On the bias of BFS (Breadth First Search)

Maciej Kurant
School of Computer & Comm. Sciences
EPFL, Lausanne, Switzerland
maciej.kurant@gmail.com

Athina Markopoulou
EECS Dept
University of California, Irvine
athina@uci.edu

Patrick Thiran
School of Computer & Comm. Sciences
EPFL, Lausanne, Switzerland
patrick.thiran@epfl.ch

***Abstract*—Breadth First Search (BFS) and other graph traversal techniques are widely used for measuring large unknown graphs, such as online social networks. It has been empirically observed that incomplete BFS is biased toward high degree nodes. In contrast to more studied sampling techniques, such as random walks, the bias of BFS has not been characterized to date.**

In this paper, we quantify the degree bias of BFS sampling. In particular, we calculate the node degree distribution expected to be observed by BFS as a function of the fraction of covered nodes, in a random graph $RG(p_k)$ with a given (and arbitrary) degree distribution p_k. Furthermore, we also show that, for $RG(p_k)$, all commonly used graph traversal techniques (BFS, DFS, Forest Fire, and Snowball Sampling) lead to the same bias, and we show how to correct for this bias. To give a broader perspective, we compare this class of exploration techniques to random walks that are well-studied and easier to analyze. Next, we study by simulation the effect of graph properties not captured directly by our model. We find that the bias gets amplified in graphs with strong positive assortativity. Finally, we demonstrate the above results by sampling the Facebook social network, and we provide some practical guidelines for graph sampling in practice.

***Index Terms*—BFS, Breadth First Search, graph sampling, bias, OSN, Online Social Networks, Facebook.**

I. Introduction

A large body of work in the networking community focuses on topology measurements at various levels, including the Internet, the Web (WWW), peer-to-peer (P2P) and online social networks (OSN). The size of these networks and other practical restrictions make measuring the entire graph impossible. Instead, researchers typically collect and study a small but "representative" sample. In this paper, we are particularly interested in sampling networks that naturally allow to explore the neighbors of a given node (which is the case in WWW, P2P and OSN). A number of graph exploration techniques use this basic operation for sampling. They can be roughly classified in two categories: (a) with replacement (random walks), and (b) without replacement (graph traversal techniques).

In the first category, random walks, nodes can be revisited. This category includes the classic Random Walk (RW) as well as the Metropolis-Hastings Random Walk (MHRW). They are used for sampling of nodes on the Web [1], P2P networks [2]–[4], OSNs [5,6] and large graphs in general [7]. Random walks are well studied [8] and result in samples that have either no bias (MHRW) or a known bias (RW) that can be corrected for. Random walks are not the focus of this paper, but are discussed as baseline for comparison.

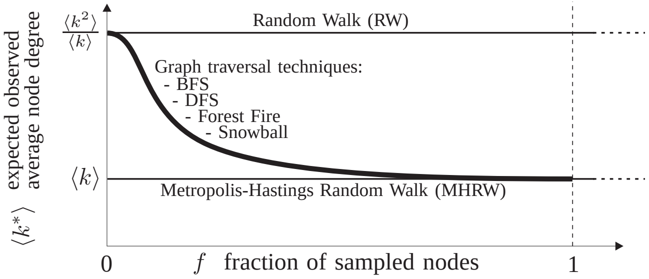

Fig. 1. **Overview of results.** We calculate the average node degree $\langle k^* \rangle$ (and the full degree distribution, not shown) expected to be observed by BFS in a random graph $RG(p_k)$ with a given degree distribution p_k, as a function of the fraction of sampled nodes f. We show RW and MHRW as a reference. $\langle k \rangle$ is the real average node degree, and $\langle k^2 \rangle$ is the real average squared node degree. *Observations:* (1) For a small sample size, BFS has the same bias as RW; with increasing f, the bias decreases; a complete BFS ($f=1$) is unbiased, as is MHRW (or uniform sampling). (2) All common graph traversal techniques (that do not revisit the same node) lead to the same bias. (3) The shape of the BFS curve depends on the real node degree distribution p_k, but it is always monotonically decreasing.

In the second category, graph traversal techniques, each node is visited exactly once (if we let the process run until completion and if the graph is connected). These methods vary in the order in which they visit the nodes; examples include BFS, Depth-First Search (DFS), Forest Fire (FF) and Snowball Sampling (SBS). Graph traversals, especially BFS, are very popular and widely used for sampling large networks, *e.g.* WWW [9] or OSNs [10]–[12]. One reason is that BFS is well-known (a textbook technique) and easy to understand. Another and probably more important reason is that (incomplete) BFS collects a full view (all nodes and edges) of some particular region in the graph. Consequently, we can study the topological characteristics (e.g., shortest path lengths, clustering coefficients, community structure) of the resulting sample, which is a big advantage of BFS over random walks. Of course, this study is correct only if the BFS sample is representative of the entire graph. At first sight it seems true, e.g., a BFS sample of a lattice is a (smaller) lattice.

Unfortunately, this intuition often fails. It was observed empirically that BFS introduces a bias towards high-degree nodes [9] [13]–[15]. We also confirmed this fact in a recent measurement of Facebook [5], where our BFS crawler found the average node degree $\langle k^{BFS} \rangle \simeq 324$, while the real value is only $\langle k \rangle \simeq 94$, *i.e.*, about 3.5 times smaller. To make

This work was supported by: grant ManCom 2110 of the Hasler Foundation, Bern, Switzerland; SNF grant PBELP2-130871, Switzerland; and NSF CAREER grant 0747110, USA.

things worse, many important properties are usually strongly correlated with the node degree (e.g., privacy settings in Facebook [5]), which makes the BFS sample far from being representative with respect to many metrics.

Despite the popularity of BFS on the one hand, and its bias on the other hand, we still know relatively little about the statistical properties of node sequences returned by BFS. The formal analysis is challenging because BFS, similarly to every sampling without replacement, introduces complex dependencies difficult to deal with mathematically.

Our work is a step toward understanding the statistical characteristics of incomplete BFS sampling. In particular, we focus on a random graph $RG(p_k)$ with a given (and arbitrary) degree distribution p_k, which we introduce in Section IV. In Section V, we calculate precisely the node degree distribution expected to be observed by BFS as a function of the fraction of sampled nodes in $RG(p_k)$.

We accompany this central result with additional related contributions. First, we show that in $RG(p_k)$, BFS is equivalent to other graph traversal techniques, such as Depth First Search (DFS), Snowball Sampling, and Forest Fire (we overview these methods in Section III). Second, we compare the bias of BFS to that of random walks. As shown in Fig. 1, at the beginning of the exploration process, BFS exhibits exactly the same bias as RW; with increasing fraction of sampled nodes f, this bias monotonically decreases; when BFS is complete ($f=1$), there is no bias, as in the case of MHRW. Third, given a biased sample, we derive in Section VI an unbiased estimator of the original node degree distribution.

In addition, in Section VII, we use simulation to confirm our analysis and investigate the effect of graph properties, such as assortativity, not captured directly by $RG(p_k)$. We complement it with real-world measurements of the Facebook social network in Section VIII. Section IX concludes and outlines future work.

II. Related Work

BFS used in practice. BFS is widely used today for exploring large networks, such as OSNs. The following list provides some examples but is by no means exhaustive. In [10], Ahn et al. used BFS to sample Orkut and MySpace. In [11] and [16], Mislove et al. used BFS to crawl the social graph in four popular OSNs: Flickr, LiveJournal, Orkut, and YouTube. In [12], Wilson et al. measured the social graph and the user interaction graph of Facebook using several BFSs, each BFS constrained in one of the largest 22 regional Facebook networks. In our recent work [5], we have also crawled Facebook using various sampling techniques, including BFS, RW and MHRW. It has been empirically observed that incomplete BFS and its variants introduce bias towards high-degree nodes [9] [13]–[15]. We also confirmed this in Facebook [5], an observation that in fact inspired this paper.

Analyzing BFS. To the best of our knowledge, the sampling bias of BFS has not been analyzed so far. [17] and [18] are the closest related papers to our methodology. The original paper by Kim [17] analyzes the size of the largest connected component in classic Erdös-Rényi random graph by essentially applying the configuration model with node degrees chosen from a Poisson distribution. To match the stubs (or 'clones' in [17]) uniformly at random in a tractable way, Kim proposes a "cut-off line" algorithm: he first assigns each stub a random index from $[0, np]$, and next progressively scans this interval. Achlioptas et al. used this powerful idea in [18] to study the bias of traceroute sampling in random graphs with a given degree distribution. The basic operation in [18] is traceroute (*i.e.*, "discover a path") and is performed from a single node to all other nodes in the graph. The union of the observed paths forms a "BFS-tree", which includes all nodes but misses some edges (*e.g.*, those between nodes at the same depth in the tree). In contrast, the basic operation in the traversal methods presented in our paper is to discover all neighbors of a node, and it is applied to all nodes in increasing distance from the origin. Another important difference is that [18] studies a completed BFS-tree, whereas we study the sampling process when it has visited only a fraction $f<1$ of nodes; a completed BFS ($f=1$) is trivial in our case (it has no bias).

Another recent paper related to BFS bias is [19]. The paper is about Snowball Sampling [20], which is similar to BFS, and proposes a heuristic approach to correct the degree biases in ith generation of Snowball based on the values measured in generation $i-1$. The authors show by simulation that this technique performs moderately well, especially when a significant fraction of nodes have been covered.

III. Graph exploration techniques

Let $G=(V,E)$ be a connected graph with the set of vertices V, and a set of undirected edges E. Initially, G is unknown, except for one (or some limited number of) seed node(s). When sampling through graph exploration, we begin at the seed node, and we recursively visit (one, some or all) of its neighbors. We distinguish two main categories of exploration techniques: with and without replacement.

A. Exploration with replacement (random walks)

Exploration *with replacement*, or simply a *walk*, allows revisiting the same node many times. Consider the following classic examples:

1) Random Walk (RW): In this classic sampling technique [8], we start at some seed node. At every iteration, the next-hop node v is chosen uniformly at random among the neighbors of the current node u. It is easy to see that RW introduces a linear bias towards nodes of high degree [8].

2) Metropolis Hastings Random Walk (MHRW): In this technique, as in RW, the next-hop node w is chosen uniformly at random among the neighbors of the current node u. However, with a probability that depends on the degrees of w and u, MHRW performs a self-loop instead of moving to w. More specifically, the probability $P_{u,w}$ of moving from u to w is as follows [21]:

$$P_{u,w} = \begin{cases} \frac{1}{k_u}\cdot\min(1,\frac{k_u}{k_w}) & \text{if } w \text{ is a neighbor of } u, \\ 1-\sum_{y\neq u} P_{u,y} & \text{if } w=u, \\ 0 & \text{otherwise,} \end{cases} \tag{1}$$

where k_v is the degree of node v. Essentially, MHRW reduces the transitions to high degree nodes and thus eliminates the degree bias of RW. This property of MHRW was recently exploited in various network sampling contexts [2,3,5,6].

3) Respondent-Driven Sampling (RDS): RDS was proposed and studied in the field of social sciences to penetrate hidden populations, such as that of drug addicts [22,23]. In the network sampling terminology, at each iteration RDS selects randomly exactly n neighbors (typically $n \simeq 3$) of the current node u and schedules them to visit later. RDS visits the nodes in the order they were scheduled. Thus, RDS is a modification of Snowball Sampling (described below) that allows node revisiting. RDS introduces a degree bias that is known and can be corrected for. It was demonstrated in [23] on the example with $n=1$, which reduces RDS precisely to Random Walk (RW). This approach was recently tested in [3] on various graph models and unstructured P2P networks.

B. Exploration without replacement (graph traversals)

In contrast, exploration without replacement, or *graph traversal*, never revisits the same node. At the end of the process, and assuming that the graph is connected, all nodes are visited.

1) Breadth First Search (BFS): BFS is a classic graph traversal algorithm that starts from the seed and progressively explores all neighbors. At each new iteration the earliest explored but not-yet-visited node is selected next. Thus, BFS discovers all nodes within some distance from the seed.

2) Depth First Search (DFS): This technique is similar to BFS, except that at each iteration we select the latest explored but not-yet-visited node. As a result, DFS explores first the nodes that are faraway (in the number of hops) from the seed.

3) Forest Fire (FF): FF is a randomized version of BFS, where for every neighbor v of the current node, we flip a coin, with probability of success p, to decide if we explore v. FF reduces to BFS for $p=1$. It is possible that this process dies out before it covers all nodes. In this case, in order to make FF comparable with other techniques, we revive the process from a random node already in the sample. Forest Fire is inspired by the graph growing model of the same name proposed in [24] and is used as a graph sampling technique in [7].

4) Snowball Sampling (SBS): Snowball Sampling is a precursor of RDS and a term loosely used for BFS-like traversal techniques. According to a classic definition by Goodman [20], an n-name Snowball Sampling is similar to BFS, but at every node v, not all k_v, but exactly n neighbors are chosen randomly out of all k_v neighbors of v. These n neighbors are scheduled to visit, but only if they have not been visited before.

IV. Graph model $RG(p_k)$

A basic important graph property is the node degree distribution p_k, *i.e.*, the fraction of nodes with degree equal to k, for all $k \geq 0$.[1] Depending on the network, the degree distribution can vary, ranging from constant-degree (in regular graphs), a distribution concentrated around the average value (*e.g.*, in Erdös-Rényi random graphs or in well-balanced P2P networks), to heavily right-skewed distributions with k covering several decades (in WWW, unstructured P2P, Internet at the Autonomous System level, OSNs). We handle all these cases by assuming that we are given *any* fixed node degree distribution p_k. Other than that, the graph G is completely random. That is, G is drawn uniformly at random from the set of all multigraphs with degree distribution p_k. We denote this model by $RG(p_k)$.

$G=(V,E)$	graph G with nodes V and edges E
k_v	degree of node v
$p_k \; = \frac{1}{\lvert V\rvert}\sum_{v\in V} 1_{k_v=k}$	degree distribution in G
q_k	expected observed degree distribution
$\widehat{q}_k$	observed degree distribution
$\widehat{p}_k$	estimated original degree distribution in G
$\langle k\rangle \; = \sum_k k\, p_k$	average node degree in G
$\langle k^*\rangle \; = \sum_k k\, q_k$	expected observed average node degree
f	fraction of nodes covered by the sample

TABLE I
NOTATION SUMMARY.

We use a classic technique to generate $RG(p_k)$, called the *configuration model* [25]: each node v is given k_v "stubs" (or "edges-to-be"). Next, all these $\sum_{v\in V} k_v = 2|E|$ stubs are randomly matched in pairs, until all stubs are exhausted (and $|E|$ edges are created). In Fig. 2 (ignore the rectangular interval [0,1] for now), we present four nodes with their stubs (left) and an example of their random matching (right).

V. Analyzing the Node Degree Bias

In this section, we study the node degree bias observed when the graph exploration techniques of Section III are run on the random graph $RG(p_k)$ of Section IV. In particular, we derive the node degree distribution q_k and the average node degree $\langle k^*\rangle$ expected to be observed, as a function of the original degree distribution p_k and, in the case of BFS, of the fraction of sampled nodes f.

A. Exploration with replacement (walks)

We begin by summarizing the relevant results known for walks, in particular for RW and MHRW. They will serve as a reference point for our main analysis of graph traversals in the next section.

1) Random Walk (RW): Random walks have been widely studied; see [8] for an excellent survey. In any given connected and aperiodic graph, the probability of being at a particular node v converges at equilibrium to the stationary distribution $\pi_v = \frac{k_v}{2|E|}$. Therefore, the expected observed degree distribution q_k is

$$q_k \;=\; \sum_v \pi_v \cdot 1_{\{k_v=k\}} \;=\; \frac{k}{2|E|}\, p_k \,|V| \;=\; \frac{k\, p_k}{\langle k\rangle}, \qquad (2)$$

where $\langle k\rangle$ is the average node degree in G. Eq. (2) is essentially similar to calculation for RDS in [23,26]. As this holds for any fixed (and connected and aperiodic) graph, it is also true for all connected graphs generated by the configuration

[1] As we define p_k as a 'fraction', not the 'probability', p_k determines the degree sequence in the graph, and vice versa.

model. Consequently, the expected observed average node degree is

$$\langle k^* \rangle \;=\; \sum_k k\, q_k \;=\; \frac{\sum_k k^2\, p_k}{\langle k \rangle} \;=\; \frac{\langle k^2 \rangle}{\langle k \rangle}, \qquad (3)$$

where $\langle k^2 \rangle$ is the average squared node degree in G. We show this value $\frac{\langle k^2 \rangle}{\langle k \rangle}$ in Fig. 1.

2) Metropolis Hastings Random Walk (MHRW): It is easy to show that the transition matrix $P_{u,w}$ shown in Eq.(1) leads to a uniform stationary distribution $\pi_v = \frac{1}{|V|}$ [21], and consequently:

$$q_k \;=\; p_k \qquad (4)$$

$$\langle k^* \rangle \;=\; \sum_k k\, q_k \;=\; \sum_k k\, p_k \;=\; \langle k \rangle. \qquad (5)$$

In Fig. 1, we show that MHRW estimates the true mean.

B. Exploration without replacement (Main Result)

In both RW and MHRW the nodes can be revisited. So the state of the system at iteration $i+1$ depends only on iteration i, which makes it possible to analyze with Markov Chain techniques. In contrast, graph traversals do not allow for node revisits, which introduces crucial dependencies between all the iterations and significantly complicates the analysis. To handle these dependencies, we adopt an elegant technique recently introduced in [17] (to study the size of the largest connected component) and extended in [18] (to study the bias of traceroute sampling). However, our work differs in many aspects from both [17] and [18], which we comment in detail in the related work Section II.

1) Exploration without replacement at the stub level: We begin by defining Algorithm 1 (below) - a general graph traversal technique that collects a sequence of nodes S, without replacements. To be compatible with the configuration model (see Section IV), we are interested in the process *at the stub level*, where we consider one stub at a time, rather than one node at a time. An integral part of the algorithm is a queue Q that keeps the discovered, but still not-yet-followed stubs. First, we enqueue on Q all the stubs of some initial node v_1, and by setting $S \leftarrow [v_1]$. Next, at every iteration, we dequeue one stub from Q, call it a, and follow it to discover its partner-stub b, and b's owner $v(b)$. If node $v(b)$ is not yet discovered, *i.e.*, if $v(b) \notin S$, then we append $v(b)$ to S and we enqueue on Q all other stubs of $v(b)$.

Depending on the scheduling discipline for the elements in Q (line 3), Algorithm 1 implements BFS (for a first-in first out scheduling), DFS (last-in first-out) or Forest Fire (first-in first-out with randomized stub losses). Line 9 guarantees that the algorithm never tracebacks the edges, *i.e.*, that stub a dequeued from Q in line 3 never belongs to an edge that has already been traversed in the opposite direction.

2) Discovery on the fly: In line 4 of Algorithm 1, we follow stub a to discover its partner b. In a fixed graph G, this step is deterministic. In the configuration model $RG(p_k)$, a fixed graph G is obtained by matching all the stubs uniformly at random. Next, we can sample this fixed graph and average the result over the space of all the random graphs $RG(p_k)$ that have just been constructed. Unfortunately, this space grows exponentially with the number of nodes $|V|$, making the problem untractable. Therefore, we adopt an alternative construction of G - by iteratively selecting b 'on the fly' (i.e, every time line 4 is executed), uniformly at random from all still unmatched stubs. By the principle of deferred decisions [27], these two approaches are equivalent.

Algorithm 1 Stub-Level Graph Traversal

```
 1: $S \leftarrow [v_1]$  and  $Q \leftarrow$ [all stubs of $v_1$]
 2: while $Q$ is nonempty do
 3:     Dequeue $a$ from $Q$
 4:     Discover $a$'s partner $b$
 5:     if $v(b) \notin S$ then
 6:         Append $v(b)$ to $S$
 7:         Enqueue on $Q$ all stubs of $v(b)$ except $b$
 8:     else
 9:         Remove $b$ from $Q$
10:     end if
11: end while
```

With the help of the 'on the fly' approach, we are able to write down the equations we need. Indeed, let us denote by $X_i \in V$ the ith selected node, and let $\mathbb{P}(X_1 = u)$ be the probability that node $u \in V$ is chosen as a starting node. It is easy to show that with $z = 2|E|$ we have

$$\mathbb{P}(X_2 = v) \;=\; \sum_{u \neq v} \frac{k_v}{z - k_u} \cdot \mathbb{P}(X_1 = u) \qquad (6)$$

$$\mathbb{P}(X_3 = w) \;=\; \sum_{v \neq w} \sum_{u \neq w,v} \frac{k_w}{z - k_v - k_u} \cdot \frac{k_v}{z - k_u} \cdot \mathbb{P}(X_1 = u), \qquad (7)$$

and so on. Theoretically, these equations allow us to calculate the expected node degree at any iteration, and thus the degree bias of BFS.

3) Breaking the dependencies: There is still one problem with the equations above. Due to the increasing number of nested sums, the results can be calculated in practice for a first few iterations only. This is because we select stub b uniformly and independently at random from all the *unmatched* stubs. So the stub selected at iteration i depends on the stubs selected at iterations $1 \ldots i-1$, which results in the nested sums. We remedy this by implementing the 'on the fly' approach as follows. First, we assign each stub a real-valued index t drawn uniformly at random from the interval $[0, 1]$. Then, every time we process line 4, we pick b as the unmatched stub with the smallest index. We can interpret this as a continuous-time process, where we determine progressively the partners of stubs dequeued from Q, by scanning the interval from 'time' $t = 0$ to $t = 1$ in a search of unmatched stubs. Because the indices chosen by the stubs are independent from each other, the above trick breaks the dependence between the stubs, which is crucial for making this approach tractable.

In Fig. 2, we present an example execution of Algorithm 1, where line 4 is implemented as described above.

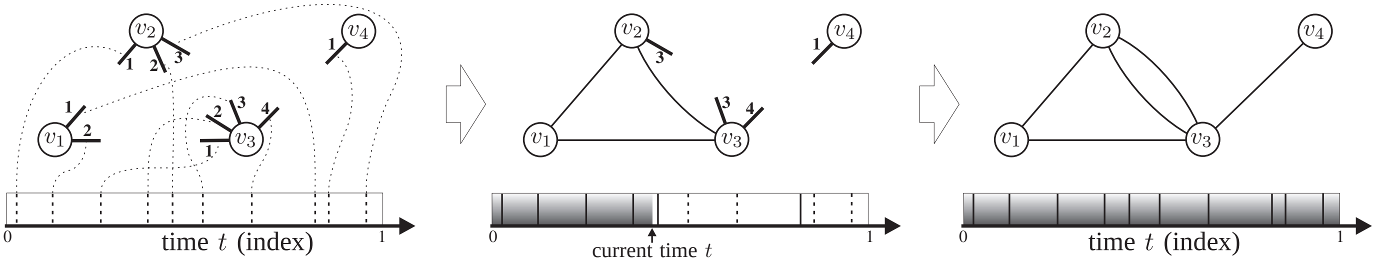

Fig. 2. An illustration of the stub-level, on-the-fly graph exploration without replacements. In this particular example, we show an execution of BFS starting at node v_1. **Left:** Initially, each node v has k_v stubs, where k_v is a given target degree of v. Each of these stubs is assigned a real-valued number drawn uniformly at random from the interval $[0,1]$ shown below the graph. Next, we follow Algorithm 1 with a starting node v_1. The numbers next to the stubs of every node v indicate the order in which these stubs are enqueued on Q. **Center:** The state of the system at time t. All stubs in $[0,t]$ have already been matched (the indices of matched stubs are set in plain line). All unmatched stubs are distributed uniformly at random on $(t,1]$. This interval can contain also some (here two) already matched stubs. **Right:** The final result is a realization of a random graph G with a given node degree sequence (i.e., of the configuration model). G may contain self-loops and multiedges.

4) Expected sampled degree distribution q_k: Now we are ready to derive the expected observed degree distribution q_k. Recall that all the stub indices are chosen independently and uniformly from $[0,1]$. A vertex v with degree k is not sampled yet at time t if the indices of all its k stubs are larger than t, which happens with probability $(1-t)^k$. So the probability that v is sampled before time t is $1-(1-t)^k$. Therefore, the expected fraction of vertices of degree k sampled before t is

$$f_k(t) = p_k(1-(1-t)^k). \tag{8}$$

By normalizing Eq. (8), we obtain the expected observed (sampled) degree distribution at time t:

$$q_k(t) \;=\; \frac{f_k(t)}{\sum_l f_l(t)} \;=\; \frac{p_k(1-(1-t)^k)}{\sum_l p_l(1-(1-t)^l)}. \tag{9}$$

Unfortunately, it is difficult to interpret $q_k(t)$ directly, because t is proportional neither to the number of matched edges nor to the number of discovered nodes. Recall that our primary goal is to express q_k as a function of fraction f of covered nodes. We achieve this by calculating $f(t)$ - the expected fraction of nodes, of any degree, visited before time t

$$f(t) = \sum_k f_k(t) = 1 - \sum_k p_k(1-t)^k \ . \tag{10}$$

Because $p_k \geq 0$, and $p_k > 0$ for at least one $k > 0$, the term $\sum_k p_k(1-t)^k$ is continuous and strictly decreasing from 1 to 0 with t growing from 0 to 1. Thus, for $f \in [0,1]$ there exists a well defined $t{=}t(f)$ that satisfies Eq.(10). Although we cannot compute $t(f)$ analytically (except in some special cases such as for $k \leq 4$), it is straightforward to find it numerically. Now, we can rewrite Eq. (9) as

$$q_k(f) \;=\; \frac{p_k(1-(1-t(f))^k)}{\sum_l p_l(1-(1-t(f))^l)}, \tag{11}$$

which is the expected observed degree distribution after covering fraction f of nodes of graph G.

5) Equivalence of traversal techniques under $RW(p_k)$: An interesting observation is that, under the random graph model $RW(p_k)$, all common traversal techniques (BFS, DFS, FF, SBS, …) are subject to exactly the same bias. The explanation is that the sampled node sequence S is fully determined by the choice of stub indices on $[0,1]$, independently of the way we manage the elements in Q.

6) Equivalence to weighted sampling without replacement: Consider a node v with a degree k_v. The probability that v is discovered before time t, given that it has not been discovered before $t_0 \leq t$, is

$$\mathbb{P}(v \text{ before time } t \mid v \text{ not before } t_0) = 1 - \left(\frac{1-t}{1-t_0}\right)^{k_v} \tag{12}$$

We now take the derivative of the above equation with respect to t, which results in the conditional probability density function $k_v(\frac{1-t}{1-t_0})^{k_v-1}$. Setting $t \to t_0$ (but keeping $t > t_0$), reduces it to k_v, which is the probability density that v is sampled at t_0, given that it has not been sampled before. This means that at every point in time, out of all nodes that have not yet been selected, the probability of selecting v is proportional to its degree k_v. Therefore, this scheme is equivalent to node sampling weighted by degree, without replacements.

7) Equivalence to RW for $f\to 0$: Finally, for $f\to 0$ (and thus $t\to 0$), we have $1-(1-t)^k \simeq kt$, and Eq. (9) simplifies to Eq. (2). This means that in the beginning of the sampling process, every traversal technique is equivalent to RW, as shown in Fig. 1 for $f\to 0$.

8) $\langle k^* \rangle$ is decreasing in f: As in Section V-B2, let $X_i \in V$ be the ith selected node, and let $z = 2|E|$. We have shown above that our procedure is equivalent to weighted sampling without replacements, thus we can write $\mathbb{P}(X_1 = u) = \frac{k_u}{z}$. Now, it follows from Eq. (6) that $\mathbb{P}(X_2 = w) = \frac{k_w}{z} \cdot \alpha_w$, where $\alpha_w = \sum_{u \neq w} \frac{k_u}{z-k_u}$. Because for any two nodes a and b, we have $\alpha_b - \alpha_a = z(k_a-k_b)/((z-k_a)(z-k_b))$, α_w strictly decreases with growing k_w. As a result, $\mathbb{P}(X_2)$ is more concentrated around nodes with smaller degrees than is $\mathbb{P}(X_1)$, implying that $\mathbb{E}[k_{X_2}] < \mathbb{E}[k_{X_1}]$. We can use an analogous argument at every iteration $i \leq |V|$, which allows us to say that $\mathbb{E}[k_{X_i}] < \mathbb{E}[k_{X_{i-1}}]$. In other words, $\langle k^* \rangle(f)$ is a decreasing function of f.

A practical consequence is that many short traversals (*e.g.,* BFS-es) are more biased than a long one, with the same total number of samples.

C. Comments on the graph connectivity

Note that the configuration model $RG(p_k)$ might result in a graph G that is not connected. In this case, every exploration technique covers only the component C in which it was initiated; consequently, the process described in Section V-B3 stops once C is covered.

VI. Correcting for node degree bias

In the previous section we derived the expected observed degree distribution q_k as a function of the original degree distribution p_k, for three general graph exploration techniques. The distribution q_k is usually biased towards high-degree nodes. In this section, we derive unbiased estimators $\widehat{p}_k$ and $\langle\widehat{k}\rangle$ of the original degree distribution p_k and its mean $\langle k\rangle$, respectively.

Let $S \subset V$ be a sequence of vertices that we sampled. Based on S, we can estimate q_k as

$$\widehat{q}_k \quad = \quad \frac{\text{number of nodes in } S \text{ with degree } k}{|S|} \tag{13}$$

A. Random Walk (RW)

In order to estimate p_k based on $\widehat{q}_k$, consider again Eq.(2), which says that q_k is proportional to $k\,p_k$. Therefore, p_k is proportional to q_k/k, and $\widehat{p}_k$ is proportional to $\widehat{q}_k/k$ which allows us to write (similarly to [3,23]):

$$\widehat{p}_k = \frac{\widehat{q}_k}{k} \;\cdot\; \left(\sum_l \frac{\widehat{q}_l}{l}\right)^{-1} \tag{14}$$

where $\sum_l \frac{\widehat{q}_l}{l}$ is a normalizing constant. From Eq.(14), we can estimate the average node degree as

$$\langle\widehat{k}\rangle \;=\; \sum_k k\,\widehat{p}_k \;=\; 1 \cdot \left(\sum_l \frac{\widehat{q}_l}{l}\right)^{-1} = \frac{|S|}{\sum_{v\in S}\frac{1}{k_v}} \tag{15}$$

B. Metropolis Hastings Random Walk (MHRW)

In this case, equations (4) and (5) trivially yield

$$\widehat{p}_k \;=\; \widehat{q}_k, \qquad \text{and} \tag{16}$$

$$\langle\widehat{k}\rangle \;=\; \sum_k k\,\widehat{p}_k \;=\; \sum_k k\,\widehat{q}_k. \tag{17}$$

C. Graph traversal

From Eq. (11) we know that $p_k(f)$ is proportional to $q_k/(1-(1-t(f))^k)$. Consequently,

$$\widehat{p}_k \;=\; \frac{\widehat{q}_k}{1-(1-t(f))^k} \;\cdot\; \left(\sum_l \frac{\widehat{q}_l}{1-(1-t(f))^l}\right)^{-1} \tag{18}$$

However, in order to evaluate this expression, we need to evaluate $t(f)$, that, in turn, requires p_k. We can solve this chicken-and-egg problem iteratively, if we know the real fraction f^{real} of covered nodes, or equivalently the graph size $|V|$. First, we evaluate Eq.(18) for some values of t and feed the resulting $\widehat{p}_k$'s into Eq. (10) to obtain the corresponding f's. By repeating this process, we can drive the values of f arbitrarily close to f^{real}, and thus find the desired $\widehat{p}_k$.

In summary, for graph traversal techniques, Eq.(18) shows how to estimate the original degree distribution p_k, given the real graph coverage f^{real}. Of course, based on $\widehat{p}_k$, we can calculate the average node degree as $\langle\widehat{k}\rangle = \sum_k k\,\widehat{p}_k$.

VII. Simulation results

In this section, we implement and simulate the considered sampling techniques, namely BFS, DFS, FF (with $p = 0.5$), RW and MHRW. The simulations confirm our analytical results derived for the random graph model $RG(p_k)$. More importantly, in simulations we can study the effect of topological properties, such as of assortativity [28], that are not directly captured by $RG(p_k)$.

A. Random graph $RG(p_k)$

Fig. 3 verifies all the formulae derived in this paper, for the random graph $RG(p_k)$ with a given degree distribution. The analytical expectations are plotted in thick plain lines in the background and the averaged simulation results are plotted in thinner lines lying on top of them. We observe almost a perfect match between theory and simulation in estimating the sampled degree distribution q_k (Fig. 3, right) and its mean $\langle k^*\rangle$ (Fig. 3, left). Indeed, all traversal techniques follow the same curve (as predicted in V-B5), that initially coincides with that of RW (see V-B7) and is monotonically decreasing in f (see V-B8). We also show that degree weighted node sampling without replacements exhibits exactly the same bias (see V-B6). Finally, applying the estimators $\widehat{p}_k$ derived in Section VI corrects for the bias of q_k.

B. The effect of degree-degree correlations (assortativity r)

Depending on the type of network, nodes may tend to connect to similar or different nodes. For example, in most social networks high degree nodes tend to connect to other high degree nodes [28]. Such networks are called *assortative*. In contrast, biological and technological networks are typically *disassortative, i.e.*, they exhibit significantly more high-degree-to-low-degree connections. This observation can be quantified by calculating the *assortativity coefficient* r [28]. Values $r<0$, $r>0$ and $r=0$ indicate disassortative, assortative, and purely random graphs such as $RG(p_k)$, respectively.

For the same initial parameters as in Fig. 3 (p_k, $|V|$), we simulated different levels of assortativity. The results (not presented here due to space constraints) show that for $r > 0$, the degree bias is even stronger than for $r = 0$. This is because the high-degree nodes are now interconnected more densely than in a purely random graph, and are thus easier to discover by sampling techniques that are inherently biased towards high degree nodes. Not surprisingly, a negative assortativity $r < 0$ has the opposite effect: every high-degree node tends to connect to low-degree nodes, which significantly slows down the discovery of the former.

In contrast, random walks RW and MHRW are not affected by the changes in assortativity. This is expected, because their stationary distributions hold for *any* fixed (connected and aperiodic) graph regardless of its topological properties.

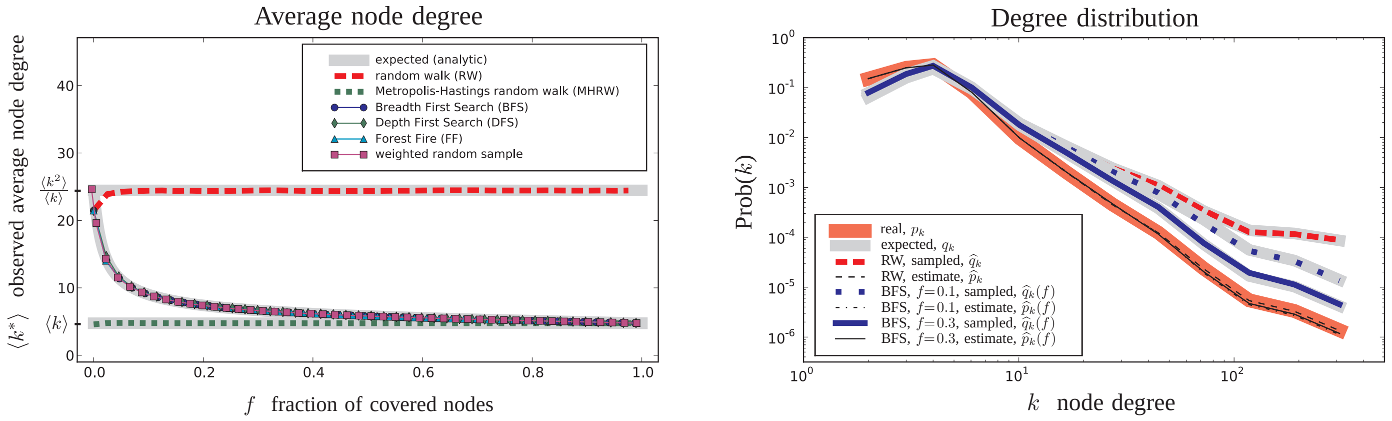

Fig. 3. **Comparison of sampling techniques in theory and in simulation. Left:** Observed (sampled) average node degree $\langle k^* \rangle$ as a function of the fraction f of sampled nodes, for various sampling techniques. The results are averaged over 1000 graphs with 10000 nodes each, generated by the configuration model with a fixed heavy-tailed degree distribution p_k (shown on the right). **Right:** Real, expected, and estimated (corrected) degree distributions for selected techniques and values of f (other techniques behave analogously). We obtained analogous results for other degree distributions and graph sizes $|V|$. The term $\langle k \rangle$ is the real average node degree, and $\langle k^2 \rangle$ is the real average squared node degree.

	UNI	RW	BFS$_{28}$	BFS$_1$	MHRW
$\lvert S \rvert$	982K	2.26M	28×81K = 2.26M	1.19M	2.26M
f	0.44%	1.03%	28×0.04%	0.54%	1.03%

TABLE II
FACEBOOK MEASUREMENTS - DATA SET OVERVIEW.

	UNI	RW	BFS$_{28}$	BFS$_1$	MHRW
$\langle k^* \rangle$ sampled	94.1	338.0	323.9	285.9	95.2
$\langle k^* \rangle$ expected	-	329.8 (3)	329.1 (11)	328.7 (11)	94.1 (5)
$\widehat{\langle k \rangle}$ estimated	-	93.9 (15)	85.4 (18)	72.7 (18)	95.2 (17)

TABLE III
FACEBOOK MEASUREMENTS - AVERAGE NODE DEGREE. IN PARENTHESIS WE GIVE THE RELEVANT EQUATIONS.

VIII. REAL LIFE EXAMPLE: SAMPLING OF FACEBOOK

In this section we apply and test the previous ideas in a real-life large-scale system - the Facebook social graph. With 250+ millions of active users, Facebook is currently the largest online social network. Crawling the entire topology of Facebook would require downloading about $50TB$ of HTML data [5], which makes sampling a very practical alternative.

A. Data collection

We have implemented a set of crawlers to collect the samples of Facebook (FB) according to the UNI, BFS, RW, MHRW techniques. The details of our implementation are described in [5]. The data sets are summarized in Table II.

UNI is a true uniform sample of FB users (see [5] for details). Thanks to a large number of samples $|S|$, UNI gives a high quality estimation of p_k and $\langle k \rangle$. Therefore, we use UNI as ground truth for comparison of various techniques.

We ran two types of BFS crawling. BFS$_{28}$ consists of 28 small BFS-es initiated at 28 randomly chosen nodes from UNI, which allowed us to easily parallelize the process. Moreover, at the time of data collection, we (naively) thought that this would reduce the BFS bias. After gaining more insight (which, nota bene, motivated this paper), we collected a single large BFS$_1$, initiated at a randomly chosen node from UNI.

B. Results

We present the Facebook sampling results in Table III and in Fig. 4. The first row of Table III shows the average node degree $\langle k^* \rangle$ observed (sampled) by several techniques. The value sampled by UNI is $\langle k^* \rangle = 94.1$, which we interpret as the real value $\langle k \rangle$. MHRW, as expected, recovers a similar value. In contrast RW and BFS are both biased towards high degrees by a factor larger than three! The degree bias of RW is the largest. It drops very slightly under the (relatively very short) BFS$_{28}$ crawl, which confirms our findings from V-B7. BFS$_1$, a sample 15 times longer than BFS$_{28}$, is significantly less biased, which is in agreement with V-B8.

The second row shows the *expected* sampled average node degrees (*i.e.*, our predictions of the values in the first row), assuming that the underlying Facebook topology is a random graph $RG(p_k)$ with degree distribution p_k equal to that sampled by UNI. As expected, this works very well for RW. However, the values predicted for BFS overshoot the reality. This is because Facebook *is not* a pure random graph $RG(p_k)$. For example, Facebook, as most social networks, is characterized by a high clustering coefficient c. We believe that it is possible to incorporate this fact in our analytical model, *e.g.*, by appropriately stretching the function $f(t)$ in Eq. (10). This is a main goal in our future work.

Finally, in the last row of Table III we apply the estimators developed in Section VI to correct the degree biases of RW and BFS. In the case of RW, the correction works very well. For the BFS estimator, the results are significantly worse, clearly for the reasons discussed in the previous paragraph.

All the above observations hold also for the *entire* degree distribution, which is shown in Fig. 4.

C. Practical recommendations

BFS is strongly biased toward high degree nodes. It is possible to correct for this bias precisely when the underlying graph is a $RG(p_k)$ (which is not the case in practice). Also, in more realistic graphs, this bias can be corrected reasonably well for a very small sample size (as is the case for BFS$_{28}$), where BFS is similar to RW (see Fig. 1). On the other

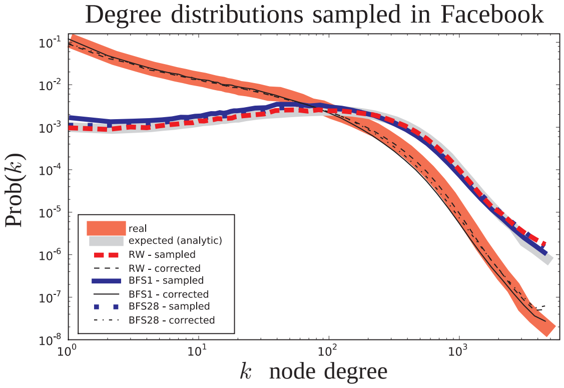

Fig. 4. **Facebook measurements - degree distribution.** Crawlers used: UNI, RW and BFS. All plots are in log-log scale with logarithmic binning of data (we take the average of all points that fall in the same bin). We also correct these distributions, as described in Section VI.

extreme, for very large sampling coverage, the bias of BFS becomes relatively small and could be sometimes neglected (even without additional correction). However, in all other cases, the results become difficult to interpret. In contrast, both RW (equipped with a correction procedure) and MHRW are unbiased, regardless of the actual graph topology. Therefore, we recommend using RW and MHRW (with a slight advantage of RW [3]) as general methods to sample the node properties.

In contrast, RW and MHRW are not useful when sampling *non-local graph properties*, such as the graph diameter or the average shortest path length. In this case, BFS seems very attractive, because it produces a full view of a particular region in the graph, which is usually a densely connected graph itself, and for which the non-local properties can be easily calculated. However, all such results should be interpreted very carefully, as they may be also strongly affected by the bias of BFS. For example, the graph diameter (usually) drops significantly with growing average node degree of a network.

IX. Conclusion and Future Directions

When crawling large, undirected networks, BFS tends to discover high-degree nodes first. As a result, the nodes sampled by an incomplete BFS are biased with respect to their degrees, and also to many other properties that correlate with the degree (see [5] for examples).

To the best of our knowledge, this is the first paper to quantify this bias. In particular, we calculated the node degree distribution expected to be observed by BFS as a function of the fraction f of covered nodes, in a random graph $RG(p_k)$ with a given degree distribution p_k. Furthermore, we also showed that, for $RG(p_k)$, all commonly used graph traversal techniques that sample nodes without replacement lead to the same bias, and we showed how to correct for this bias.

We compared our analytical results (obtained for random graphs) with experimental results obtained via BFS sampling of the Facebook social graph (which is not expected to be a random graph). Qualitatively, the same trends hold: For a small sample size, $f \rightarrow 0$, BFS has the same bias as the classic Random Walk, and with increasing f, the bias monotonically decreases. Quantitatively, however, the bias of our Facebook samples does not match precisely the one we calculated for $RG(p_k)$. This is the effect of topological properties other than the degree distribution, in which Facebook and the random graph vary. For example, we found in simulations that the degree bias gets amplified in graphs with strong positive assortativity. In future work, we plan to extend our theoretical framework to capture such topological properties as well.

References

[1] M. R. Henzinger, A. Heydon, M. Mitzenmacher, and M. Najork, "On near-uniform url sampling," in *Proc. of WWW*, 2000.

[2] D. Stutzbach, R. Rejaie, N. Duffield, S. Sen, and W. Willinger, "On unbiased sampling for unstructured peer-to-peer networks," in *IMC*, 2006.

[3] A. Rasti, M. Torkjazi, R. Rejaie, N. Duffield, W. Willinger, and D. Stutzbach, "Respondent-driven sampling for characterizing unstructured overlays," in *INFOCOM Mini-Conference*, April 2009.

[4] C. Gkantsidis, M. Mihail, and A. Saberi, "Random walks in peer-to-peer networks," in *Proc. of Infocom*, 2004.

[5] M. Gjoka, M. Kurant, C. T. Butts, and A. Markopoulou, "Walking in facebook: A case study of unbiased sampling of osns," *Infocom*, 2010.

[6] B. Krishnamurthy, P. Gill, and M. Arlitt, "A few chirps about twitter," in *Proc. of WOSN*, 2008.

[7] J. Leskovec and C. Faloutsos, "Sampling from large graphs," in *Proc. of ACM SIGKDD*, 2006.

[8] L. Lovasz, "Random walks on graphs. a survey," in *Combinatorics*, 1993.

[9] M. Najork and J. L. Wiener, "Breadth-first search crawling yields high-quality pages," in *Proc. of WWW*, 2001.

[10] Y. Ahn, S. Han, H. Kwak, S. Moon, and H. Jeong, "Analysis of Topological Characteristics of Huge Online Social Networking Services," in *Proc. of WWW*, 2007.

[11] A. Mislove, M. Marcon, K. P. Gummadi, P. Druschel, and S. Bhattacharjee, "Measurement and Analysis of Online Social Networks," in *Proc. of IMC*, 2007.

[12] C. Wilson, B. Boe, A. Sala, K. P. Puttaswamy, and B. Y. Zhao, "User interactions in social networks and their implications," in *Proc. of EuroSys*, 2009.

[13] S. H. Lee, P.-J. Kim, and H. Jeong, "Statistical properties of sampled networks," *Phys. Rev. E*, vol. 73, p. 016102, 2006.

[14] L.Becchetti, C.Castillo, D.Donato, and A.Fazzone, "A comparison of sampling techniques for web graph characterization," in *LinkKDD*, 2006.

[15] S. Ye, J. Lang, and F. Wu, "Crawling Online Social Graphs," in *APWeb*, 2010.

[16] A. Mislove, H. S. Koppula, K. P. Gummadi, P. Druschel, and B. Bhattacharjee, "Growth of the flickr social network," in *WOSN*, 2008.

[17] J. H. Kim, "Poisson cloning model for random graphs," *International Congress of Mathematicians (ICM)*, 2006 (preprint in 2004).

[18] D. Achlioptas, A. Clauset, D. Kempe, and C. Moore, "On the bias of traceroute sampling: or, power-law degree distributions in regular graphs," in *STOC*, 2005.

[19] J. Illenberger, G. Flötteröd, , and K. Nage, "An approach to correct bias induced by snowball sampling," *Sunbelt Social Networks Conference*, 2009.

[20] L. Goodman, "Snowball sampling," *Annals of Mathematical Statistics*, vol. 32, p. 148170, 1961.

[21] W. Gilks, S. Richardson, and D. Spiegelhalter, *Markov Chain Monte Carlo in Practice*. Chapman and Hall/CRC, 1996.

[22] D. Heckathorn, "Respondent-driven sampling: A new approach to the study of hidden populations," *Social Problems*, vol. 44, p. 174199, 1997.

[23] M. Salganik and D. Heckathorn, "Sampling and estimation in hidden populations using respondent-driven sampling," *Sociological Methodology*, vol. 34, p. 193239, 2004.

[24] J. Leskovec, J. Kleinberg, and C. Faloutsos, "Graphs over time: densification laws, shrinking diameters and possible explanations," in *KDD*, 2005.

[25] M. Molloy and B. Reed, "A critical point for random graphs with a given degree sequence," pp. 161–179, 1995.

[26] M. E. J. Newman, "Ego-centered networks and the ripple effect," *Social Networks*, vol. 25, pp. 83–95, 2003.

[27] R. Motwani and P. Raghavan, *Randomized Algorithms*. Cambridge University Press, 1990.

[28] M. Newman, "Assortative mixing in networks," in *Phys. Rev. Lett. 89*, 2002.