Cheap, Easy, and Massively Effective Viral Marketing in Social Networks: Truth or Fiction?

Thang N. Dinh, Dung T. Nguyen and My T. Thai
University of Florida
Gainesville, FL, 32603
{tdinh, dtnguyen, mythai}@cise.ufl.edu

ABSTRACT

Online social networks (OSNs) have become one of the most effective channels for marketing and advertising. Since users are often influenced by their friends, "word-of-mouth" exchanges so-called viral marketing in social networks can be used to increases product adoption or widely spread content over the network. The common perception of viral marketing about being cheap, easy, and massively effective makes it an ideal replacement of traditional advertising. However, recent studies have revealed that the propagation often fades quickly within only few hops from the sources, counteracting the assumption on the self-perpetuating of influence considered in literature. With only limited influence propagation, is massively reaching customers via viral marketing still affordable? How to economically spend more resources to increase the spreading speed?

We investigate the cost-effective massive viral marketing problem, taking into the consideration the limited influence propagation. Both analytical analysis based on power-law network theory and numerical analysis demonstrate that the viral marketing might involve costly seeding. To minimize the seeding cost, we provide mathematical programming to find optimal seeding for medium-size networks and propose VirAds, an efficient algorithm, to tackle the problem on large-scale networks. VirAds guarantees a relative error bound of $O(1)$ from the optimal solutions in power-law networks and outperforms the greedy heuristics which realizes on the degree centrality. Moreover, we also show that, in general, approximating the optimal seeding within a ratio better than $O(\log n)$ is unlikely possible.

Categories and Subject Descriptors

G.2.2 [**Mathematics of Computing**]: Discrete Mathematics—*Graph theory*; I.1.2 [**Computing Methodologies**]: Algorithms—*Analysis of algorithms*

HT'12, June 25–28, 2012, Milwaukee, Wisconsin, USA.

General Terms

Theory, Algorithm, Measurement

Keywords

Social media, influence propagation, power-law networks, approximation algorithm, hardness proof.

1. INTRODUCTION

Digitizing real world connections, online social networks (OSNs) such as Twitter, Facebook have been steadily growing. Two-third of everyone online is using social networks with 800 million active users using Facebook [2], 200 million twitters, 40 millions Google+ subscribers and so on. Social network sites such as Facebook and Youtube are often among top-ten visited websites on the Internet [1]. Much like real-world social networks, OSNs inherent the viral property in which information can spread and disseminate widely into networks via 'word-of-mouth' exchanges, creating an effective platform for marketing. OSNs quickly become one of the most attractive choices for brand awareness, encouraging discussion on improving products and for recruiting. Notable examples include the recent unrest in many Arab countries which are triggered by Facebook shared posts [23]; the customer outreach of Toyota on Twitter to repair its image after the massive safety recalls of its vehicles [22], and many others. Despite the huge economic and political impact, viral marketing in large scale OSNs is not well understood due to the extremely large numbers of users and complex structures of social links.

A major portion of viral marketing research has been devoted to the question of efficiently targeting a set of *influential nodes* in order to spread information widely into the network [16, 17, 11]. Two essential components to address the question are the diffusion models and the algorithms to select the initial set of nodes, called seeding. For a social network represented as a graph, a diffusion model defines the stochastic process that specifies how influence is propagated from the seeding to their neighbors, and further. In [16] Kempe et al. proposed two basic diffusion models, namely *independent cascade* model and *linear threshold* model. These two models and their extensions set the foundation to almost all existing algorithms to find seeding in social networks [16, 17, 11].

However, all mentioned models and algorithms ignore one important aspect of influence propagation in the real world. That is influence propagation often happens only within a close proximity of the seeding. For examples, study of Cha

et. al. in Flickr network [9] reveals that the typical chain length is less than four; another study of Leskovec et. al. [18] suggests that social influence happens on the level of direct friends. Moreover, shared information in social networks such as Facebook, in most cases, can be seen only by friends or friends of friends i.e. the propagation is basically limited within two hops from the source. When the influence only propagates locally, is massively reaching customers via viral marketing still affordable? In addition, can we speed up the information spreading for time-critical applications such as political campaigns?

We formulate a new optimization problem, called *the cost-effective, fast, and massive viral marketing* (CFM) problem. The problem seeks for a minimal cost seeding, measured as the number of nodes, to massively and quickly spread the influence to the whole network (or a large segment of the network). The new aspect in our model is that the influence is limited to the nodes that are within d hops from the seeding for some constant $d \geq 1$. In other words, the influence is forced to spread to the whole networks within d propagation rounds. Hence, adjusting d gives us an important ability to control how fast the spread of influence within a network. Unfortunately, the huge magnitude of OSN users and data available on OSNs poses a substantial challenge to control how information can quickly spread out to the whole network.

In this paper, we develop solutions to the CFM problem and address the above two questions. More specifically, our contributions are summarized as follows:

- Our first finding shows that the seeding for fast and massive spreading must contain a non-trivial fraction of nodes in the networks, which is cost-prohibitive for large social networks. This is confirmed by both our theoretical analysis based on the power-law model in [4] and our extensive experiments.

- We propose VirAds, a scalable algorithm to find a set of minimal seeding to expeditiously propagate the influence to the whole network. VirAds outperforms the greedy heuristics based on well-known degree centrality and scales up to networks of hundred of million links. We prove that the algorithm guarantees a relative error bound of $O(1)$, assuming that the network is power-law.

- We show how hard to obtain a near optimal solution for CFM by proving the impossibility to approximate the optimal solution within a ratio better than $O(\log n)$.

Related Work. Viral marketing can be thought of as a diffusion of information about the product and its adoption over the network. Kempe et al. [16, 17] formulated the influence maximization problem as an optimization problem. They showed the problem to be NP-complete and devised an $(1 - 1/e - \epsilon)$ approximation algorithm. A major drawback of their algorithm is that the accuracy ϵ, and efficiency depends on the number of times running Monte-Carlo simulation of the propagation model. Later, Leskovec et al. [19] study the influence propagation in a different perspective in which they aim to find a set of nodes in networks to detect the spread of virus as soon as possible. They improve the simple greedy method to run faster. The greedy algorithm is furthered improved by Chen et al. [10] by using an influence estimation. However, the proposed algorithm might only perform well for small values of propagation probabilities. In addition, the algorithm time complexity should be $O((m + k) \log n)$ instead of the claimed $O(k \log m + m)$.

Influence propagation with limited number of hops is first considered in Wang *et al.* [26] in which the proposed heuristic has high time complexity. Feng et al. [27] show NP-completeness for the problem. We note that none of the mentioned approaches handled large-scale social networks of million of nodes as we shall study in Section 6.

Organization. We introduce the limited hop influence model and the cost-effective, massive and fast propagation problem (CFM) in Section 2. In Section 3, we answer the question on the seeding cost by analyzing the propagation process on power-law networks. We present VirAds, a scalable algorithm to find a minimal seeding for the CFM problem in Section 4. The hardness of finding a cost-effective seeding is addressed in Section 5. Finally, we perform extensive experiments on large social networks such as Facebook and Orkut to confirm the efficiency of our proposed algorithm and analyze the results to give new observations to information diffusion process in networks.

2. PROBLEM DEFINITIONS

We are given a *social network* modeled as an undirected graph $G = (V, E)$ where the vertices in V represent users in the network and the edges in E represent social links between users. We use n and m to denote the number of vertices and edges, respectively. The set of neighbors of a vertex $v \in V$ is denoted by $N(v)$ and we denote by $d(v) = |N(v)|$ the degree of node v.

We continue with specifying the diffusion model that governs the process of influence propagation. Existing diffusion models can be categorized into two main groups [16]:

- *Threshold model.* Each node v in the network has a threshold $t_v \in [0, 1]$, typically drawn from some probability distribution. Each connection (u, v) between nodes u and v is assigned a weight $w(u, v)$. For a node v, let $F(v)$ be the set of neighbors of v that are already influenced. Then v is influenced if $t_v \leq \sum_{u \in F(v)} w(u, v)$.

- *Cascade model.* Whenever a node u is influenced, it is given a single chance to activate each of its neighbor v with a given probability $p(u, v)$.

Most viral marketing papers assume that the probabilities $p(u, v)$ or weights $w(u, v)$ and thresholds t_v are given as a part of the input. However, they are generally not available and inferring those probabilities and thresholds has remained a non trivial problem [15]. Therefore, in addition to the bounded propagation hop, we use a simplified variation of the linear threshold model in which a vertex is activated if a fraction ρ of its neighbors are active as follows.

Locally Bounded Diffusion Model. Let $R_0 \subset V$ be the subset of vertices selected to initiate the influence propagation, which we call the *seeding*. We also call a vertex $v \in R_0$ a seed. The propagation process happens in round, with all vertices in R_0 are influenced (thus active in adopting the behavior) at round $t = 0$. At a particular round $t \geq 0$, each vertex is either active (adopted the behavior) or inactive and each vertex's tendency to become active increases

when more of its neighbors become active. If an inactive vertex u has more than $\lceil \rho\, d(u) \rceil$ active neighbors at round t, then it becomes active at round $t+1$, where ρ is the *influence factor* as discussed later. The process goes on for a maximum number of d rounds and a vertex once becomes active will remain active until the end. We say an initial set R_0 of vertices to be a *d-seeding* if R_0 can make all vertices in the networks active within at most d rounds.

The influence factor $0 < \rho < 1$ is a constant that decides how widely and quickly the influence propagates through the network. Influence factor ρ reflects real-world factors such as how easy to share the content with others, or some intrinsic benefit for those who initially adopt the behavior. In case $\rho = 1/2$ the model is also known as the *majority* model that has many application in distributed computing, voting system [21], etc.

Problem Definition. Given the diffusion model, the *Cost-effective, Fast, and Massive viral marketing (CFM)* problem is defined as follows

DEFINITION 1 (CFM PROBLEM). *Given an undirected graph $G = (V, E)$ modeling a social network and an influence factor $0 < \rho < 1$, find in V a minimum size d-seeding i.e. a subset of vertices that can activate all vertices in the network within at most d rounds.*

Generalization. The diffusion model can be generalized in several ways. For example, the model can be extended naturally to cover directed networks or specify different influence factor ρ_v for each node $v \in V$. For simplicity we stick with the current model to avoid setting parameters during the experiments. Nevertheless, major results such as the approximation ratio of the VirAds algorithm in Section 4 or the hardness of approximation result in Section 5 still hold for the generalized models.

3. COST OF MASSIVE MARKETING

In this section, we give a negative answer for the first question in the introduction about the initial seeding cost. We exploit the power-law topology found in most social networks [7, 8, 12] to demonstrate that when the propagation hop is limited, a large number of seeding nodes is needed to spread the influence throughout the network. The size of seeding is proved to be a constant fraction of the number of vertices n, which is prohibitive for large social networks of millions of nodes. We first summarize the well-known power-law model in [3]; then we use the model to prove the prohibitive seeding cost for the CFM problem.

3.1 Power-law Network Model.

Many complex systems of interest including OSNs are found to have the degree distributions approximately follows the power laws [7, 8, 12]. That is the fraction of nodes in the network having k connections to other nodes is proportional to $k^{-\gamma}$, where γ is a parameter whose value is typically in the range $2 < \gamma < 3$. Those networks have been used in studying different aspects of the scale-free networks [3, 5, 14]. We follow the $P(\alpha, \gamma)$ power-law model in [3] in which the number of vertices of degree k is $\lfloor \frac{e^\alpha}{k^\gamma} \rfloor$ where e^α is the normalization factor. For convenience, we shall refer to such a network as a $P(\alpha, \gamma)$ network.

We can deduce that the maximum degree in a $P(\alpha, \gamma)$ network is $e^{\frac{\alpha}{\gamma}}$ (since for $k > e^{\frac{\alpha}{\gamma}}$, the number of edges will be less than 1). The number of vertices and edges are

$$n = \sum_{k=1}^{e^{\frac{\alpha}{\gamma}}} \frac{e^\alpha}{k^\gamma} \approx \begin{cases} \zeta(\gamma) e^\alpha & \text{if } \gamma > 1 \\ \alpha e^\alpha & \text{if } \gamma = 1 \\ \frac{e^{\frac{\alpha}{\gamma}}}{1-\gamma} & \text{if } \gamma < 1 \end{cases},$$

$$m = \frac{1}{2} \sum_{k=1}^{e^{\frac{\alpha}{\gamma}}} k \frac{e^\alpha}{k^\gamma} \approx \begin{cases} \frac{1}{2}\zeta(\gamma-1) e^\alpha & \text{if } \gamma > 2 \\ \frac{1}{4}\alpha e^\alpha & \text{if } \gamma = 2 \\ \frac{1}{2}\frac{e^{\frac{2\alpha}{\gamma}}}{2-\gamma} & \text{if } \gamma < 2 \end{cases} \quad (3.1)$$

where $\zeta(\gamma) = \sum_{i=1}^{\infty} \frac{1}{i^\gamma}$ is the Riemann Zeta function [3] which converges for $\gamma > 1$ and diverges for all $\gamma \leq 1$. Without affecting the conclusion, we will simply use real numbers instead of rounding down to integers. The error terms are sufficiently small and can be bounded in our proofs.

While the scale of the network depends on α, the parameter γ decides the connection pattern and many other important characterizations of the network. For instance, the larger γ, the sparser and the more "power-law" the network is. Hence, the parameter γ is often regarded as the characteristic constant for scale-free networks.

3.2 Prohibitive Seeding Costs

We prove that the seeding must contain at least $\Omega(n)$ vertices if the propagation is locally bounded. The result is stated in the following theorem.

THEOREM 1. *Given a power-law network $G \in P(\alpha, \gamma)$, with $\gamma > 2$ and constant $0 < \rho < 1$, any d-seeding is of size at least $\Omega(n)$.*

PROOF. The proof consists of two parts. In the first part, we show that the volume i.e. the total degree of vertices, of any d-seeding must be $\Omega(m)$. In the second part, we prove that any subset of vertices $S \subset V$ with volume $\text{vol}(S) = \Omega(m)$ in a power-law network with power-law exponent $\gamma > 2$, will imply that $|S| = \Omega(n)$. Thus, the theorem follows.

In the first part, we consider two separate cases

Case $\rho > \frac{1}{2}$: Let $S = R_0$ be the optimal solution for the CFM problem on $G = (V, E)$, and $S = R_0, R_1, R_2, \ldots, R_d$ are vertices that become active at round $0, 1, 2, \ldots, d$, respectively (see Fig. 3). Notice that $\{R_i\}_{i=0}^d$ form a partition of V. Moreover, for each $1 \leq t \leq d$ the following inequality holds.

$$|\phi(R_t, \bigcup_{i=0}^{t-1} R_i)| \geq \frac{\rho}{1-\rho} \left(|\phi(R_t, \bigcup_{j=t+1}^{d} R_j)| + 2|\phi(R_t, R_t)| \right) \quad (3.2)$$

where $\phi(A, B)$ denotes the set of edges connecting one vertex in A to one vertex in B. The inequality means that at least a fraction $\frac{\rho}{1-\rho}$ among edges incident with the vertices activated in round t must be incident with active vertices in the previous rounds.

Sum up all inequalities in (3.2) for $t = 1..d$, we have

$$\sum_{t=1}^{d} |\phi(R_t, \bigcup_{i=0}^{t-1} R_i)| \geq \frac{\rho}{1-\rho} \sum_{t=1}^{d} \left(|\phi(R_t, \bigcup_{j=t+1}^{d} R_j)| + 2|\phi(R_t, R_t)| \right)$$

Eliminate the common factors in both sides, we have

$$\sum_{i=0}^{d-1} |\phi(R_i, \bigcup_{t=i+1}^{d} R_t)| \geq \frac{\rho}{1-\rho} \sum_{j=1}^{d-1} |\phi(R_j, \bigcup_{t=j+1}^{d} R_t)| + 2\sum_{t=1}^{d-1} |\phi(R_t, R_t)|$$

After some algebra, we obtain

$$\begin{aligned} \text{vol}(R_0) &\geq |\phi(R_0, \bigcup_{t=1}^{d} R_t)| \\ &\geq \frac{2\rho-1}{1-\rho} \sum_{j=1}^{d-1} |\phi(R_i, \bigcup_{t=j+1}^{d} R_t)| + 2\sum_{t=1}^{d} |\phi(R_t, R_t)| \\ &\Leftrightarrow \frac{\rho}{1-\rho} |\phi(R_0, V)| - |\phi(R_0, R_0)| \\ &\geq \frac{2\rho-1}{1-\rho} |E| + \frac{3-4\rho}{1-\rho} \sum_{t=1}^{d} |\phi(R_t, R_t)| \end{aligned} \quad (3.3)$$

Hence, when $\rho > 1/2$, $\text{vol}(R_0) \geq \frac{2\rho-1}{1-\rho}|E| = \Omega(m)$ for any d-seeding R_0.

Case $\rho \leq \frac{1}{2}$: We say that an edge is active if it is incident to at least one active vertex. At round $t = 0$, there are at most $\text{vol}(R_0)$ active edges, those who are incident to R_0. Eq. 3.2 implies that the number of active edges in each round increases at most ρ^{-1} times. After d rounds, the number of active edges will be bounded by $\text{vol}(R_0) \times \rho^{-d}$. Since, all edges are active at the end we have the inequality:

$$\text{vol}(R_0) \geq \rho^{-d}|E|.$$

In the second part of the proof, we show that if a subset $S \subset V$ has $\text{vol}(S) = \Omega(m)$, then $|S| = \Omega(n)$ whenever the power-law exponent $\gamma > 2$. Assume that $\text{vol}(S) \geq cm$, for some positive constant c. The size of S is minimum when S contains only the highest degree vertices of V. Let k_0 be the minimum degree of vertices in S in that extreme case, by Eq. 3.1 we have

$$cm = \frac{c}{2} \sum_{k=1}^{e^{\frac{\alpha}{\gamma}}} k\frac{e^\alpha}{k^\gamma} \leq \text{vol}(S) \leq \frac{1}{2} \sum_{k=k_0}^{e^{\frac{\alpha}{\gamma}}} k\frac{e^\alpha}{k^\gamma}$$

Simplify two sides, we have

$$\sum_{k=1}^{k_0-1} \frac{1}{k^{\gamma-1}} \leq (1-c) \sum_{k=1}^{e^{\frac{\alpha}{\gamma}}} \frac{1}{k^{\gamma-1}} = (1-c)\zeta(\gamma-1)$$

Since, the zeta function $\zeta(\gamma-1)$ converges for $\gamma > 2$, there exists a constant $k_{\rho,\gamma}$ that depends only on ρ and γ that satisfies

$$\sum_{k=1}^{k_{\rho,\gamma}} \frac{1}{k^{\gamma-1}} > (1-c)\zeta(\gamma-1)$$

Obviously, we have $k_0 \leq k_{\rho,\gamma}$. Thus, the number of vertices that are in S is at least

$$\sum_{k=k_{\rho,\gamma}}^{e^{\frac{\alpha}{\gamma}}} \frac{e^\alpha}{k^\gamma} = (1 - \sum_{k=1}^{k_{\rho,\gamma}} \frac{1}{k^\gamma})n = \Omega(n)$$

We have the last step because the sum $\sum_{k=1}^{k_{\rho,\gamma}} \frac{1}{k^\gamma}$ is bounded by a constant since $k_{\rho,\gamma}$ is a constant. □

In both cases $\rho > 1/2$ and $\rho \leq 1/2$, the size of a d-seeding set is at least $\Omega(n)$. However, we can see a clear difference in the propagation speed with respect to d between two cases. When $\rho < 1/2$, the number of active edges can increase exponentially (but is still bounded if d is a constant) and, it is likely that the number of active vertices also exponentially increases. In contrast, when $\rho > 1/2$, exploding in the number of active edges (and hence active vertices) is impossible as the volume of the d-seeding is tied to the number of edges m by a fixed constant $\frac{2\rho-1}{1-\rho}$, regardless of the value of d.

4. COST-EFFECTIVE & EXPEDITIOUS SOCIAL MARKETING ALGORITHM

In order to understand the influence propagation when the number of propagation hops is bounded, we propose VirAds, an efficient algorithm for the CFM problem. With the huge magnitude of OSN users and data available on OSNs, scalability becomes the major problem in designing algorithm for CFM. VirAds is scalable to network of hundred of millions links and provides high quality solutions in our experiments.

Before presenting VirAds, we consider a natural greedy for the CFM problem in which the vertex that can activate the most number of inactive vertices within d hops is selected in each step. This greedy is unlikely to perform well on practice for following two reasons. First, at early steps, when not many vertices are selected, every vertex is likely to activate only itself after being chosen as a seed. Thus, the algorithm cannot distinguish between good and bad seeds. Second, the algorithm suffers serious scalability problems. To select a vertex, the algorithm has to evaluate for each vertex v how many vertices will be activated after adding v to the seeding, e.g. by invoking an $O(m+n)$ Breadth-First Search procedure rooted at v. In the worst-case when $O(n)$ vertices are needed to evaluate, this alone can take $O(n(m+n))$. Moreover, as shown in the previous section, the seeding size can be easily $\Omega(n)$; thus, the worst-case running time of the naive greedy algorithm is $O(n^2(m+n))$, which is prohibitive for large-scale networks.

As shown in Algorithm 1, our VirAds algorithm overcomes the mentioned problems in the naive greedy by favoring the vertex which can activate the most number of *edges* (indeed, it also considers the number of active neighbor around each vertex). This avoids the first problem of the naive greedy algorithm. At early steps, the algorithm behaves similar to the degree-based heuristics that favors vertices with high degree. However, when a certain number of vertices are selected, VirAds will make the selection based on the information within d-hop neighbor around the considered vertices rather than only one-hop neighbor as in the degree-based heuristic.

The scalability problem is tackled in VirAds by efficiently keeping track of the following measures for each vertex v.

- $\mathbf{r_v}$: the round in which v is activated
- $\mathbf{n_v^{(e)}}$: The number of new active edges after adding v into the seeding
- $\mathbf{n_v^{(a)}}$: The number of extra active neighbors v needs in order to activate v

Algorithm 1: VirAds - Viral Advertising in OSNs

Input: Graph $G = (V, E)$, $0 < \rho < 1$, $d \in \mathbb{N}^+$
Output: A small d-seeding
$n_v^{(e)} \leftarrow d(v), n_v^{(a)} \leftarrow \rho \cdot d(v), r_v \leftarrow d+1, v \in V$;
$r_v^{(i)} = 0, i = 0..d, P \leftarrow \emptyset$;
while *there exist inactive vertices* **do**
  repeat
    $u \leftarrow \text{argmax}_{v \notin P} \{n_v^{(e)} + n_v^{(a)}\}$;
    Recompute $n_v^{(e)}$ as the number of new active edges after adding u.
  until $u = argmax_{v \notin P} \{n_v^{(e)} + n_v^{(a)}\}$;
  $P \leftarrow P \cup \{u\}$;
  Initialize a queue: $Q \leftarrow \{(u, r_v)\}$;
  $r_u \leftarrow 0$;
  foreach $x \in N(u)$ **do**
    $n_x^{(a)} \leftarrow \max\{n_x^{(a)} - 1, 0\}$;
  while $Q \neq \emptyset$ **do**
    $(t, \tilde{r}_t) \leftarrow Q.pop()$;
    foreach $w \in N(t)$ **do**
      foreach $i = r_t$ *to* $\min\{\tilde{r}_t - 1, r_w - 2\}$ **do**
        $r_w^{(i)} = r_w^{(i)} + 1$;
        if $(r_w^{(i)} \geq \rho \cdot d_w) \wedge (r_w \geq d) \wedge (i + 1 < d)$ **then**
          foreach $x \in N(w)$ **do**
            $n_x^{(a)} \leftarrow \max\{n_x^{(a)} - 1, 0\}$;
          $r_w = i + 1$;
          if $w \notin Q$ **then**
            $Q.push((w, r_w))$;
  Output P;

- $\mathbf{r_v^{(i)}}$: The number of activated neighbors of v up to round i where $i = 1..d$.

Given those measures, VirAds selects in each step the vertex u with the highest *effectiveness* which is defined as $n_u^{(e)} + n_u^{(a)}$. After that, the algorithm needs to update the measures for all the remaining vertices.

Except for $n_v^{(e)}$, we show that all other measures can be effectively kept track of in only $O((m+n)d)$ during the whole algorithm. When a vertex u is selected, it causes a chain-reaction and activate a sequence of vertices or lower the rounds in which vertices are activated. New activated vertices together with their active rounds are successively pushed into the queue Q for further updating much like what happens in the Bellman-Ford shortest-paths algorithm. Everytime we pop a vertex v from Q, if r_v, the current active round of v, is different from $\tilde{r}_v$, the active round of v when v is pushed into Q, we update for each neighbor w of v the values of r_w and $r_w^{(i)}$. If any neighbor w of v changes its active round and w is not in Q, we push w into Q for further update. The update process stops when Q is empty. Note that for each node $u \in V$, changing of r_u can cause at most d update for $r_w^{(.)}$ where w is a neighbor of u. For all neighbors of u, the total number of update is, hence, $O(d \cdot d(u))$. Thus, the total time for updating $r_w^{(.)}$ $\forall w \in V$ in VirAds will be at most $O((m+n) \cdot d)$.

To maintain $n_v^{(e))}$, the easiest approach is to recompute all $n_v^{(e)}$. This approach, called *Exhaustive Update*, is extremely time-consuming as discussed in the naive greedy. Instead, we only update $n_v^{(e)}$ when "necessary". In details, vertices are stored in a max priority queue in which the priority is their *effectiveness*. In each step, the vertex u with the highest effectiveness is extracted and $n_u^{(e)}$ is recomputed. If after updating, u still has the highest effectiveness, u is then selected. Otherwise, u is pushed back to the priority queue, and the new vertex with the highest effectiveness is considered, and so on.

Approximation Ratio for Power-law Networks. The CFM problem can be easily shown to be NP-hard by a reduction from the set cover problem. Thus, we are left with two choices: designing heuristics which have no worst-case performance guarantees or designing approximation algorithms which can guarantee the produced solutions are within a certain factor from the optimal. Formally, a β- approximation algorithm for a minimization (maximization) problem always returns solutions that are at most β times larger (smaller) than an optimal solution.

Unfortunately, there is unlikely an approximation algorithm with factor less than $O(\log n)$ as shown in next section. However, if we assume the network is power-law, our VirAds is an approximation algorithm for CFM with a constant factor.

THEOREM 2. *In power-law networks, VirAds is an $O(1)$ approximation algorithm for the CFM problem for bounded value of d.*

The theorem follows directly from the result in previous section that the optimal solution has size at least $\Omega(n)$ in power-law networks. Thus, the ratio between the VirAds's solution and the optimal solution is bounded by a constant.

5. HARDNESS OF IDENTIFYING SEEDING WITH GUARANTEES

This section provides the hardness of approximating the optimal solutions of the CFM problem, the impossibility of finding near-optimal solutions in polynomial time. In previous Section, we can obtain $O(1)$ approximation algorithms for CFM when the network is power-law. However, without the power-law assumption, there is no algorithm that can approximate the problem within a factor less than $O(\log n)$. We first prove the hardness for the case when $d = 1$, which is an essential step in proving the hardness for the general case $d \geq 1$.

5.1 One-hop CFM

We prove that the CFM problem cannot be approximated within a factor $\ln \Delta - O(\ln \ln \Delta)$ in graphs of maximum degree Δ, unless P=NP. The proof uses a gap-reduction from an instance of the *Bounded Set Cover* problem (SC_B) to an instance of CFM problem whose degrees are bounded by $B' = B$ poly log B. For background on hardness of approximation and gap-reduction we refer to reference [6].

DEFINITION 2 (BOUNDED SET COVER). *Given a set system $(\mathcal{U}, \mathcal{S})$, where $\mathcal{U} = \{e_1, e_2, \ldots, e_{n_s}\}$ is a universe and $\mathcal{S}$ is a collection of subsets of $\mathcal{U}$. Each subset in $\mathcal{S}$ has at most B elements and each element belongs to at most B subsets, for a predefined constant $B > 0$. A cover is a subfamily*

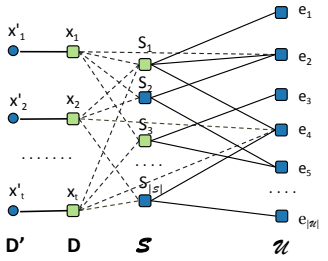


Figure 1: Reduction from SC_B to CFM when $d = 1$

$\mathcal{C} \subseteq \mathcal{S}$ of sets whose union is $\mathcal{U}$. Find a cover which uses the minimum number of subsets.

We state the tight inapproximability result for the bounded set cover by Trevisan [24] in the following lemma.

LEMMA 1. *There exist constants $B_0, c_0 > 0$ such that for every $B \geq B_0$ it is NP-hard to approximate the SC_B problem within a factor of $\ln B - c_0 \ln \ln B$.*

The proof in [24] reduces an instance of $GAP - SAT_{1,\gamma}$ of size n_S to an instance $\mathcal{F} = (\mathcal{U}, \mathcal{S})$ of SC_B by settings parameters l, m in Feige's construction [13] to be $\theta(\ln \ln B)$ and $\frac{B}{\text{poly } \log(B)}$, respectively. Denote by Δ_S the maximum cardinality of sets, and by f the maximum frequency of elements in $\mathcal{U}$, we have

- $|\mathcal{U}| = mn_S^l$ poly $\log B, |\mathcal{S}| = n_S^l$ poly log B
- $\Delta_S \leq B, f \leq$ poly log B for sufficient large B.

SC_B-CFM reduction. For each instance $\mathcal{F} = (\mathcal{U}, \mathcal{S})$ of SC_B, we construct a graph $\mathcal{H} = (V, E)$ as follows (Fig. 1):

- Construct a bipartite graph with the vertex set $\mathcal{U} \cup \mathcal{S}$ and edges between S and all elements $e_i \in S$, for each $S \in \mathcal{S}$.
- Add a set D consisting of t vertices and a set D' with same number of vertices, say $D = \{x_1, x_2, \ldots, x_t\}$ and $D' = \{x'_1, x'_2, \ldots, x'_t\}$, where $t = \frac{|\mathcal{U}|}{B \ln^2 B}$.
- Connect x_i to $x'_i, \forall i = 1 \ldots t$. This enforces the selection of x_i in the optimal CFM.
- Connect each vertex $e_j \in \mathcal{U}$ to $\lceil \frac{\rho}{1-\rho} f(e_j) \rceil - 1$ and each vertex $S_k \in \mathcal{S}$ to $\lceil \frac{\rho}{1-\rho} |S_k| \rceil$ vertices in D, where $f(e_j)$ is the frequency of element e_j. During the connection, we balance the degrees of vertices in D.

We can assume w.l.o.g. that optimal solutions of CFM contains all vertices in D but not ones in D'. Then, all vertices in $\mathcal{S}$ will be activated after the first round, and the a vertex in $\mathcal{U}$ is activated if and only if one of its neighbors in $\mathcal{S}$ is selected into the solution. Thus, the following lemma holds.

LEMMA 2. *The size difference between the optimal CFM of $\mathcal{H}$ and the optimal SC_B of $\mathcal{F}$ is exactly the cardinality of D, i.e., $OPT_{CFM}(\mathcal{H}) = OPT_{SC}(\mathcal{F}) + t$.*

The key to preserve the hardness ratio is to keep the degree of vertices in $\mathcal{H}$ bounded and the gap between the optimal solutions' sizes small.

LEMMA 3. *If $t = \frac{|\mathcal{U}|}{B \ln^2 B}$, then the maximum degree of vertices in $\mathcal{H}$ will be $B' = \Delta(\mathcal{H}) = O(B \text{ poly log } B)$.*

PROOF. We can verify that vertices in $\mathcal{S}$ and $\mathcal{U}$ have degree $O(B)$. Vertices in D have degrees at most $\frac{\text{vol}(D)}{t} + 1$, where $\text{vol}(D)$ is the total degree of vertices in D. Define $\phi(X, Y)$ as the set of edges crossing between two vertex subsets X and Y. We have

$$\begin{aligned}
&\text{vol}(D) = |\phi(D, D')| + |\phi(D, \mathcal{U})| + |\phi(D, \mathcal{S})| \\
&= |D| + \sum_{S_k \in \mathcal{S}} \lceil \frac{\rho}{1-\rho} |S_k| \rceil + \sum_{e_j \in \mathcal{U}} \lceil \frac{\rho}{1-\rho} f(e_j) - 1 \rceil \\
&\leq \frac{2\rho}{1-\rho} |\mathcal{S}| B + |\mathcal{S}| + t = \left(\frac{2\rho}{1-\rho} B + 1 \right) |\mathcal{S}| + t \qquad (5.1)
\end{aligned}$$

We have used the facts that $\sum_{S_k \in \mathcal{S}} |S_k| = \sum_{e_j \in \mathcal{U}} f(e_j)$ and $|S_k| \leq B, \ \forall S_k \in \mathcal{S}$.

Thus,

$$\begin{aligned}
B' &\leq \frac{1}{t} \left(\left(\frac{2\rho}{1-\rho} B + 1 \right) |\mathcal{S}| + t \right) + 1 \\
&\leq \left(\frac{2\rho}{1-\rho} B + 1 \right) \frac{B \ln^2 B \ n_S^l \text{ poly log } B}{m n^l \text{ poly log } B} \\
&\leq O(B \text{ poly log } B) \qquad (5.2)
\end{aligned}$$

This completes the proof. □

THEOREM 3. *When $d = 1$, it is NP-hard to approximate the CFM problem in graphs with degrees bounded by B' within a factor of $\ln B' - c_1 \ln \ln B'$, for some constant $c_1 > 0$.*

PROOF. We prove by contradiction. Assume there exists algorithm $\mathcal{A}$ to find in graph with degrees bounded by B' and $d = 1$ a CFM of size at most $(\ln B' - c_1 \ln \ln B')\mathrm{OPT}_{CFM}$, where OPT_{CFM} is the size of an optimal CFM. Let $\mathcal{F} = (\mathcal{U}, \mathcal{S})$ be an instance of SC_B with the optimal solution of size OPT_{SC}. Construct an instance $\mathcal{H}$ of CFM problem using the reduction SC_B-CFM as shown above. From (5.2), there exists constant $\beta > 0$ so that $B' \leq B \ln^\beta B$. Using algorithm $\mathcal{A}$ on $\mathcal{H}$, we obtain a solution of size at most $(\ln B' - c_1 \ln \ln B')\mathrm{OPT}_{CFM}$. We can then convert that to a solution of SC_B by excluding vertices in D (see Lemma 2) and obtain a set cover of size at most

$$(\ln B' - c_1 \ln \ln B')(\mathrm{OPT}_{SC} + t) - t \qquad (5.3)$$

Since each set in $\mathcal{S}$ can cover at most B elements, we have $\mathrm{OPT}_{SC} \geq \frac{|\mathcal{U}|}{B} = \frac{tB \ln^2 B}{B}$, thus $t \leq \frac{\mathrm{OPT}_{SC}}{\ln^2 B}$. If we select $c_1 = c_0 + \beta + 1$, the solution of SC_B is then, after some algebra, at most $(\ln B - c_0 \ln \ln B)\mathrm{OPT}_{SC}$ that contradicts the Lemma 1. □

Similarly, with appropriate setting in Feige's construction [13], we obtain the following hardness result regarding the network size n (the proof detail can be found in the technical report on our website).

THEOREM 4. *For any $\epsilon > 0$, the CFM problem, when $d = 1$, cannot be approximated within a factor $(\frac{1}{2} - \epsilon) \ln n$, unless NP $\subset$ DTIME$(n^{O(\log \log n)})$.*

Note that Theorems 3 and 4 are incomparable in general. Let Δ be the maximum degree, Theorem 3 implies the hardness of approximation with factor $(1-\epsilon) \ln \Delta$, which is larger

than $(\frac{1}{2} - n) \ln n$ if $\Delta \approx n$, but smaller when $\Delta < \sqrt{n}$, for example in power-law graphs with the exponent $\gamma > 2$. In addition, the Theorem 4 uses a stronger assumption than that in Theorem 3.

5.2 Multiple-hop CFM

We now present a gap reduction from the CFM problem to the one-hop CFM problem with $d \geq 2$. The hardness result follows immediately by the Theorem 3 in the previous section.

Given a graph $G = (V, E)$ as an instance of the CFM problem. We will construct an instance $G' = (V', E')$ of the CFM problem as follows (and as illustrated in Fig. 3). We

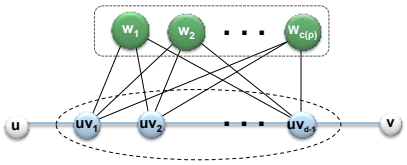


Figure 2: The transmitter gadget.

add $c(\rho)$ vertices $w_1, w_2, \ldots, w_{c(\rho)}$, called flashpoints, where $c(\rho) = \min\{t \in \mathbb{N} \mid \frac{t-1}{t+1} \leq \rho < \frac{t}{t+1}\}$. These vertices will be selected at the beginning to kick off the activation of other nodes. Furthermore, each "flashpoint" w_p is connected to a dummy vertex z_p.

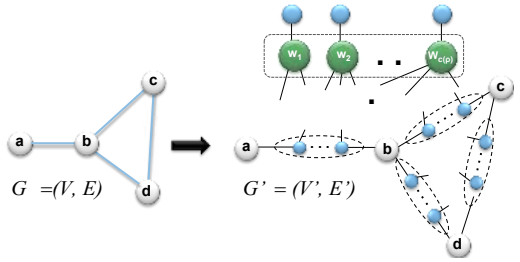


Figure 3: Gap-reduction from one-hop CFM to d-hop CFM.

Replace each edge $(u, v) \in E$ by a gadget called transmitter. The transmitter connecting vertex u and v is a chain of $d-1$ path, named uv_1 to uv_{d-1}. The vertex u is connected to uv_1, uv_1 is connected to uv_2 and so on, vertex uv_{d-1} is connected to v. Each vertex uv_i, $i = 1..d-1$ is connected to all flashpoints. An example for transmitter is shown in Fig. 2. The transmitter is designed so that if all flashpoints and vertex u are selected at the beginning, then vertex uv_{d-1} will be activated after $d-1$ rounds. Hence, the number of activated neighbors of v after $d-1$ rounds will equal the number of selected neighbors of v in the original graph.

Finally, we replace each edge (w_p, z_p) by a transmitter. In order to activate all dummy vertices z_p after d rounds, we can assume, w.l.o.g., that all flashpoints must be selected in an optimal solution. The following lemma follows directly from the construction.

LEMMA 4. *Every solution of size k for the one-hop ($d = 1$) CFM problem in G induces a solution of size $k + c(\rho)$ for the d-hop CFM problem in G'.*

On another direction, we also have the following lemma.

LEMMA 5. *An optimal solution of size k' for the d-hop CFM problem induces a size $k' - c(\rho)$ solution for the one-hop CFM problem in G.*

PROOF. For a transmitter connecting u to v, if the solution of the d-hop CFM problem contains any of the intermediate vertices $uv_1, \ldots, uv_{d-1}$, we can replace that vertex in the solution with either u or v to obtain a new solution of same size (or less). Hence, we can assume, w.l.o.g., that none of the intermediate vertices are selected. Therefore, all flashpoints must be selected in order to activate the dummy vertices. It is easy to see that the solution of d-hop CFM excluding the flashpoints will be a solution of one-hop CFM in G with size $k' - c(\rho)$. □

Note that the number of vertices in G' is upper-bounded by dn^2 i.e. $\ln|V'| < 2ln|V| + lnd$. Thus, using the same arguments used in the proof of Theorem 4, we can show that a $(\frac{1}{4} - \epsilon) \ln n$ approximation algorithm algorithm lead to a $(\frac{1}{2} - \epsilon) \ln n$ approximation algorithm for the one-hop CFM problem (contradicts Theorem 4).

THEOREM 5. *The CFM problem cannot be approximated within $(\frac{1}{4} - \epsilon) \log n$ for $d \geq 1$, unless NP $\subset$ DTIME($n^{O(\log \log n)}$)*

6. EMPIRICAL STUDY

In this section we perform experiments on OSNs to show the efficiency of our algorithms in comparison with simple degree centrality heuristic and study the trade-off between the number of times the information is allowed to propagate in the network and the seeding size.

6.1 Comparing to Optimal Seeding

One advantage of our discrete diffusion model over probabilistic ones [16, 17] is that the exact solution can be found using mathematical programming. This enables us to study the exact behavior of the seeding size when the number of propagation hop varies.

We formulate the CFM problem as an $0-1$ Integer Linear Programming (ILP) problem below.

$$\text{minimize} \sum_{v \in V} x_v^0 \tag{6.1}$$

$$\text{subject to} \sum_{v \in V} x_v^d \geq |V| \tag{6.2}$$

$$\sum_{w \in N(v)} x_w^{i-1} + \lceil \rho \cdot d(v) \rceil x_v^{i-1} \geq \lceil \rho \cdot d(v) \rceil \; x_v^i$$

$$\forall v \in V, i = 1..d \tag{6.3}$$

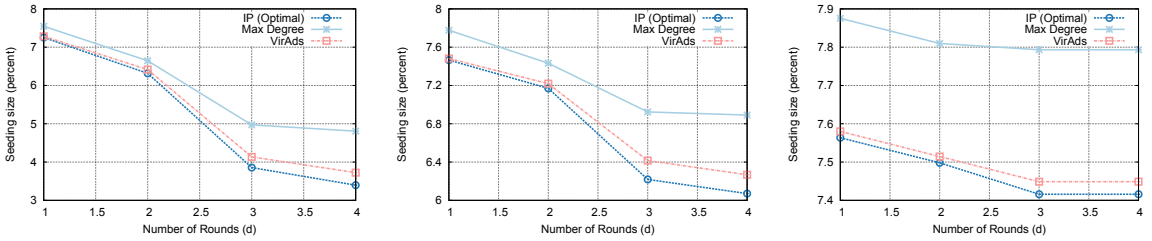
$$x_v^i \geq x_v^{i-1} \qquad \forall v \in V, i = 1..d \tag{6.4}$$

$$x_v^i \in \{0, 1\} \qquad \forall v \in V, i = 0..d \tag{6.5}$$

where $x_v^i = \begin{cases} 0 & \text{if v is inactive at round } i \\ 1 & \text{otherwise} \end{cases}$.

The objective of the ILP is to select a minimum number of seeds at the beginning. The constraint (2) guarantees all nodes are activated at the end, while (3) deals with propagation condition; the constraint (4) is simply to keep vertices active once they are activated.

We solve the ILP problem on Erdos collaboration networks, the social network of famous mathematician, [8]. The network consists of 6100 vertices and 15030 edges. The ILP



(a) $\rho = 0.4$ (b) $\rho = 0.6$ (c) $\rho = 0.8$

Figure 4: Seeding size (in percent) on Erdos's Collaboration network. VirAds produces close to the optimal seeding in only fractions of a second (in comparison to 2 days running time of the IP(optimal))

is solved with the optimization package GUROBI 4.5 on Intel Xeon 2.93 Ghz PC and setting the time limit for the solver to be 2 days. The running time of the IP solver increases significantly when d increases. For $d = 1, 2$, and 3, the solver return the optimal solutions. However, for $d = 4$, the solver cannot find the optimal solutions within the time limit and returns sub-optimal solutions with relative errors at most 15%.

The optimal (or sub-optimal) seeding sizes are shown in Figs. 4a, 4b, and 4c for $\rho = 0.4, 0.6$ and 0.8, respectively. VirAds provides close-to-optimal solutions and performs much better Max Degree. Especially, when $\rho = 0.8$ the VirAds's seeding is only different with the optimal solutions by one or two nodes. In addition, VirAds only takes fractions of a second to generate the solutions.

As proven in Section 3, the seeding takes a constant fraction of nodes in the network. For Erdos Colloboration Network, the seeding consists of 3.8% to 7% the number of nodes in the networks. Further, the seeding can consist as high as 20% to 40% nodes in the network for larger social networks in next section.

Although the mathematical approach can provide accurate measurement on the optimal seeding size, it cannot be applied for larger networks. The rest of our experiments measures the quality and scalability of our proposed algorithm VirAds on a collection of large networks.

6.2 Large Social Networks

We select networks of various sizes including Coauthors network in Physics sections of the e-print arXiv[16], Facebook[25] and Orkut[20], a social networking run by Google. Links in all three networks are undirected and unweighted. The sizes of the networks are presented in Table 1.

Table 1: Sizes of the investigated networks

	Physics	**Facebook**	**Orkut**
Vertices	37,154	90,269	3,072,441
Edges	231,584	3,646,662	223,534,301
Avg. Degree	12.5	80.8	145.5

Physics: We shall refer the physics coauthors network as Physics network or simply Physics. Each node in the network represents an author and there is an edge between two authors if they coauthor one or more papers. *Facebook* dataset consists 52% of the users in the New Orleans [25]. *Orkut* dataset is collected by performing crawling in last 2006 [20]. It contains about 11.3% of Orkut's users.

6.3 Solution Quality in Large Social Networks

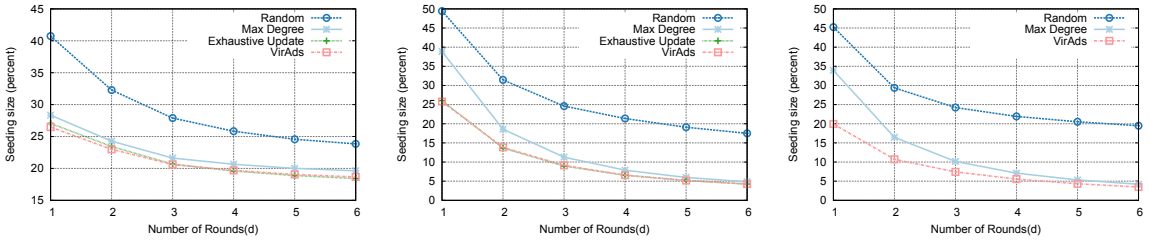
We compare our VirAds algorithm with the following heuristics *Random* method in which vertices are picked up randomly until forming a d-seeding and *Max Degree* method in which vertices with highest degree are selected until forming a d-hop seeding. Finally, we compare VirAds with its naive implementation, called *Exhaustive Update*, in which after selecting a vertex into the seeding, the effectiveness of all the remaining vertices are recalculated. With more accurate estimation on vertex effectiveness, Exhaustive Search is expected to produce higher quality solutions than those of VirAds.

The seeding size with different number of propagation hop d when $\rho = 0.3$ are shown in Fig. 5. To our surprise, VirAds even performs equal or better than *Exhaustive Update* despite that it uses significantly less effort to update vertex effectiveness. VirAds has smaller seeding in Physics than *Exhaustive Update*; both of them give similar results for Faceboook; while *Exhaustive Update* cannot finish on Orkut after 48 hours and was forced to terminate. Sparingly update the vertices' effectiveness turns out to be efficient enough since the influence propagation is locally bounded. In addition, the seeds produced by VirAds are almost two times smaller than those of *Random*.

The gap between VirAds and Max Degree is narrowed when the number of maximum hops increases. Hence, selecting nodes with high degrees as seeding is a good long-term strategy, but might not be efficient for fast propagation when the number of hops is limited. In Facebook and Orkut, when $d = 1$, *Max Degree* has 60% to 70% more vertices in the seeding than *VirAds*. In Physics, the gap between VirAds and the *Max Degree* is less impressive. Nevertheless, VirAds consistently produces the best solutions in all networks.

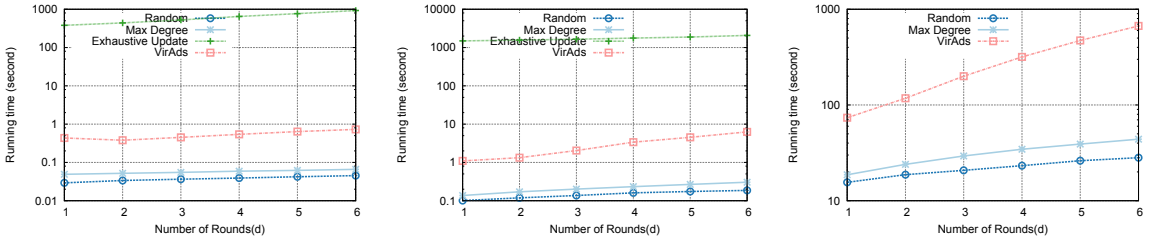
6.4 Scalability

The running time of all methods at different propagation hop d are presented in Fig 6. The time is measured in second and presented in the log scale. The running times increase slightly together with the number of propagation rounds d, and are proportional to the size of the network. The *Exhaustive Update* has the worst running time, taking up to 15 minutes for Physics, 20 minutes for Facebook. For Orkut, the algorithm cannot finish within 2 days, as mentioned. The three remaining algorithms *VirAds*, *Max Degree*, and *Ran-*



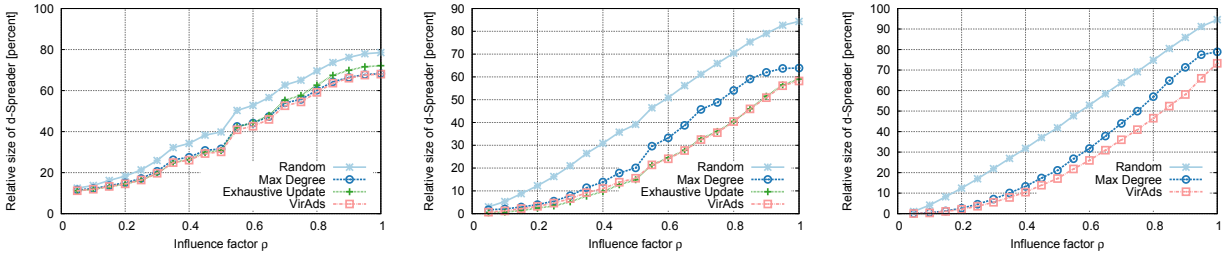
(a) Physics (b) Facebook (c) Orkut

Figure 5: Seeding size when the number of propagation hop d varies ($\rho = 0.3$). VirAds consistently has the best performance.



(a) Physics (b) Facebook (c) Orkut

Figure 6: Running time when the number of propagation hop d varies ($\rho = 0.3$). Even for the largest network of 110 million edges, VirAds takes less than 12 minutes.



(a) Physics (b) Facebook (c) Orkut

Figure 7: Seeding size at different influence factors ρ (the maximum number of propagation hops is $d = 4$).

dom take less than one second for Physics, and less than 10 seconds for Facebook. Even on the largest network Orkut with more than 220 million edges, VirAds requires less than 12 minutes to complete.

6.5 Influence factor

We study the performance of VirAds and the other method at different influence factor ρ. The number of propagation rounds d is fixed to 4. The size of d-seeding sets are shown in Figures 7. VirAds is clearly still the best performer. The seeding sizes of VirAds are up to 5 times smaller than those of Max Degree for small ρ (although it's hard to see this on the charts due to small seeding sizes).

Since all tested networks are social networks with small diameter, the seeding sizes go to zero when ρ is close to zero. The exception is the Physics, in which the seeding sizes do not go below 10% the number of vertices in the networks even when $\rho = 0.05$. A closer look into the Physics network reveals that the network contain many isolated cliques of small sizes (2, 3, 4, and so on) which correspond to authors that appear in only one paper. In each clique, regardless of the threshold ρ, at least one vertex must be selected, thus the seeding size cannot get below the number of isolated cliques in the networks. To eliminate the effect of isolated cliques, a possible approach is to restrict the problem to the largest component in the network.

7. CONCLUSIONS

We present the first work that explores the time aspect of influence propagation in social networks. We demonstrate that massively advertising involves costly seeding when imposing the limit on the propagation. Because of the power-law degree distribution observed in social networks, the seed-

ing might involve a constant fraction of nodes in the networks, which is prohibitive for large networks. The old strategy for viral marketing that targets nodes with high degree in the network might be no longer suitable when we need the influence to propagate quickly throughout the network. Instead, an optimization-based solution such as VirAds is more suitable to discover a low-cost set of influential users.

8. ACKNOWLEDGEMENT

This work is partially supported by the DTRA YIP grant number HDTRA1-09-1-0061 and the NSF CAREER Award number 0953284.

9. REFERENCES

[1] Alexa 2010. `http://www.alexa.com/topsites`.

[2] Facebook statistics 2010. `http://www.facebook.com/press/info.php?statistics`.

[3] W. Aiello, F. Chung, and L. Lu. A random graph model for massive graphs. In *STOC '00*, New York, NY, USA, 2000. ACM.

[4] W. Aiello, F. Chung, and L. Lu. A random graph model for power law graphs. *Experimental Math*, 10:53–66, 2000.

[5] W. Aiello, F. Chung, and L. Lu. Random evolution in massive graphs. In *In Handbook of Massive Data Sets*. Kluwer Academic Publishers, 2001.

[6] S. Arora and B. Barak. *Computational complexity: a modern approach*. Cambridge University Press, 2009.

[7] A. Barabasi, R. Albert, and H. Jeong. Scale-free characteristics of random networks: the topology of the world-wide web. *Physica A*, 281, 2000.

[8] A. Barabasi, H. Jeong, Z. Neda, E. Ravasz, A. Schubert, and T. Vicsek. Evolution of the social network of scientific collaborations. *Physica A: Statistical Mechanics and its Applications*, 311(3-4):590–614, 2002.

[9] M. Cha, A. Mislove, and K. P. Gummadi. A measurement-driven analysis of information propagation in the flickr social network. In *WWW '09*, pages 721–730, New York, NY, USA, 2009. ACM.

[10] N. Chen. On the approximability of influence in social networks. *SIAM Journal of Discrete Mathematics*, 23(3):1400–1415, 2009.

[11] W. Chen, Y. Wang, and S. Yang. Efficient influence maximization in social networks. In *KDD '09*, pages 199–208, New York, NY, USA, 2009. ACM.

[12] A. Clauset, C. R. Shalizi, and M. E. J. Newman. Power-law distributions in empirical data. *SIAM Reviews*, 2007.

[13] U. Feige. A threshold of ln n for approximating set cover. *Journal of ACM*, 45(4):634–652, 1998.

[14] A. Ferrante. Hardness and approximation algorithms of some graph problems, 2006.

[15] A. Goyal, F. Bonchi, and L. V. S. Lakshmanan. Learning influence probabilities in social networks. *WSDM '10*, pages 241–250, 2010.

[16] D. Kempe, J. Kleinberg, and É. Tardos. Maximizing the spread of influence through a social network. In *KDD'03*, pages 137–146. ACM New York, NY, USA, 2003.

[17] D. Kempe, J. Kleinberg, and E. Tardos. Influential nodes in a diffusion model for social networks. In *ICALP '05*, pages 1127–1138, 2005.

[18] J. Leskovec, L. A. Adamic, and B. A. Huberman. The dynamics of viral marketing. *ACM Trans. Web*, 1, 2007.

[19] J. Leskovec, A. Krause, C. Guestrin, C. Faloutsos, J. VanBriesen, and N. Glance. Cost-effective outbreak detection in networks. In *ACM KDD '07*, pages 420–429, New York, NY, USA, 2007. ACM.

[20] A. Mislove, M. Marcon, K. P. Gummadi, P. Druschel, and B. Bhattacharjee. Measurement and Analysis of Online Social Networks. In *IMC'07*, San Diego, CA, October 2007.

[21] D. Peleg. Local majority voting, small coalitions and controlling monopolies in graphs: A review. In *SIROCCO'96*, pages 152–169, 1996.

[22] L. Rao. Toyota Turns To Twitter To Repair Its Image. http://techcrunch.com/2010/03/02/toyota-turns-to-twitter-to-repair-its-image/, Mar. 2010.

[23] C. Shirky. The Political Power of Social Media: Technology, the Public Sphere, and Political Change, 2011.

[24] L. Trevisan. Non-approximability results for optimization problems on bounded degree instances. In *ACM STOC '01*, pages 453–461, New York, NY, USA, 2001. ACM.

[25] B. Viswanath, A. Mislove, M. Cha, and K. P. Gummadi. On the evolution of user interaction in facebook. In *WOSN'09*, August 2009.

[26] F. Wang, E. Camacho, and K. Xu. Positive influence dominating set in online social networks. In *COCOA '09*, pages 313–321, Berlin, Heidelberg, 2009. Springer-Verlag.

[27] F. Zou, Z. Zhang, and W. Wu. Latency-bounded minimum influential node selection in social networks. In B. Liu, A. Bestavros, D.-Z. Du, and J. Wang, editors, *WASA*, LNCS, pages 519–526, 2009.