model does not carry over to the non-spreading model and as of yet we cannot rule out a constant factor approximation.

	Spreading	Non-Spreading
Max-Save	$(1-1/e)$ appx	$n^{(1-\epsilon)}$-hard for $\epsilon > 0$
Min-Budget	$\ln n$ appx $(1-o(1))\ln n$-hard	General: $2\sqrt{n}$-appx Directed ℓ-Layered Graph: $H(\ell)$-appx

Table 1: The summary of our approximation results. appx stands for approximation, hard stands for hard to approximate under complexity assumptions.

Related Work. Recently, a few groups [6, 20, 31, 34] have considered the spread of viruses or ideas on Internet-like topologies, such as small-world networks [37] and preferential attachment models [5, 27]. Several papers also study targeted vaccinations in this context [11, 14], and show that they can be used to significantly reduce the effect of epidemics. These studies assume certain properties of the networks (based on where these networks arise from).

Several recent papers considered modeling vaccination by using graph cuts. For example, the work of Hayrapetyan et al. [24] and others [2, 13] fully utilizes the social-network structure to "cut off" and contain various diffusive processes in a social network.

As mentioned earlier, all this work is only concerned with vaccinating a set of nodes before the infection begins, however, and does not have the temporal component of the Firefighter problem. A lot more work has been done on maximizing the spread of an infection (instead of trying to stop it using vaccinations), by selecting the best nodes to infect initially [12, 25].

The *Firefighter problem* was first introduced by B. Hartnell [22], and there has been much work on this problem; see, e.g., [17] for a survey. However, much of the work has focused on special graph structures, such as grids [10, 19, 36], and that too usually with the MaxSave objective. The Firefighter problem is NP-complete even when the underlying graph is a tree [16], although [23] and [28] give approximation algorithms for this case, and [30] shows how to solve the problem in in polynomial time for special cases of trees.

The Firefighter problem with the MinBudget objective, has some structural similarity with the problem of length-bounded cuts [3], where the goal is to form a minimum s-t cut that destroys all paths of length at most L. However, as illustrated by the example above, the main difference here is that our paper deals with *dynamic cuts* i.e., cuts which arise over a period of time, while graph cuts are static in nature.

Very recently, and independent of our work, Chalermsook and Chuzhoy [8] obtained a $O(\log^* n)$ approximation for the MinBudget problem in the non-spreading model when the graph is a tree. They also obtained a $O(\log \ell)$ approximation for directed layered networks with ℓ layers.

2 Formal description of our model

We are given a directed[1] graph $G = (V, E)$ and a source node s. We assume $|V| = n$ and $|E| = m$. We let the distance between two nodes u and v be the path from u to v of minimum length.

All nodes in the graph can have one of three states: they can be *infected*, *vaccinated*, or *vulnerable*, that is neither vaccinated nor infected. At time $\tau = 0$, all nodes are vulnerable, except node s, which is infected. At each $\tau > 0$, any vulnerable vertex v which is connected to an infected node u, such that $(u, v) \in E$, gets infected at time $\tau + 1$, unless it is vaccinated at time step τ. Infected and vaccinated nodes stay infected and vaccinated respectively. We call a node *saved* if it is either vaccinated or if all paths from any infected node to it contains at least one vaccinated node.

Definition 2.1. A vaccination strategy is a set $\Psi \subseteq V \times J$ where V is the set of vertices of graph G and $J = \{1, 2, \ldots |V|\}$. The vertex v is vaccinated at time $\tau \in J$ by the vaccination strategy Ψ if $(v, \tau) \in \Psi$. A vaccination strategy Ψ is valid with respect to budget B, if the following two conditions are satisfied:

i. if $(v, \tau) \in \Psi$ then v is not infected at time τ,

ii. let $\Psi_\tau = \{(v, \tau) \in \Psi\}$; then $|\Psi_\tau| \leq B \quad \text{for} \quad \tau = 1 \ldots |V|$.

The first condition implies we can only vaccinate vulnerable nodes, and the second condition requires us to obey the budget constraint.

We consider two models of vaccination. In the *Non-spreading Vaccination Model*, vaccinating a vertex simply means that it can no longer be infected. Vaccination of a node doesn't affect the vaccination of other neighboring nodes.

[1] We use a directed graph since it is more general – an undirected graph is just a directed graph with two arcs per edge.

In the *Spreading Vaccination Model*, however, the vaccination spreads to all its neighboring nodes which are still vulnerable, thereby vaccinating them. That is, at time step $\tau > 0$, if a node v is vaccinated and there is a vulnerable node u such that $(v, u) \in E$, then at time $\tau + 1$, the node v also gets vaccinated. Thus, the vaccination also spreads like the infection. Note that it could be a vulnerable node is adjacent to both an infected node and a vaccinated node. We will assume that the vaccine prevails over the infection, and in the subsequent time step, the vulnerable node is vaccinated, rather than being infected. This is actually a weak assumption as assuming otherwise doesn't change the quality of our results. In the spreading model, we will say that a node is vaccinated *directly* when it is vaccinated by the vaccination strategy, and it is vaccinated *indirectly* when it is vaccinated by the spread of the vaccine through the network.

The process stops when there are no vulnerable nodes adjacent to an infected node, so the infection cannot spread any further. This must occur before time n, for n being the number of nodes.

Objectives. The main two objectives we consider when developing a vaccination strategy are as follows.

MaxSave(G, B, s, T)
Instance: A rooted graph $(G(V, E), s)$, integer $B \geq 1$ and $T \subseteq V$
Objective: Find a valid vaccination strategy Ψ such that if s is the only infected node at time 0, then at the end of the above process the number of non-infected nodes that belong to T is maximized.

This problem is also referred to as the Firefighter problem in the literature when $T = V$.

MinBudget(G, s, T)
Instance: A rooted graph $(G(V, E), s)$, and $T \subseteq V$
Objective: Find a valid vaccination strategy Ψ with minimum possible budget B, such that if s is the only infected node at time 0, then at the end of the above process all nodes in T are saved.

In other words, in MaxSave we are interested in saving as many nodes of T as possible given a fixed budget, and in MinBudget we are interested in finding the minimum necessary budget to save all nodes in T.

3 Spreading Vaccination Model

We achieve tight results in the spreading vaccination model for both the MaxSave and MinBudget problems. We give a $(1 - 1/e)$ approximation for the MaxSave problem. For the MinBudget problem, we use the above approximation to get a $\ln n$-factor approximation. In fact, more generally this reduction (see Theorem 3.9) implies that a $(1 - 1/e^{\alpha})$-approximation for MaxSave (where $\alpha \geq 0$) yields a $\ln n/\alpha$ approximation guarantee for the MinBudget problem. We also show a set cover hardness (i.e., $(1 - o(1)) \ln n$ hardness) for the MinBudget problem. Thus, this hardness result also implies that approximating the MaxSave problem to any constant factor $c > (1 - 1/e)$ is hard. To do so, we use a characterization of the saved vertices given a vaccination strategy. We describe this first.

3.1 Characterizing Saved Vertices in the Spreading Model

We start with a useful observation. Let $d(u, v)$ be the shortest distance between the nodes u and v in graph G, and let $N(v, i)$ be the set of all the nodes that are a distance of at most i from v.

Lemma 3.1. *At time τ, all nodes in the neighborhood $N(s, \tau)$ will either be vaccinated or infected.*

Proof. Let $P(u, v)$ be a shortest path between nodes u and v.

We prove this lemma by induction. The base case for time $\tau = 0$ is true since all nodes in neighborhood $N(s, 0)$, which contains only s, are infected. Let us assume that the statement of the lemma is true for all time-steps $\tau = \{1 \dots k\}$ for some $k \geq 0$. Our assumption implies that all nodes in the neighborhood $N(s, k)$ are either infected or vaccinated at time $\tau = k$. Now consider the set of nodes $\Delta = \{v : v \in N(s, k+1) \backslash N(s, k)\}$. Each of these nodes have a neighbor in $N(s, k)$ which are either infected or vaccinated according to the hypothesis. According to the definition of the model, all nodes that are infected\vaccinated, infect\vaccinate their neighboring nodes in the next time-step. Hence all nodes in Δ will be either infected or vaccinated at time-step $k + 1$. This proves the inductive hypothesis and hence statement of the lemma. ∎

A consequence of the above lemma is that if (u, τ) is in a optimal vaccination strategy, we will have $\tau < d(s, u)$. Henceforth, we will enforce any vaccination strategy to satisfy this property.

We now define a set $\Gamma(v) \subseteq V \times J$ for every node $v \in V$ which will be used to characterize if v is saved by a vaccination strategy Ψ or not. Let

$$\Gamma(v) := \{(u, \tau) | u \in V \text{ and } 0 < \tau \leq d(s, v) - d(u, v)\}$$

Recall, the tuple (u, τ) represents the direct vaccination of the node u at time τ. The following theorem states that a vertex v is saved by a vaccination strategy Ψ in the spreading model, if and only if the strategy vaccinates some u directly at time τ, such that $(u, \tau) \in \Gamma(v)$. This allows us to get a handle on the structure of Ψ.

Theorem 3.2. *A node $v \in V$ is saved by the vaccination strategy Ψ if, and only if, $\Psi \cap \Gamma(v) \neq \emptyset$.*

Proof. ***(if:)*** Let $(u, \tau_u) \in \Psi \cap \Gamma(v)$. Let $P(u, v)$ be a path from u to v of length $d(u, v)$. We prove a stronger statement: all the vertices in $P(u, v)$ are saved. Suppose a vertex $w \in P(u, v)$ is infected. Let it be the nearest such vertex to u. Then it must be the case that $d(s, w) < \tau_u + d(u, w)$, for otherwise the vaccination from u would've spread to w. This implies, $d(s, v) \leq d(s, w) + d(w, v) < \tau_u + d(u, w) + d(w, v) = \tau_u + d(u, v)$; the last equality follows since $w \in P(u, v)$. However, this contradicts $(u, \tau_u) \in \Gamma(v)$.

(only if:) Suppose a vertex v is saved by Ψ. v must be vaccinated either directly or indirectly. Either v is vaccinated directly at time τ_v; in this case by Lemma 3.1 we must have $0 < \tau_v \leq d(s, v)$ and thus $(v, \tau_v) \in \Gamma(v)$. Or, v is vaccinated indirectly. Let the first vaccination to reach v be from vertex u. Lemma 3.1 implies that u had to vaccinated at time $0 < \tau_u \leq d(s, u)$. Furthermore, $\tau_u + d(u, v) \leq d(s, v)$, otherwise v would have been infected by s before vaccination reached v from u. Therefore, (u, τ_u) belongs to the set $\Gamma(v)$. ∎

3.2 The MaxSave Problem

Using Theorem 3.2, we now show that the MaxSave problem in the spreading vaccination model can be cast as the problem of maximizing a monotone submodular function over a partition matroid constraint.

Recall, given a ground set U, a function $f : 2^U \to \mathbb{R}_{\geq 0}$ is a monotone submodular function iff

- $f(A) \leq f(B)$ whenever $A \subseteq B \subseteq U$; and
- $f(A) + f(B) \geq f(A \cup B) + f(A \cap B) \, \forall \, A, B \subseteq U$

A set system $(U, \mathcal{F})$ is a partition matroid if there is a partition of the ground set $U = U_1 \cup \cdots \cup U_k$, and there exists integers $\ell_1, \ldots, \ell_k$ such that $X \subset U$ is in $\mathcal{F}$ if and only if $|X \cap U_i| \leq \ell_i$, for all i. The problem of maximizing a monotone submodular function f over a partition matroid $(U, \mathcal{F})$ is to find a subset $X \in \mathcal{F}$ which maximizes $f(X)$.

The following is known about the problem.

Theorem 3.3. *There is a deterministic algorithm which achieves a 2-approximation for the problem of maximizing a monotone submodular function over any matroid. ([32]).*
There is a randomized algorithm which achieves a $e/(e-1)$-approximation for the problem of maximizing a monotone submodular function over any matroid. ([7]).

Theorem 3.4. *In the spreading vaccination model,* MaxSave *is a special case of maximizing a monotone submodular function over a partition matroid constraint.*

Proof. For the sake of expositional simplicity, we consider the MaxSave Problem where $T = V$; the same analysis holds for any $T \subseteq V$.

We define $\mathcal{E}$ as the set of all possible direct vaccination tuples (v, τ) where $v \in V$ and $\tau = 1 \ldots n$. Note that from Lemma 3.1, we can assume $\tau \leq d(s, v)$ for all $(v, \tau) \in \mathcal{E}$. $\mathcal{E}$ can be partitioned as follows

$$\mathcal{E} = \bigcup_{\tau=1}^{n} \mathcal{E}_\tau \quad \text{where} \quad \mathcal{E}_\tau = \{(v, \tau) | (v, \tau) \in \mathcal{E}\}$$

Claim 3.5. *A vaccination strategy Ψ is valid if and only if $|\Psi \cap \mathcal{E}_\tau| \leq B$, for all τ. That is, the collection of vaccination strategies forms a partition matroid over $\mathcal{E}$.*

Proof. A valid vaccination strategy should directly vaccinate at most B nodes at each time-step. Thus, if Ψ is valid, then it satisfies the condition of the claim. On the other hand, since we assume $\tau \leq d(s, v)$ for all $(v, \tau) \in \mathcal{E}$, each tuple (v, τ) is a valid direct vaccination option, and the partition condition satisfies the budget constraint. ■

We now show that the optimal Ψ, the vaccination strategy which maximizes the number of saved nodes, also maximizes a certain submodular function over $\mathcal{E}$. This function will be a *coverage* function. That is, for each element $(u, \tau) \in \mathcal{E}$, we define a set $S_{(u,\tau)}$, and the function $f : 2^{\mathcal{E}} \to \mathbb{R}_{\geq 0}$ is given as

$$f(\Psi) := \left| \bigcup_{(u,\tau) \in \Psi} S_{(v,\tau)} \right|$$

Such a function is known to be submodular (see [32], for instance).

We now define the sets $S_{(u,\tau)}$ and show that the number of nodes saved by a vaccination strategy Ψ, is precisely $f(\Psi)$. This completes the proof. These sets will be derived using Theorem 3.2. Recall, we proved that a node v is saved iff $\Gamma(v) \cap \Psi$ is nonempty, where $\Gamma(v) = \{(u, \tau) | u \in V \,\text{and}\, 0 < \tau \leq d(s, v) - d(u, v)\}$ Define

$$S_{(u,\tau)} := \{v \in V : (u, \tau) \in \Gamma(v)\}.$$

Thus a node v is saved if and only if $v \in S_{(u,\tau)}$ for some (u, τ) in Ψ. Therefore, the number of nodes saved by Ψ is precisely $f(\Psi)$. ■

Thus using Theorem 3.3 and 3.4, we get constant factor approximations for MaxSave in the spreading vaccination model.

3.3 The MinBudget Problem

We first show that in the spreading model, the MinBudget problem on directed graphs is as hard as set cover. This implies a logarithmic inapproximability. Subsequently, we show a $\ln n$ factor approximation.

Theorem 3.6. *In the spreading vaccination model, the* MinBudget *problem is as hard as set cover.*

Proof. Consider an instance of Set Cover: a collection $\mathcal{C}$ of finite subsets of U. Let $|\mathcal{C}| = k$ and $|U| = n$. An instance of the MinBudget problem can be constructed as follows:

- Construct node s which will be the root node.
- For each subset $S \in \mathcal{C}$, construct a node v_S with a directed edge (s, v_S).
- For each element $e \in U$, construct k nodes $v_{e_1} \dots v_{e_k}$ such that there is a directed edge (v_S, v_{e_i}) for $i = 1 \dots k$ if $e \in S$.

This graph forms the input for the MinBudget problem with $T = \{v_{e_1} \dots v_{e_k} \forall e \in U\}$. Each element of U is represented by k nodes, each of them connected by an edge coming from the nodes that represent the sets to which the elements belong. The following claim proves the theorem.

Claim 3.7. *There is a set cover of size B if and only if there is a vaccination strategy immunizing at most B nodes per time step.*

Proof. ***(only if:)*** Suppose there is a set cover $\mathcal{C}' \subseteq \mathcal{C}$ of size B. Consider the vaccination strategy of immunizing all v_S for $S \in \mathcal{C}'$ in the first time step. Since $\mathcal{C}'$ is a set cover, all vertices v_{e_i} get indirectly vaccinated on the second time step.

(if:) We may assume $B < k$, for otherwise the claim is trivial. We may assume that in any optimal vaccination strategy, no vertex v_{e_i} is directly vaccinated in the first time step. This is because it is better to vaccinate v_S for any S containing e, as that would lead to spread of vaccination to other v_{e_j}. Let $\mathcal{C}' := \{S : v_S \text{ is immunized on the first time step }\}$. Note that $|\mathcal{C}'| \leq B$. We claim that this is a set cover. If not, there is an element e not in any set in $\mathcal{C}'$. But then, on the second time step, the infection reaches all the k copies of this element. Since $B < k$, one cannot directly vaccinate all these copies, and hence we reach a contradiction. ∎

Corollary 3.8. *There is no $(1 - o(1)) \ln n$ approximation for the* MinBudget *problem in the spreading model, unless $NP \subseteq DTIME(n^{\mathtt{polylog}(n)})$.*

The above follows from the result of Feige [15]. ∎

Theorem 3.9. *There is a polynomial time $\ln n$-factor approximation algorithm for* MinBudget *in the spreading model.*

Proof. Such an algorithm can be found by using the $O(1)$ approximation algorithms for the MaxSave problem, described in the previous section. We guess the optimum B of the MinBudget problem by doing a binary search. Since there is a vaccination strategy with budget B which saves all of T, we can save at least $(1 - 1/e)$ fraction of the vertices in T by running the algorithm described in Theorem 3.3. Let T_1 be the set of nodes in T not saved. Repeating the argument, there is a strategy with budget B, which saves $(1 - 1/e)$ fraction of the nodes of T_1. This implies, by running the two strategies together, there is a budget $2B$ vaccination strategy, which saves all but a $1/e^2$ fraction of nodes of T. Going on, we can compute a vaccination strategy with budget $(\ln n) \cdot B$, which saves all the nodes of T. ∎

4 Non-Spreading Vaccination Model

The non-spreading model is considerably more difficult to reason about than the spreading model. For instance, the structural lemma, Lemma 3.1 (or any simple modification of it), is no longer true in this model. Thus, it is difficult to characterize the set of nodes which are saved by a vaccination strategy. We show that the MaxSave problem is hard to approximate in this model to any nontrivial factor. Nonetheless, we achieve a $O(\sqrt{n})$ for the MinBudget problem, which we improve to $O(\log n)$ for directed layered networks.

4.1 The MaxSave Problem

Unlike the spreading vaccination model where we got constant factor approximation algorithms for the MaxSave problem, we show in our next theorem that it is NP hard to get an approximation factor of $n^{1-\epsilon}$, for any constant $\epsilon > 0$. In fact, the theorem holds even when the budget parameter B is set to 1.

Theorem 4.1. *For any $\epsilon > 0$, it is NP hard to obtain an $n^{1-\epsilon}$ factor approximation to the* MaxSave $(G, 1, s, T)$ *problem.*

Proof. We introduce an auxiliary problem, Save-t, whose input is a graph G, and two specified nodes s and t. The problem is to decide if t can be saved from infection spreading from s, by vaccinating 1 node (other than t) at each time step. The NP-completeness of this problem follows from known NP-completeness of the MinBudget problem [16] (let the neighbors of t be T for the MinBudget problem). The hardness of MaxSave is obtained by replacing t with a polynomially large set.

Given an instance $(G = (V, E), s, t)$ of Save-t with $|V| = n$, we construct an instance $(G' = (V', E'), 1, s, T')$ of MaxSave as follows. The vertex set V' contains a copy of each vertex in $V \setminus t$ and an additional set T of n^β vertices, the value of β to be fixed later. We abuse notation and call the set of copies of $V \setminus t$, as V as well. The edge set E' consists of all original edges $\{(u, v) : (u, v) \in E\}$ of G not incident on t, and the set of edges $\{(u, v) : u \text{ is incident to } t \text{ in } G, \text{ and } v \in T\}$. Thus, G' is exactly G, except we take n^β copies of t, which we call T. The target set T' in the MaxSave problem is the set V', that is, we want to save all the vertices of G'. We let $N = |V'| = n^\beta + n - 1$.

The proof of the theorem follows from the following two claims.

Claim 4.2. *If there is a vaccination strategy to save t in (G, s, t), then there is a vaccination strategy in $(G', 1, s, T')$ which can save at least n^β nodes.*

Proof. It is clear that exactly the same strategy used in the Save-t instance will save all the nodes of T in the MaxSave instance. Thus, one can save at least n^β nodes. ■

Claim 4.3. *If there is no vaccination strategy to save t in (G, s, t), then any vaccination strategy in $(G', 1, s, T')$ can save at most $2n$ nodes.*

Proof. Since there are only n nodes in G, we know that in n steps of any vaccination strategy, t is infected. Therefore, in n steps of any vaccination strategy for the MaxSave instance, all of the unvaccinated nodes of T are infected. In n steps, one can save at most n nodes of T, and at most all the nodes (n) of V. Thus, one can save at most $2n$ nodes. ■

The above two claims imply that it is NP hard to approximate MaxSave to a factor better than $(n^{\beta-1})/2$. Let $\beta = \frac{1}{\epsilon}$. But, $N = n^\beta + n - 1 < 2n^\beta$, for any $\beta \geq 1$. Thus we have that $N^{(1-\frac{1}{\beta})} < 2^{1-\epsilon} n^{\beta-1}$, and so we get a hardness of $\Omega(N^{1-\epsilon})$ which completes the proof. ■

4.2 The MinBudget Problem

Recall that we need to save *all* the nodes in a set T with the minimum number of vaccinations required per time step. To simplify notation, we consider the following equivalent problem: we add a new node t with edges from all nodes in T to t, and consider the problem of saving t with minimum budget *under the additional constraint that t itself cannot be vaccinated.* We call s the source and t the sink. Let $\mathcal{P}$ denote the collection of all s-t paths.

In the remainder of the section, we give two approximation algorithms. Our first is a $2\sqrt{n}$ factor algorithm for the problem; contrast this with the hardness of approximation for the MaxSave version. In fact, we show that the minimum s-t cut achieves this factor. We also give an improved $O(\log n)$ algorithm for the case of directed layered graphs. The analysis of our algorithm utilizes a natural LP relaxation for the problem, and as a corollary we obtain that the integrality gap of this LP is $O(\log n)$. We also give an example that proves a matching lower bound of $\Omega(\log n)$ on the integrality gap of this LP relaxation.

We start with the $O(\sqrt{n})$ factor algorithm.

Theorem 4.4. *There is a $2\sqrt{n}$ factor approximation algorithm for the* MinBudget *problem.*

Proof. Let B^* be the budget required by the optimum solution. We will show that the minimum s-t cut is of size at most $(B^*+1)\sqrt{n}$, which will prove the factor since $B^* \geq 1$. Note that vaccinating any s-t cut on the first time step is a valid solution.

Consider the set of nodes Y vaccinated in the first $\sqrt{n}$ time steps. Note that $|Y| \leq B^*\sqrt{n}$. Since this is a valid vaccination strategy, any path from s of t of length at most $\sqrt{n}$ contains at least one vertex of Y. Consider the graph G' induced by the vertex set $V' = V \setminus Y$. If s and t are disconnected in G', then Y is an s-t cut. Otherwise, the shortest path from s to t in G' is at least $\sqrt{n}$. This implies that the minimum s-t cut, call it X, in G' is of size at most $\sqrt{n}$. This follows from Menger's theorem which states that if the minimum s-t cut is k, there are k internally vertex disjoint paths from s to t; since each such path is of length at least $\sqrt{n}$, we must have $k \leq \sqrt{n}$. Thus there is a cut $(X \cup Y)$ of size $(B^*+1)\sqrt{n}$ separating s and t. ■

Now we give an approximation algorithm for directed layered graphs. In an *s-t directed ℓ-layered network*, the vertex set consists of $V = (L_0 := \{s\}) \cup L_1 \cup \cdots \cup L_\ell \cup \{t\}$, and all arcs except those entering t are from a vertex in some layer L_i to a vertex in L_{i+1}; arcs entering t may originate from any vertex (other than t). We give an $H(\ell)$ factor, where $H(t) = 1 + \frac{1}{2} + \cdots + \frac{1}{t}$. The algorithm is based on the LP relaxation and its dual described in Figure 2.

$$\begin{array}{lll}
\text{Minimize} & B & \text{(Primal)} \\
& \displaystyle\sum_{v \in V} x_v^\tau \leq B \quad \forall \tau = 1, \ldots, \ell & (1) \\
& \displaystyle\sum_{i=1}^{k} \sum_{\tau=1}^{i} x_{v_i}^\tau \geq 1 \quad \forall (s, v_1, \cdots, v_k, t) \in \mathcal{P} & (2) \\
& x_v^\tau \geq 0, \quad \forall v \in V,\ \forall \tau = 1, \ldots, \ell & (3)
\end{array}$$

$$\begin{array}{lll}
\text{Maximize} & \displaystyle\sum_{P \in \mathcal{P}} f_P & \text{(Dual)} \\
& \displaystyle\sum_{\tau=1}^{\ell} z_\tau \leq 1 & (4) \\
& \displaystyle\sum_{P \in \mathcal{P} : v \in P^{(\tau)}} f_P \leq z_\tau \quad \forall v \in V, \tau = 1, \ldots, \ell & (5) \\
& z, f \geq 0 & (6)
\end{array}$$

Figure 2: The LP relaxation for MinBudget in the non-spreading model and its dual

The primal LP has a variable x_v^τ which indicates whether vertex v is vaccinated at time τ or not. $\ell \leq n$ is the length of the longest path from s to t; it is easy to see that we will not vaccinate any vertex after time ℓ. The first constraint bounds the number of vaccinations at every time instance. The second constraint says that for every path $(s, v_1, \cdots, v_k, t)$ to the sink t, one of the nodes, say v_i, must be vaccinated *by* time i. This is a necessary and sufficient condition for this path not to transmit the infection to t. In the dual, we have a flow for every s-t path P. We also have a variable z_τ which add up to 1. The second constraint in the dual is a bit subtle: it says, for every τ, the total flow through a vertex v via paths such that v lies at a distance τ or more from s on the path, is at most z_τ. In the LP, $P^{(\tau)}$ denotes the portion of the path from the τth vertex to t. That is if $P = (s, v_1, \ldots, v_k, t)$, then $P^{(\tau)} = (v_\tau, v_{\tau+1}, \ldots, t)$.

Although the primal LP above has exponentially many constraints, it can be solved in polynomial time since one can obtain the separation oracle in polynomial time. However, we will give a combinatorial algorithm and use the LP (in fact, the dual) only for analysis. Strictly speaking the LP (Primal) may have an integrality gap of $n = |V|$. However note that if OPT denotes the optimal value of (Primal), then in fact $\lceil OPT \rceil$ is a lower bound on the minimum budget, and by comparing the budget of our solution against this lower bound, we prove the following theorem.

Theorem 4.5. *If the network is a layered directed graph with ℓ layers, then there is a poly-time $H(\ell)$ approximation algorithm to the* MinBudget *problem.*

Proof. The algorithm first computes a 'fractional' vaccination strategy, that is, the strategy would be feasible except it could possibly vaccinate a non integral number of vertices. The second step converts this strategy into a feasible one with essentially the same guarantees. For this last step we need the following fact which follows from standard results about minimum-cost network flows.

Fact 4.6 (see, e.g., Application 6.3 in [1]). *Given a matrix M' with possibly fractional entries, one can obtain another* integral *matrix M such that for every row i and column j, we have (i) $M_{ij} \in \{\lfloor M'_{ij} \rfloor, \lceil M'_{ij} \rceil\}$; (ii) the row-sum of row i in M is the floor or ceiling of the row-sum of row i in M'; and (iii) the column-sum of column j in M is the floor or ceiling of the column-sum of column j in M'.*

Algorithm DirLayNet

1. Set the capacity of each vertex $v \in L_i$ at $\frac{1}{iH(\ell)}$.
2. Find the minimum cut in this capacitated network, let it be $(N_1 \cup \cdots \cup N_\ell)$ with $N_i \subseteq L_i$.
3. Construct the following possibly non-integral matrix M'. Let $M'_{ij} := |N_j|/j$, for all $1 \leq i \leq j \leq \ell$. Note that for any column j, the column sum of M' is exactly $|N_j|$.
4. Apply Fact 4.6 to construct the corresponding integral matrix M from M'. The vaccination strategy is as follows: on day i, vaccinate M_{ij} nodes from layer j, for all $i \leq j \leq \ell$.

Claim 4.7 shows that the above algorithm returns a feasible vaccination strategy. Define $ALG := |N_1| + \frac{|N_2|}{2} + \cdots + \frac{|N_\ell|}{\ell}$. The number of vertices vaccinated on any day i is

$$\sum_{j \geq i} M_{ij} \leq \lceil \sum_{j \geq i} M'_{ij} \rceil \leq \lceil \sum_{j \geq 1} M'_{ij} \rceil = \lceil ALG \rceil.$$

Thus, the cost of algorithm DirLayNet (i.e., the maximum number of vertices vaccinated in any time step) is at most $\lceil ALG \rceil$. Claim 4.8 proves that there is a feasible dual solution to (Dual) of value $\frac{1}{H(\ell)} \cdot ALG$, and hence $OPT \geq \frac{1}{H(\ell)} \cdot ALG$. This implies that the cost of algorithm DirLayNet is at most $\lceil H(\ell) \cdot OPT \rceil \leq H(\ell) \lceil OPT \rceil$. ■

Claim 4.7. *Algorithm* DirLayNet *returns a feasible vaccination strategy.*

Proof. We claim that for any $1 \leq j \leq \ell$, all the nodes of N_j are vaccinated by time step j. This ensures that the vaccination strategy is feasible, since any path $(s, v_1, v_2, \ldots, v_k, t)$ from s to t (where $k \leq \ell$), must have $v_j \in N_j$ for some j ($\bigcup_j N_j$'s form a cut).

By day j, we vaccinate $\sum_{i \leq j} M_{ij}$ nodes of N_j. But, $\sum_{i \leq j} M'_{ij} = |N_j|$, and therefore integral; this implies $\sum_{i \leq j} M_{ij}$ is also precisely that. This completes the proof. ■

Claim 4.8. *There is a feasible dual solution to* (Dual) *of value $ALG/H(\ell)$.*

Proof. Look at the LHS of the second constraint in the dual. We claim that for layered networks, if vertex v lies in L_i, for any feasible dual solution f we have

$$\sum_{P: v \in P^{(\tau)}} f_P = \begin{cases} \sum_{P: v \in P} f_P & \text{if} \quad t \leq i \\ 0 & \quad \text{otherwise} \end{cases} \tag{7}$$

This is because *every path that contains $v \neq t$, contains it at exactly the ith position.* Now we exhibit the desired dual solution. To do so, note that corresponding to the minimum vertex cut $(N_1 \cup \ldots \cup N_\ell)$, we have a feasible flow f of the same value. Furthermore, because of the capacity constraints, we get for $v \in L_i$,

$$\sum_{P: v \in P} f_P \leq \frac{1}{iH(\ell)} \tag{8}$$

Construct the dual solution with this f and let $z_i = \frac{1}{iH(\ell)}$ for $i = 1, \ldots, \ell$. Note that the first constraint of the dual is satisfied. The second constraint follows from (7) and (8), and the fact that z_i's are decreasing. Note that the value of the dual is equal to the capacity of the minimum cut which is

$$\frac{1}{H(\ell)}|N_1| + \frac{1}{2H(\ell)}|N_2| + \cdots + \frac{1}{\ell H(\ell)}|N_\ell| = \frac{ALG}{H(\ell)}.$$

■

We remark that the special case where the underlying graph G is a tree rooted at s has received a lot of attention [23, 28] and is computationally difficult [16]. Notice that for trees the spreading model and non-spreading model are equivalent due to the following reason. For the spreading model on general graphs we defined a function $\Gamma(v)$ as a set of all tuples (u, τ) such that if u is vaccinated directly at time τ then the node v will be saved. For a tree, it is easy to observe that a node v will be saved if any of its ancestors is vaccinated directly before the infection reaches v. Therefore, the optimal strategy will be the same on a given tree irrespective of the vaccination model being spreading or non-spreading. This implies that all the positive results from Section 3 also hold for trees. Also, observe that the MinBudget problem on a tree G with height h yields an instance of MinBudget on an s-t directed graph with h layers. Hence, we immediately obtain the following corollary of Theorem 4.5.

Corollary 4.9. *There is an $O(\log h)$-approximation for* MinBudget *on trees, where the set T is the set of leaves and h is the height of the tree.*

4.3 An integrality-gap example

We conclude this section by showing an integrality gap of $\Omega(\log n)$ for the above LP relaxation. The example is an ℓ layered graph containing $O(\ell^3)$ vertices. We show a gap of $H(\ell) = \Omega(\log n)$.

Consider an s-t ℓ-layered graph with layers ($L_0 = \{s\}, L_1, \ldots, L_\ell$) with an arc from s to all vertices of L_1 and an arc from any vertex in L_ℓ to t. Furthermore, there will be an arc from any vertex in L_i to any vertex in L_{i+1} for $i = 1, \ldots, \ell - 1$. The size of L_i is $i\ell$. Thus, the total number of vertices is $O(\ell^3)$. Note that the integral optimum will be ℓ; one can show if one vaccinates less than $i\ell$ nodes by time i, then the infection spreads to layer $i+1$. To see a fractional solution, consider the solution where for each i, we set $x_v^i = \frac{1}{iH(\ell)}$ for all $v \in L_i$, and $x_v^i = 0$ for all other v. Any path from s to t then has the LHS of constraint 2 precisely equal to $\frac{1}{H(\ell)}(1 + 1/2 + \cdots + 1/\ell) = 1$. The value of the solution is $i\ell \frac{1}{iH(\ell)} = \frac{\ell}{H(\ell)}$.

5 Concluding Remarks

In this paper we looked at two models: one where the rate of spread of the vaccination is the same as the rate of spread of the infection (*spreading model*); and the other where the rate of spread of vaccination is 0, i.e. it does not spread (*non-spreading model*). In reality the rate of spread of the vaccination may lie somewhere in the middle. The lemmas applied for the spreading model fail to hold for these intermediate cases. Hence the study of such models give rise many interesting open problems.

Another area of open problems is where the model of infection spread is more complex. For example, an infected individual might go through stages of incubation and initial symptomatic period causing a small stochastic delay before the person becomes contagious for his neighbors. We can also consider adding probabilities of transmission on the edges or probabilities on nodes indicating their susceptibility to the disease.

Acknowledgments

The authors would like to thank David Kempe, Sanjeev Khanna, Jon Kleinberg, and Malik Magdon-Ismail for interesting and productive discussions on the Firefighter problem.

References

[1] R. Ahuja, T. Magnanti, and J. Orlin. *Network Flows: Theory, Algorithms, and Applications.* Prentice Hall, 1993.

[2] J. Aspnes, K. Chang, and A. Yamposlkiy. Inoculation strategies for victims of viruses and the sum-of-squares partition problem. In *Proc. 16th ACM Symp. on Discrete Algorithms (SODA)*, 2005.

[3] G. Baier, T. Erlebach, A. Hall, E. Köhler, H. Schilling and M. Skutella M. Length-Bounded Cuts and Flows. In *Proc. 33rd International Colloquium on Automata, Languages and Programming (ICALP)*, pages 679–690, 2006.

[4] N. Bailey. *The Mathematical Theory of Infectious Diseases and its Applications.* Hafner Press, 1975.

[5] A. Barabasi, R. Albert, and H. Jeong. Mean-field theory for scale-free random networks. *Physica A*, 272:173–187, 1999.

[6] N. Berger, C. Borgs, J. Chayes, and A. Saberi. On the spread of viruses on the internet. In *Proc. 16th ACM Symp. on Discrete Algorithms*, 2005.

[7] G. Calinescu, C. Chekuri, M. Pal, J. Vondrak. Maximizing a Monotone Submodular Function subject to a Matroid Constraint. *Integer Programming and Combinatorial Optimization, 12th International IPCO Conference*, Ithaca, NY, USA, 2007.

[8] P. Chalermsook and J. Chuzhoy. Resource Minimization for Fire Containment. To appear in Proc. ACM SODA, 2010.

[9] Stuart Crosby, Art Finbow, Bert Hartnell, Raina Moussi, Kate Patterson and Dania Wattar. Designing Fire Resistant Graphs. *Congr. Numerantium* 173, 207-222, 2005.

[10] M. Develin, S.G. Hartke. Fire Containment in grids of dimension three or higher. *Discrete Applied Math*,155(17), 2257-2268, 2007.

[11] Z. Dezső and A. Barabási. Halting viruses in scale-free networks. *Physical Review E*, 65, 2002.

[12] P.A. Dreyer Jr. and F. S. Roberts. Irreversible k-threshold processes: Graph-theoretical threshold models of the spread of disease and of opinion. *Discrete Applied Mathematics*, 157(7), 1615-1627, 2009.

[13] R. Engelberg, J. Könemann, S. Leonardi, and J. Naor. Cut problems in graphs with a budget constraint. In *Proc. 7th Latin American Theoretical Informatics Symposium*, 2006.

[14] S. Eubank, V. Kumar, M. Marathe, A. Srinivasan, and N. Wang. Structural and algorithmic aspects of massive social networks. In *Proc. 15th ACM Symp. on Discrete Algorithms*, pages 711–720, 2004.

[15] U. Feige A threshold of $\ln n$ for approximating set cover, *J. ACM* 45, 634-652, 1998.

[16] S. Finbow, A. D. King, Gary MacGillivray, R. Rizzi. The Fire fighter problem on graphs of maximum degree three. *Discrete Mathematics 307* 2094-2105 2007.

[17] Stephen Finbow, Gary MacGillivray The Firefighter Problem: A survey of results, directions and questions. manuscript, 2007.

[18] M.L. Fisher, G.L. Nemhauser and L.A. Wolsey An analysis of approximations for maximizing submodular set functions - II. *Math. Prog. Study*, 8:7387, 1978.

[19] P. Fogarty. *Catching Fire on Grids*, M.Sc. Thesis, Department of Mathematics, University of Vermont 2003.

[20] A. Ganesh, L. Massoulie, and D. Towlsey. The effect of network topology on the spread of viruses. In *Proc. 24th IEEE INFOCOM Conference*, 2005.

[21] G. Giakkoupis, A. Gionis, E. Terzi, P. Tsaparas. Models and algorithms for network immunization. *Technical Report C-2005-75*, Department of Computer Science, University of Helsinki, 2005.

[22] B.L. Hartnell. Firefighter! An application of domination. Presentation, *25th Manitoba Conference on Combinatorial Mathematics and Computing*, University of Manitoba in Winnipeg, Canada, 1995.

[23] B. Hartnell, Q. Li. Firefighting on trees: How bad is the greedy algorithm? *Congr. Numer.* 145, 187-192 2000.

[24] A. Hayrapetyan, D. Kempe, M. Pal and Z. Svitkina Unbalanced graph cuts. In *European Symposium on Algorithms (ESA)*, Mallorca, Spain 2005.

[25] D. Kempe, J. Kleinberg and E Tardos. Influential Nodes in a Diffusion Model for Social Networks. In *Proceedings of the 32nd International Colloquium on Automata, Languages and Programming (ICALP)*, 2005.

[26] Andrew King, Gary MacGillivray. The Firefighter Problem For Cubic Graphs. *Discrete Mathematics*, 307: 20942105, 2007.

[27] R. Kumar, P. Raghavan, S. Rajagopalan, D. Sivakumar, A. Tomkins, and E. Upfal. Stochastic models for the web graph. In *Proc. 41st IEEE Symp. on Foundations of Computer Science (FOCS)*, pages 5765, 2000.

[28] C. Leizhen, E. Verbin and L. Yang Firefighting on Trees: (1 - 1/ e)Approximation, Fixed Parameter Tractability and a Subexponential Algorithm, *ISAAC 2008*: 258269, 2008.

[29] Cai Leizhen and Wang Weifan. The Surviving Rate of a Graph. to appear in *SIAM Journal on Discrete Mathematics*, 2009.

[30] Gary MacGillivray, P. Wang. On The Firefighter Problem *JCMCC 47*, 83-96, 2003.

[31] C. Moore and M. Newman. Epidemics and percolation in small-world networks. *Physical Review E*, 61:5678–5682, 2000.

[32] G. Nemhause and L. Wolsey. Integer and Combinatorial Optimization John Wiley and Sons, 1988.

[33] M. Nowak and R. May. *Virus Dynamics: Mathematical Principles of Immunology and Virology.* Oxford University Press, 2000.

[34] R. Pastor-Satorras and A. Vespignani. Epidemic dynamics in finite scale-free networks. *Physical Review E*, 65, 2002.

[35] R. Raz and S. Safra. A sub-constant error-probability low-degree test, and sub-constant error-probability PCP characterization of NP. In *Proc. 29th Ann. ACM Symp. on Theory of Comp., ACM*, 475-484, 1997.

[36] P. Wang, S. Moeller. Fire Control on graphs. *J. Combin. Math. Combin. Comput.* 41, 19-34, 2002.

[37] D. Watts and S. Strogatz. Collective dynamics of 'small-world' networks. *Nature*, 393:440–442, 1998.