Pay Few, Influence Most: Online Myopic Network Covering

K. Avrachenkov[1], P. Basu[2], G. Neglia[1], B. Ribeiro[3], and D. Towsley[4]

[1]INRIA
06902 Sophia Antipolis
France
{konstantin.avratchenkov,
giovanni.neglia}@inria.fr

[2]Raytheon BBN Technologies
Cambridge, MA
pbasu@bbn.com

[3]School of Computer Science
Carnegie Mellon University
5000 Forbes Ave
Pittsburgh, PA 15213
ribeiro@cs.cmu.edu

[4]School of Computer Science
UMass Amherst
140 Governors Drive
Amherst, MA 01003
towsley@cs.umass.edu

***Abstract*—Efficient marketing or awareness-raising campaigns seek to recruit a small number, w, of influential individuals – where w is the campaign budget – that are able to cover the largest possible target audience through their social connections. In this paper we assume that the topology is gradually discovered thanks to recruited individuals disclosing their social connections.**

We analyze the performance of a variety of online myopic algorithms (i.e. that do not have a priori information on the topology) currently used to sample and search large networks. We also propose a new greedy online algorithm, Maximum Expected Uncovered Degree (MEUD). Our proposed algorithm greedily maximizes the expected size of the cover, but it requires the degree distribution to be known. For a class of random power law networks we show that MEUD simplifies into a straightforward procedure, denoted as MOD because it requires only the knowledge of the Maximum Observed Degree.

I. Introduction

This paper addresses the need to efficiently select w individuals in a network such that they cover, through their neighbors, the largest possible fraction of the network. Online social networks have generated much attention as a breeding ground for new forms of social studies, social mobilization, and online campaigns. The recent 2012 U.S. presidential election, for instance, presents a real-life example on a social network setting. A candidate's Facebook app asked its subscribers to send get-out-to-vote reminders to their like-minded friends in swing states [1]. Thus, the effectiveness of a subscriber is measured by how many of his or her friends live in swing states.

Recruiting individuals from a population is no easy task. The recruitment of each individual comes at a cost in time, money, and social capital; and the total budget is often small with respect to the budget required to recruit all individuals in the population. Most of the works on network cover, e.g. [2], [3], [4], consider the social network topology to be known in advance, which is often not the case in the wild. In this work we look at the cover problem when the network topology is unknown. Following previous works in the literature, we assume that any individual in the network can be recruited, but in our case recruitments happen through friends recruiting friends.

Problem Formulation: We formulate the target subpopulation cover problem as a Maximum *Connected* Network Cover (MCNC) problem on an unknown connected undirected graph $G = (V, E)$, where V is the set of target individuals and E the set of individuals' mutual connections. The graph G has $n = |V|$ nodes, $m = |E|$ edges and degree distribution $\{p_k\}_{k=1,\dots n-1}$. Unless otherwise specified, we assume all graph parameters to be unknown. Denote $\mathcal{N}_a(v)$ the set of neighbors of node $v \in V$ and $k_v = |\mathcal{N}_a(v)|$ the degree of v. The volume of a set of nodes $U \subset V$ is defined as the sum of all the degrees of the nodes in U and is denoted as $\text{vol}(U)$.

Let w be a given campaign budget (say, funds to recruit individuals) and, to simplify our exposition, assume that each node has a unitary recruitment cost. Our main goal is to design efficient online algorithms to solve the following problem: determine a group of w individuals to be recruited in order to maximize the size of the covered subset, i.e. the set including the recruited nodes and their neighbors. Initially, the only available information is a single node sampled from the population. The algorithm progresses as each recruited node discloses its neighbors, increasing the number of nodes that can be recruited. It follows that the recruited nodes form a connected subgraph.

More formally, at each step $t = 1, \dots, w$, we classify the nodes in V into three disjoint sets. The set $\mathcal{B}(t)$ denotes the recruited nodes at step t. Unrecruited neighbors of recruited nodes are denoted as *observed nodes* and form the set $\mathcal{N}(\mathcal{B}(t)) = \cup_{v \in \mathcal{B}(t)} \mathcal{N}_a(v) \backslash \mathcal{B}(t)$. We say a node $v \in V$ is *covered* after t recruitments if $v \in \mathcal{B}(t) \cup \mathcal{N}(\mathcal{B}(t))$. The set of all *uncovered* nodes is denoted as $\mathcal{W}(t)$ ($= V - (\mathcal{B}(t) \cup \mathcal{N}(\mathcal{B}(t)))$) and the set of covered nodes as $\overline{\mathcal{W}}(t)$. The sizes of the three sets $\mathcal{B}(t)$, $\mathcal{N}(\mathcal{B}(t))$, and $\mathcal{W}(t)$ are t, $N(\mathcal{B}(t))$ and $W(t)$, respectively. The three different sets are shown in Fig. 1.

Corresponding author: ribeiro@cs.cmu.edu; **Acknowledgements**: This work was partially supported by NSF grant CNS-1065133, ARL Cooperative Agreement W911NF-09-2-0053, ADR "Network Science" of the Alcatel-Lucent Inria Joint Lab and by the European Commission within the framework of the CONGAS project FP7-ICT-2011-8-317672. The views and conclusions contained in this document are those of the author and should not be interpreted as representing the official policies, either expressed or implied of the NSF, ARL, or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation hereon.

978-1-4799-3088-3/14/$31.00 ©2014 IEEE

The online algorithm proceeds as follows: at step t, $1 < t \leq w$, the algorithm recruits node $v \in \mathcal{N}(\mathcal{B}(t-1))$ and performs the update $\mathcal{B}(t) = \mathcal{B}(t-1) \cup \{v\}$. The objective of the online algorithm is to maximize the size of the network cover set $\mathcal{B}(w) \cup \mathcal{N}(\mathcal{B}(w))$ without having a priori access to topology information. Note that for most of the algorithms considered in this paper, knowledge of the topology is limited to the recruited nodes and their connections. So, after t recruitments, the algorithm is aware of the covered nodes and all their neighbors, but it is unaware of the existence of nodes in $\mathcal{W}(t)$ and links between the nodes in $\mathcal{N}(\mathcal{B}(w))$.

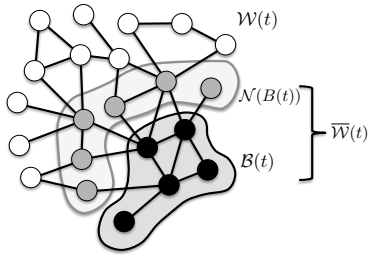


Fig. 1: Network sampling evolving sets.

Contributions: We make the following contributions:

(1) We thoroughly evaluate through extensive simulations on social network datasets.The performance of several known network sampling algorithms: Breadth-First Search (BFS), Depth-First Search (DFS) and Random Walk (RW). We observe that on social network topologies RW consistently outperforms (sometimes significantly) BFS and DFS. Moreover BFS performance exhibits a high variability as a function of network homophily and the choice of starting node. DFS performs the worst, because it tends to recruit nodes that have a surprisingly small number of neighbors. We also explain why this is the case.

(2) We propose a new online algorithm (MEUD, Maximum Expected Uncovered Degree) that greedily maximizes the expected size of the cover. Although MEUD requires knowing the degree distribution, we show that, for a broad class of power law networks, MEUD simplifies to the Maximum Observed Degree (MOD) algorithm that (like BFS, DFS, and RW) *does not* require degree distribution or network topology side information. We show through extensive simulations over a variety of social network datasets that MOD consistently outperforms (sometimes significantly) all other analyzed algorithms.

Due to space constraints most of our experimental results are described in the companion technical report [5].

II. Network Covers, Oracles, & Approximate Solutions

To simplify the following discussion we introduce the following time-dependent quantities for observed nodes (nodes in $\mathcal{N}(\mathcal{B}(t))$). The *observed degree* at time t of an observed node is the number of its neighbors that are recruited at time t. The *excess degree* of the node is the difference between its actual degree and its observed degree, i.e. it is the number of its neighbors that belong to $\mathcal{W}(t) \cup \mathcal{N}(\mathcal{B}(t))$. Finally, its *uncovered degree* is the number of its neighbors that belong to $\mathcal{W}(t)$. We observe that, being the graph knowledge at time t limited to the recruited nodes and their connections, the observed and excess degrees of an observed node correspond to the number of its links that are known and unknown, respectively. Its uncovered degree is equal to the increment in the number of covered nodes if the node is recruited at the following step.

The problem we study is closely related to the well-studied *Maximum Coverage* (MC) problem that is NP-hard [6]. The MC problem can be described as finding the cover of maximal size recruiting at most w nodes. Unlike our problem, MC assumes the topology to be known and the subset of recruited nodes is not required to be connected. A similar problem, the Minimum Dominating Set (MDS) problem aims to find a subset of nodes $D \subseteq V$ with the minimum cardinality such that all nodes in G are either in D or are neighbors to a node in D. The Minimum *Connected* Dominating Set (MCDS) problem imposes the additional restriction that the subgraph induced by the vertices in D has to be connected. Guha and Khuller [3] proposed an approximation algorithm for the MCDS problem, that corresponds to *growing* a tree T in an online fashion, starting from a single node and recruiting at each step the neighbor that has the largest *uncovered degree*. They showed that the above algorithm has a guaranteed approximation ratio $O(\Delta)$, where Δ is the maximum degree. In this paper we assume that the uncovered degree of an observed node is unknown. Only the observed degree is available. We compare the performance of the different algorithms with those of Guha and Khuller's algorithm. We refer to it as the "Oracle" because of its one-hop lookahead capability. As expected, the Oracle achieves better performance than other algorithms that do not use this additional information.

Our MEUD algorithm (described in Sec. V) is similar in the spirit to Guha and Khuller's Oracle, but it greedily recruits the node that maximize the *expected* uncovered degree instead of the actual uncovered degree, that we assume to be unknown. MEUD, however, requires the degree distribution of the network as side information ($\{p_k\}_{k=1,\ldots,n-1}$). In the absence of such information, we show that for a class of random power law networks, a natural myopic online greedy algorithm recruiting the node with the maximum observed degree approximates MEUD – this is our MOD algorithm. Expected value analysis as well as simulations in Sec. V show that MOD is a good heuristic when operating on realistic social networks, such as those obeying a power law degree distribution.

III. Datasets & Simulation Setup

We run experiments on 6 different social networks datasets (Enron, Slashdot, Wiki-talk, EmailEU, Youtube, Flickr) as well as three different types of "non-social" networks

(Gnutella [7], HepTh, Amazon). All these datasets, except Flickr [8], are available online at the SNAP repository [9]. Due to space constraints, in this paper we only show the results for the Enron email dataset (with $n = 36,692$, average degree $\langle k \rangle = 10.02$ and average clustering coefficient $c = 0.5$), Slashdot blog commentators' dataset ($n = 82{,}168$, $\langle k \rangle = 0.1$, $c = 0.23$), Wiki-talk ($n = 2{,}394{,}385$, $\langle k \rangle = 3.9$, $c = 0.2$) is Wikipedia user-to-user discussion graph, and the Flickr online photosharing network with $n = 1{,}715{,}255$ nodes and average degree $\langle k \rangle = 12.2$. The results of the remaining datasets are in the companion technical report [5].

Our metrics consist of averages over $1{,}000$ simulation runs. We use colored shaded regions (shadows) in our plots to show the value of standard deviation plotted around the average. In our simulations $\mathcal{B}(1)$ is initialized with a single node recruited uniformly at random from V. The order in which neighbors of a node appear on its list of neighbors is randomized from run to run to avoid arbitrary biases that may arise from the choice of node IDs in the dataset.

IV. BFS, DFS, AND RW COVERS

We begin our study by comparing the performance of two different approaches derived from basic graph traversal algorithms: **Breadth-First Search (BFS)** and **Depth-First Search (DFS)**. We will then discuss **Random Walks (RW)**. BFS is widely used in network sampling [8], [10], [11], [12].

In BFS and DFS the order in which nodes are recruited depends on the time they have been observed. When a new node is recruited, both algorithms check if each of its neighbors is already recruited or has already been observed. The difference is that BFS stores the observed nodes in a FIFO data structure while DFS in a LIFO one.

Fig. 2 shows the average cover size $\langle \overline{W}(t) \rangle$ of BFS and DFS as a function of the number of recruitments t on the Enron and on the Slashdot networks. We find similar results on all of our social network datasets, please refer to our technical report [5]. The simulations show that, while both BFS and DFS achieve the full coverage for $t \approx N$, BFS significantly outperforms DFS for all other values of t. To understand this difference, we qualitatively analyze these algorithms. The performance difference is due to the fact that larger degree nodes are discovered earlier (in a stochastic order sense) than lower degree ones, and then BFS recruits them earlier than DFS because it uses a FIFO data structure. Indeed, if we assume $t \ll N$, the probability $q_v(t)$ that node $v \in V$ is first observed (and then inserted in $\mathcal{N}(\mathcal{B}(t))$) at step t is approximately

$$q_v(t) \approx (\gamma_v/N)(1 - \gamma_v/N)^{t-1}, \tag{1}$$

where $\gamma_v = k_v/\langle k \rangle$ (this simple formula is a good approximation in a configuration model, where nodes are recruited independently by both algorithms). Thus, large degree nodes tend to be observed earlier in the process than small degree nodes. As BFS recruits the earliest observed nodes from $\mathcal{N}(\mathcal{B}(t))$, BFS tends to recruit large degree nodes first. On the other hand, DFS recruitment policy leaves the first observed nodes to be recruited last, effectively leaving large degree nodes to be recruited last.

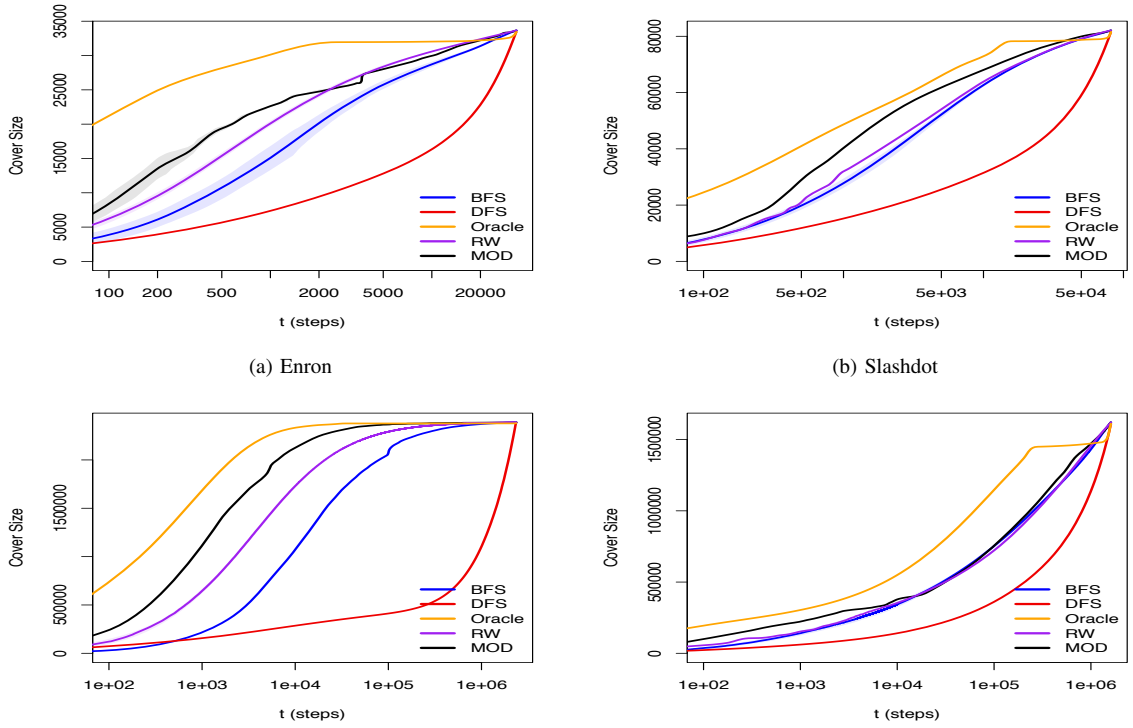
Figs. 2 consistently show larger standard deviations for BFS than for DFS. This is because the cover size of a non-negligible fraction of the BFS runs deviates from the average. This instability is due to the strong dependence of the BFS cover size on the initial node $\mathcal{B}(0) = \{i\}$. As BFS explores the network in "waves" (expanding rings from i), the initial node selection may significantly impact BFS's cover size. In highly clustered networks (where nodes tend to be connected in communities) we expect BFS to recruit nodes with significantly coverage overlap, reducing its performance.

Interestingly, our results contradict previous results in the literature [4], where is reported that DFS outperforms BFS (and most other algorithms). The reason behind the disparity between our conclusions and those of the previous works [4] is explored in detail in [5]. In a few words, these previous works considered mostly technological networks whose topologies differ significantly from social networks (more precisely, the HepTh citation network and Amazon "Customers Who Bought This Item Also Bought" product networks (see Sec. III for a brief description of these datasets). To understand this disparity, we reproduce the results in Maiya and Berger-Wolf [4] over the same datasets (see Figures 3(a) and 3(b)) and then perform another set of simulations now randomizing some of the network connections (see Figures 3(c) and 3(d)). Surprisingly, the randomized networks show results consistent to those of social networks presented in our work: for instance, that BFS significantly outperforms DFS. This points to the peculiar structure of the networks analyzed in Maiya and Berger-Wolf, which are not social networks, and that led to their results.

We now discuss the cover of Random Walks (RW), that have received increased attention as a tool for network sampling [16], [17], [18], [19], [20], [21] mostly due to their good statistical properties. In this paper we consider a basic Random Walk where the node visited by the walk at step $v+1$ is selected uniformly at random among the neighbors of the node visited at step v. All the nodes visited by the walk are recruited. The RW shares commonalities with DFS in that it also traverses the graph from the last recruited node. Despite of this similarity, it does not suffer from DFS's drawbacks of delaying recruitment of large degree nodes. In fact, the next recruited node is selected independently from the time it was

For the sake of simplicity, we allow a slight abuse of notation, denoting by $\langle . \rangle$ both the empirical mean and the expected value. Moreover, we use the convention that $\langle k \rangle$ denotes the average degree and we also define $\langle x|y \rangle$ to be the average of x given y.

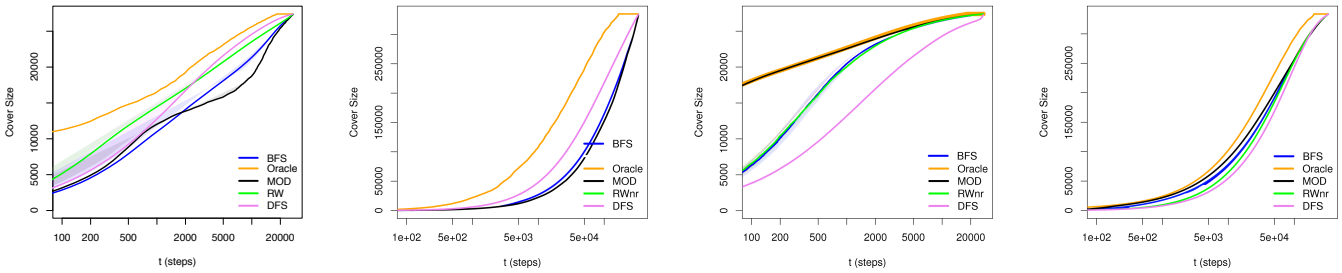
A configuration model [13] is a random graph parameterized by degree distribution $\{p_k\}_{k=1,\ldots n-1}$. Samples can be generated as follows. The degree k_i is attributed to each node i according to the selected degree distribution. Each vertex i can then be thought as having k_i stubs attached to it. The graph is generated in steps by randomly matching two unmatched stubs in the graph at each step. The configuration model is widely used in the complex network literature [14], [15] also to simplify the analysis.



(a) Enron (b) Slashdot

(c) Wiki-talk Network (d) Flickr Network

Fig. 2: Empirical average cover size $\langle \overline{W}(t) \rangle$ of various social networks. Comparison between Oracle, RW, BFS, DFS, and MOD algorithms. x-axis in log-scale.



(a) HepTh (b) Amazon (c) Randomized HepTh (d) Randomized Amazon

Fig. 3: The two datasets in which DFS outperforms MOD. Comparison between the empirical average cover size $\langle \overline{W}(t) \rangle$ of Oracle, RW, BFS, DFS, and MOD algorithms. Figs. 3(a) and 3(b) show the results on the original networks and Figs. 3(c) and 3(d) show the results in their randomized counterparts. Note that when randomized, we see results similar to those seen in the social networks. Thus, the good performance of DFS and poor performance of MOD are due to the peculiar network topology of these graphs. x-axis in log-scale.

first observed. Moreover, under a configuration model, Eq. 1 characterizes the probability that the next selected node has degree k. This probability is clearly skewed to large degree nodes.

The plots in Figs. 2 confirm that RW performs significantly better than DFS. More surprisingly, in most of the analyzed networks (Enron, Wiki-talk and for Youtube and EmailEU in our technical report [5]), RW significantly outperforms BFS too, while in Slashdot and Flickr their performance are comparable. This difference cannot be explained by the considerations above: indeed Eq. 1 predicts that BFS and RW have the same skew toward selecting larger degree nodes. We support the claim that the difference is due to the clustering structure of the graph. For example in the companion technical report [5] we show that on an infinite grid BFS asymptotically covers only a new additional node at each recruitment. On the contrary, RW in-depth search performs significantly better than BFS on 3D grids when the probability for the RW to go back to an already visited node is 0.34. The higher the dimensionality of the grid the larger the performance gap. To test if this explanation is valid also for the traces considered here, we performed simulations (not reported here) on randomly rewired instances of the above networks. In this rewired networks the clustering effect is significantly reduced and we observe that the performance of RW and BFS are indistinguishable.

V. MEUD Algorithm

One lesson to take away from Guha and Khuller's Oracle (see Sec. II) is that the knowledge of which node in $\mathcal{N}(\mathcal{B}(t))$

has the largest uncovered degree (number of neighbors not in the cover) is the key to achieving a good cover. While the uncovered degrees of nodes in $\mathcal{N}(\mathcal{B}(t))$ are not available to us, we may still be able to estimate them. In this section we propose an algorithm, denoted **Maximum Expected Uncovered Degree (MEUD)**, that at each step t recruits the node in $\mathcal{N}(\mathcal{B}(t))$ with the largest expected uncovered degree.

Some preliminary observations will make clearer the complexity of the problem MEUD is addressing. Let $d(v,t)$ be the *observed degree* of node v after t recruitments. Consider two observed node $v_1, v_2 \in \mathcal{N}(\mathcal{B}(t))$ where v_1 has a larger degree than v_2. Intuitively, we expect v_1 to have a larger expected observed degree than v_2. This result can be formally proven under a configuration model. Still, it would be incorrect to conclude that the node with the largest observed degree has also the largest expected uncovered degree. Indeed this conclusion may be true or false depending on the degree distribution of the observed nodes (it is easy to build examples where opposite choices should be done for different degree distributions even when the observed degrees are the same). We have observed in the previous section that both BFS and RW recruit earlier nodes with larger degree. Then the degree distribution of the observed and uncovered nodes keep changing (these sets become poorer and poorer of large degree nodes). A consequence is then that the optimal choice of the next node to recruit depends also on the advancement of the recruitment process.

In our analysis we assume G follows a configuration model and we omit t from some variables for the sake of conciseness. Let $\langle k|d\rangle$ and $\langle u|d\rangle$ denote the expected degree and the expected uncovered degree of a node with observed degree d, respectively. Under the configuration model of G, if a node has a larger expected excess degree than another node, then it also has a larger expected uncovered degree. In fact each edge that contributes to the excess degree has a (positive) probability of being connected to an uncovered node. Thus, larger excess degree implies larger expected uncovered degree. Then, we want to identify the node in $\mathcal{N}(\mathcal{B}(t))$ with the largest excess degree.

In what follows we obtain an approximation of $\langle k-d|d\rangle$ using $\{p_k\}_{k=1,\ldots}$, that we assume to be known for the moment. Later we show that for some important families of random networks, the node with maximum observed degree is also the node with the maximum expected excess degree. Let $\zeta_k(t)$ be the probability that a random node in $\overline{\mathcal{B}}(t)$ has degree k. Note that $\zeta_k(0) = p_k$ as, by definition, the initial node in $\mathcal{B}(0)$ is randomly sampled from V. In general we have

$$\zeta_k(t) = C p_k (1 - b_k(t)),$$

where $b_k(t)$ is the fraction of nodes of degree k in $\mathcal{B}(t)$ and C is a normalization constant.

Let $\mathcal{N}^{(d)}(\mathcal{B}(t)) \subseteq \overline{\mathcal{B}}(t)$ denote the set of nodes with at least d recruited neighbors. Note that $\mathcal{N}^{(1)}(\mathcal{B}(t)) = \mathcal{N}(\mathcal{B}(t))$ and $\mathcal{N}^{(0)}(\mathcal{B}(t)) = \overline{\mathcal{B}}(t)$. Under the configuration model we can determine $\{\mathcal{N}^{(d)}(\mathcal{B}(t))\}_{d=1,\ldots}$ and $\mathcal{W}(t)$ through the following process that dynamically assigns nodes from $\overline{\mathcal{B}}(t)$ to these sets. Let's assume $N \gg 1$ so we do not need to worry about self-loops. Detach all nodes $v \in V$ from their neighbors such that node v with degree k_v has k_v "active stubs." Iteratively match an active stub in $\mathcal{B}(t)$ to another random active stub in V. Whenever an active stub of a node $u \in \mathcal{N}^{(d)}(t)$, $d \in \{0, 1, \ldots\}$, is selected, we add v to $\mathcal{N}^{(d+1)}(t)$ and mark both stubs of the edge "inactive", that is, we promote u to $\mathcal{N}^{(d+1)}(t)$ but reduce its active degree by one.

Now we need to consider the degree distribution of the unrecruited nodes. The following recursion describes the degree distribution $\{\zeta_k^{(d+1)}\}_{k=d+1,\ldots}$ of the nodes in $\mathcal{N}^{(d+1)}(t)$ in terms of the degree distribution $\{\zeta_k^{(d)}\}_{k=d,\ldots}$ of nodes in $\mathcal{N}^{(d)}(t)$:

$$\zeta_k^{(d+1)} = \frac{(k-d)\zeta_k^{(d)}}{\langle k\rangle_{\zeta^{(d)}} - d}, \quad k \geq d, \tag{2}$$

and $\zeta_k^{(d)} = 0$ for $k < d$, where $\langle k\rangle_{\zeta^{(d)}} \equiv \sum_{k\geq 0} k\zeta_k^{(d)}$ is the average degree of nodes with at least d recruited (black) neighbors.

To obtain $\langle k|d\rangle$ from $\langle k\rangle_{\zeta^{(d)}}$ we use the fact that $\mathcal{N}^{(0)}(\mathcal{B}(t)) \supseteq \mathcal{N}^{(1)}(\mathcal{B}(t)) \supseteq \cdots$. Let N_d be the number of nodes in $\mathcal{N}^{(d)}(\mathcal{B}(t))$ with observed degree d. Note that $N_1 = N(\mathcal{B}(t))$. For any two sets A, A', such that $A' \subseteq A$, the following holds: $\text{vol}(A - A') = \text{vol}(A) - \text{vol}(A')$. Considering $A = \mathcal{N}^{(d)}$ and $A' = \mathcal{N}^{(d+1)}$ it is easy to show that

$$\langle k|d\rangle = \left\langle \frac{N_d}{N_d - N_{d+1}} \right\rangle \langle k\rangle_{\zeta^{(d)}} - \left\langle \frac{N_{d+1}}{N_d - N_{d+1}} \right\rangle \langle k\rangle_{\zeta^{(d+1)}}.$$

The expectations of $\langle N_d/(N_d - N_{d+1})\rangle$ and $\langle N_{d+1}/(N_d - N_{d+1})\rangle$ are hard to compute, as they are expectations over all sample paths. However, in practice, the algorithm can estimate the values of $\langle N_d/(N_d - N_{d+1})\rangle$ and $\langle N_{d+1}/(N_d - N_{d+1})\rangle$ empirically using the sample average, $\langle N_d/(N_d - N_{d+1})\rangle \approx N_d/(N_d - N_{d+1})$ and $\langle N_{d+1}/(N_d - N_{d+1})\rangle \approx N_{d+1}/(N_d - N_{d+1})$.

Our calculations of $\langle N_d/(N_d - N_{d+1})\rangle$ and $\langle N_{d+1}/(N_d - N_{d+1})\rangle$ should be used with caution as they do not consider the extra density of connections inside $\mathcal{B}(t)$ created by the MEUD recruitment process. Taking this bias into account is not trivial and is the subject of future work. However, in our MEUD simulations we observe that $N_{d+1} \ll N_d$ for large d, and, under such scenario, it is reasonable to make the following simplification $\langle k-d|d\rangle \approx \langle k-d\rangle_{\zeta^{(d)}}$.

Unfortunately, obtaining $\langle k-d\rangle_{\zeta^{(d)}}$ still requires knowing $\zeta_k^{(d)}$, $\forall k, d$, which in turn requires knowing the degree distribution. We observe that if $\langle k-d\rangle_{\zeta^{(d)}}$ is increasing in the observed degree d, then MEUD simplifies to an algorithm that always selects the node $v^\star$ with the maximum observed degree in $\mathcal{N}(\mathcal{B}(t))$. We denote this simplified MEUD heuristic **Maximum Observed Degree (MOD)**.

In the technical report [5] we perform the calculation of $\langle k-d\rangle_{\zeta^{(d)}}$ for different random graph families and in particular for random graphs for which ζ_k follows a power law with exponential cut-off (that has been shown to well approximate

a variety of real world networks [22], [23]):

$$\zeta_k = \frac{k^{-\tau} C_t^k}{\mathrm{Li}_\tau(C_t)}, \quad \text{for } k \geq 1,$$

where $C_t < 1$ is a parameter that depends on $\mathcal{B}(t)$ and t and the normalization factor $\mathrm{Li}_h(x) = \sum_{k=1}^{\infty} x^k/k^h$ is the h-th polylogarithm function of x. For such networks we prove analytically that at step t we should recruit: 1) the node $v \in \mathcal{N}(\mathcal{B}(t))$ with the largest observed degree if $\tau = 1$, 2) the node with either the largest or the second largest observed degree if $\tau = 2$, 3) a node with observed degree at least $\lceil \tau \rceil$ if $C_t \to 1$ and $\tau > 0$. These results suggest that MOD is a good approximation of MEUD in these networks.

Also, in practice our simulations show that MOD outperforms all other heuristics over all social networks (see Enron, Slashdot, Wiki-talk, and Flickr in Figure 2 and Youtube and EmailEU in our technical report [5]). We also see that in the randomized versions of the HepTh and Amazon product networks MOD is either significantly better (for the random HepTh, Figure 3(c)) or no worse than all other methods (for the random Amazon, Figure 3(d)). The results are clear, MOD is superior to all other algorithms (in the randomized HepTh it even matches the performance of Oracle). In more structured networks, unlike in social networks, such as the original HepTh citation network and Amazon's co-purchase product network, DFS outperforms MOD due to the particular structure of these networks. It remains an open question whether, in respect to the network cover problem, most social networks are more similar to "random networks" or more similar to "highly structured networks". Our datasets and simulation results suggest that social networks tend to be unstructured, more similar to random networks through the network exploration point of view.

VI. Conclusions & Related Work

We have considered the problem of providing an online algorithm that, by recruiting nodes through their neighbors, greedily maximizes the network cover of an online social network. In our setting the network topology was unknown and the only topological information available came from the identity of the neighbors of already recruited nodes.

In this scenario, we have evaluated the efficacy of existing network sampling algorithms (BFS, DFS, RW) and proposed a new algorithm, Maximum Expected Uncovered Degree (MEUD), inspired by the one-hop lookahead greedy approximation to the minimum connected dominating set of Guha and Khuller [3], (denoted "Oracle" in this work) to recruit at every step the node with the largest excess degree. The MEUD heuristic recruits the node with the largest expected uncovered degree with the help of degree distribution side information. In the absence of degree distribution information, we have shown that on random power law and Erdös-Rényi networks MEUD can be approximated by MOD (Maximum Observed Degree), a greedy heuristic that at every step recruits the node with the largest observed degree. We have shown through extensive simulations on real world social network datasets that MOD outperforms all other algorithms, often quite significantly.

Finally, we have uncovered an interesting previously unknown property of DFS: DFS performs remarkably poorly on social networks. In fact, DFS seems to avoid recruiting nodes with large excess degrees. We have argued that this is due to its tendency to keep large degree nodes at the bottom of its recruitment queue. We note in passing that this property of DFS may find applications in undercover military operations where one seeks to recruit target individuals with the minimum exposure (number of connections) to unrecruited targets.

References

[1] C. *How Obama's data crunchers helped him win*, "http://liveweb.archive.org/http://www.cnn.com/2012/11/07/tech/web/obama-campaign-tech-team/index.html."
[2] M. R. Garey and D. S. Johnson, *Computers and Intractability; A Guide to the Theory of NP-Completeness*. W. H. Freeman & Co., 1990.
[3] S. Guha and S. Khuller, "Approximation algorithms for connected dominating sets," *Algorithmica*, vol. 20, no. 4, pp. 374–387, Apr. 1998.
[4] A. S. Maiya and T. Y. Berger-Wolf, "Benefits of bias: towards better characterization of network sampling," in *SIGKDD*, 2011, pp. 105–113.
[5] K. Avrachenkov, P. Basu, G. Neglia, B. Ribeiro, and D. Towsley, "Online myopic network covering," University of Massachusetts Amherst, Tech. Rep. UM-CS-2012-034.
[6] G. Nemhauser, L. Wolsey, and M. Fisher, "An analysis of approximations for maximizing submodular set functions—i," *Mathematical Programming*, vol. 14, no. 1, pp. 265–294, 1978.
[7] M. Ripeanu, A. Iamnitchi, and I. Foster, "Mapping the Gnutella Network," *IEEE Internet Computing*, vol. 6, no. 1, pp. 50–57, Jan. 2002.
[8] A. Mislove, M. Marcon, K. P. Gummadi, P. Druschel, and B. Bhattacharjee, "Measurement and analysis of online social networks," in *IMC*, 2007, pp. 29–42.
[9] J. L. S. network data repository, "http://snap.stanford.edu/data/."
[10] O. Frank and T. Snijders, "Estimating the size of hidden populations using snowball sampling," *Journal of Official Statistics*, vol. 10, pp. 53–67, 1994.
[11] M. Kurant, A. Markopoulou, and P. Thiran, "Towards Unbiased BFS Sampling," *JSAC*, vol. 29, no. 9, pp. 1799 –1809, Oct. 2011.
[12] M. Najork and J. L. Wiener, "Breadth-first crawling yields high-quality pages," in *WWW*, 2001, pp. 114–118.
[13] M. E. J. Newman, "The structure and function of complex networks," *SIAM REVIEW*, vol. 45, pp. 167–256, 2003.
[14] A. Barrat, M. Barthélemy, and A. Vespignani, *Dynamical Processes on Complex Networks*. Cambridge University Press, 2008.
[15] M. E. J. Newman, *Networks: An Introduction*. Oxford University Press, 2010.
[16] K. Gile and M. S. Handcock, "Respondent-driven sampling: An assessment of current methodology," *Sociological Methodology*, vol. 40, no. 285–327, 2010.
[17] M. Gjoka, M. Kurant, C. Butts, and A. Markopoulou, "Practical recommendations on crawling online social networks," *JSAC*, vol. 29, no. 9, pp. 1872 –1892, october 2011.
[18] S. Goel and M. J. Salganik, "Assessing respondent-driven sampling," *PNAS*, vol. 107, no. 6743–6747, 2010.
[19] M. Kurant, M. Gjoka, C. T. Butts, and A. Markopoulou, "Walking on a graph with a magnifying glass: stratified sampling via weighted random walks," in *SIGMETRICS*, 2011, pp. 281–292.
[20] M. Mihail, A. Saberi, and P. Tetali, "Random walks with lookahead on power law random graphs," *Internet Mathematics*, vol. 3, no. 2, pp. 147–152, 2006.
[21] B. Ribeiro and D. Towsley, "Estimating and sampling graphs with multidimensional random walks," in *IMC*, 2010, pp. 390–403.
[22] L. Amaral, A. Scala, M. Barthelemy, and H. Stanley, "Classes of small-world networks," *Proc. Natl. Acad. Sci.*, no. 97, pp. 11 149–11 152, 2000.
[23] M. E. J. Newman, "The structure of scientific collaboration networks," *PNAS*, vol. 98, pp. 404–409, 2001.