Estimating and Sampling Graphs with Multidimensional Random Walks

Bruno Ribeiro
Computer Science Department
University of Massachusetts at Amherst
Amherst, MA, 01002
ribeiro@cs.umass.edu

Don Towsley
Computer Science Department
University of Massachusetts at Amherst
Amherst, MA, 01002
towsley@cs.umass.edu

ABSTRACT

Estimating characteristics of large graphs via sampling is a vital part of the study of complex networks. Current sampling methods such as (independent) random vertex and random walks are useful but have drawbacks. Random vertex sampling may require too many resources (time, bandwidth, or money). Random walks, which normally require fewer resources *per sample*, can suffer from large estimation errors in the presence of disconnected or loosely connected graphs. In this work we propose a new m-dimensional random walk that uses m *dependent* random walkers. We show that the proposed sampling method, which we call *Frontier sampling*, exhibits all of the nice sampling properties of a regular random walk. At the same time, our simulations over large real world graphs show that, in the presence of disconnected or loosely connected components, *Frontier sampling* exhibits lower estimation errors than regular random walks. We also show that *Frontier sampling* is more suitable than random vertex sampling to sample the tail of the degree distribution of the graph.

Categories and Subject Descriptors

G.3 [**Probability and Statistics**]: Statistical computing

General Terms

Experimentation

Keywords

Frontier Sampling, Random Walks, MCMC, Estimates, Power Laws, Assortativity, Global Clustering Coefficient

1. INTRODUCTION

A number of recent studies [7, 11, 15, 19, 20, 26, 30, 29, 36] (to cite a few) are dedicated to the characterization of complex networks. A complex network is a network with non-trivial topological features (features that do not occur in simple networks such as lattices or random networks). Examples of such networks include the Internet, the World Wide Web, social, business, and biological networks [7, 28]. This work represents a complex network as a directed graph with labeled vertices and edges. A label can be, for instance, the degree of a vertex or, in a social network setting, someone's hometown. Examples of network characteristics include the degree distribution, the fraction of HIV positive individuals in a population [24], or the average number of copies of a file in a peer-to-peer (P2P) network [16].

Characterizing the labels of a graph requires querying vertices and/or edges; each query has an associated cost in resources (time, bandwidth, money). Characterizing a large graph by querying the whole graph is often too costly. As a result, researchers have turned their attention to the estimation of graph characteristics based on incomplete (sampled) data. In this work we present a new tool, *Frontier Sampling*, to characterize complex networks. In what follows *random vertex (edge) sampling* refers to sampling vertices (edges) independently and uniformly at random (with replacement).

Distinct sampling strategies have different resource requirements depending on the network being sampled. For instance, in a network where each vertex is assigned a unique user-id (e.g., travelers and their passport numbers, Facebook, MySpace, Flickr, and Livejournal) it is a widespread practice to perform random vertex sampling by querying randomly generated user-ids. This approach can be resource-intensive if the user-id space is sparsely populated as the hit-to-miss ratio is low (e.g., less than 10% of all MySpace user-ids between the highest and lowest valid user-ids are currently occupied [30]). Another way to sample a network is by querying edges instead of vertices. Randomly sampling edges can be harder than randomly sampling vertices if edges are not be associated to unique IDs (or if edge IDs cannot be randomly queried). We summarize some drawbacks of random vertex and edge sampling:

- Random edge sampling may be impractical when edges cannot be randomly queried (e.g., online social networks like Facebook [15], MySpace [30], and Twitter or a P2P network like Bittorrent).

- Random vertex sampling may be undesirable when user-ids are sparsely populated (low hit-to-miss ratio) and queries are subject to resource constraints (e.g., queries are rate-limited in Flickr, Livejournal [26], and Bittorrent [18]). In a P2P network like Bittorrent, a client can randomly sample peers (vertices) by querying a tracker (server); however, trackers may rate-limit client queries [18].

- Even when random vertex sampling is not severely resource-constrained, some characteristics may be better estimated with random edge sampling (e.g., the tail of the degree distribution of a graph).

An alternative, and often cheaper, way to sample a network is by means of a random walk (RW). A RW samples a graph by moving a particle (walker) from a vertex to a neighboring vertex (over an edge). By this process edges and vertices are sampled. The probability by which the random walker selects the next neighboring vertex determines the probability by which vertices and edges are sampled. In this work we are interested in random walks that sample *edges* uniformly. The edges sampled by RW can then be used to obtain unbiased estimates of a variety of graph characteristics (we present two examples in Section 4).

In this work we assume that a random walker has the ability to query a vertex to obtain all of its incoming and outgoing edges (Section 4 details the reason behind this assumption). This is possible for online networks such as Twitter, LiveJournal [26], YouTube [26], Facebook [15], MySpace [30], P2P networks [29], and the arXiv citations network. We revisit the theory behind random walks in Section 4.

Sampling graphs with random walks is not without drawbacks. The accuracy of the estimates depends not only on the graph structure but also on the characteristic being estimated. The graph structure can create distortions in the estimates by "trapping" the random walker inside a subgraph. An extreme case happens when the graph consists of two or more disconnected components (subgraphs). For instance, wireless mobile social networks exhibit connection graphs with multiple disconnected components [11]. But even connected graphs can suffer from the same problem. A random walker can get "temporarily trapped" and spend most of its sampling budget exploring the local neighborhood near where it got "trapped". In the above scenarios estimates may be inaccurate if the characteristics of the local neighborhood differ from the overall characteristic of the graph. This problem is well documented (see [21]) and our goal is to mitigate it.

Contributions

This work proposes a new m-dimensional random walk sampling method (*Frontier sampling*) that, starting from a collection of m randomly sampled vertices, preserves all of the important statistical properties of a regular random walk (e.g., vertices are visited with a probability proportional to their degree). While the vertices are visited with a probability proportional to their degree, we show that the joint steady state distribution of Frontier Sampling (the joint distribution of all m vertices) is closer to uniform (the starting distribution) than that of m independent random walkers, for any $m > 0$. This property has the potential to dramatically reduce the transient of random walks.

In our simulations using real world graphs we see that Frontier Sampling mitigates the large estimation errors caused by disconnected or loosely connected components that can "trap" a random walker and distort the estimated graph characteristic, i.e., Frontier sampling (FS) estimates have smaller Mean Squared Errors (MSEs) than estimates obtained from regular random walkers (single and multiple independent walkers, reviewed in Section 4.4) in a variety of scenarios.

We make two additional contributions: (1) we compare random walk-based estimates to those obtained from random vertex and random edge sampling. We show analytically that the tail of the degree distribution is better estimated using random edge sampling than random vertex sampling. We observe from simulations over real world networks (in Section 6.4) that FS accuracy is comparable to the accuracy of random edge sampling. These results help explain recent empirical results [29]; (2) we present asymptotically unbiased estimators using the edges sampled by a RW for the assortative mixing coefficient (defined in Section 4.2.2) and the global clustering coefficient (defined in Section 4.2.4).

Outline

The outline of this work is as follows. Section 2 presents the notation used in this paper. Section 3 contrasts random vertex with random edge sampling. Section 4 revisits single and multiple independent random walk sampling and estimation. Section 5 introduces *Frontier Sampling* (FS), a sampling process that uses m dependent random walkers in order to mitigate the high estimation errors caused by disconnected or loosely connected components. Section 5 also shows that FS can be seen as an m-dimensional random walk over the m-th Cartesian power of the graph (formally defined in Section 5). In Section 6 we see that FS outperforms both single and multiple independent random walkers in a variety of scenarios. We also compare (independent) random vertex and edge sampling with FS. Section 7 reviews the relevant literature. Finally, Section 8 presents our conclusions and future work.

2. DEFINITIONS

In what follows we present some definitions. Let $G_d = (V, E_d)$ be a labeled directed graph representing the (original) network graph, where V is a set of vertices and E_d is a set of ordered pairs of vertices (u, v) representing a connection from u to v (a.k.a. edges). We assume that each vertex in G_d has at least one incoming or outgoing edge. The in-degree of a vertex u in G_d is the number of distinct edges $(v_1, u), \ldots, (v_i, u)$ into u, and its out-degree is the number of distinct edges $(u, v_1), \ldots, (u, v_j)$ out of u. Some complex networks can be modeled as undirected graphs. In this case, when the original graph is undirected, we model G_d as a symmetric directed graph, i.e., $\forall (u, v) \in E_d, (v, u) \in E_d$.

Let $\mathcal{L}_v$ and $\mathcal{L}_e$ be a finite set of vertex and edge labels, respectively. Each edge $(u, v) \in E_d$ is associated with a set of labels $\mathcal{L}_e(u, v) \subseteq \mathcal{L}_e$. For instance, the label of edge (u, v) can be the in-degree of v in G_d. Similarly, we can associate a set of labels to each vertex, $\mathcal{L}_v(v) \subseteq \mathcal{L}_v, \forall v \in V$. Some edges and vertices may not have labels. If edge (u, v) is unlabeled then $\mathcal{L}_e(u, v) = \emptyset$. Similarly, if vertex v is unlabeled then $\mathcal{L}_v(v) = \emptyset$.

When performing a random walk, we assume that a random walker has the ability to retrieve incoming and outgoing edges from a queried vertex (and vertices are distinguishable). With this assumption we are able to build (on-the-fly) a symmetric directed graph while walking over G_d. Let $G = (V, E)$ be the symmetric counterpart of G_d, i.e.,

$$E = \bigcup_{\forall (u,v) \in E_d} \{(u, v), (v, u)\}.$$

Note that G may not be connected. As G is symmetric, we denote by $\deg(v)$ to be the in-degree or the out-degree of $v \in V$ as they are equal. Let $\mathrm{vol}(S) = \sum_{\forall v \in S} \deg(v)$, $\forall S \subseteq V$, denote the volume of the vertices in S.

Let $\hat{\theta}_l$ be the estimated fraction of *vertices* with label l obtained by some estimator. The two error metrics used in most of our examples are the normalized root mean square error of $\hat{\theta}_l$, which is a normalized measure of the dispersion of the estimates, defined as

$$\mathrm{NMSE}(l) = \frac{\sqrt{E[(\hat{\theta}_l - \theta_l)^2]}}{\theta_l}. \tag{1}$$

and the normalized root mean square error of the Complementary Cumulative Distribution Function (CCDF) $\gamma = \{\gamma_l\}$, where $\gamma_l = \sum_{k=l+1}^{\infty} \theta_k$, defined as

$$\mathrm{CNMSE}(l) = \frac{\sqrt{E[(\hat{\gamma}_l - \gamma_l)^2]}}{\gamma_l}. \tag{2}$$

For the sake of simplicity, and unless stated otherwise, in the remainder of this paper we assume that all queries of edges and vertices have unitary cost and that we have a fixed sampling budget B.

3. VERTEX V.S. EDGE SAMPLING

We consider a straightforward estimation problem to illustrate a tradeoff between random edge and random vertex sampling. Consider the problem of estimating the out-degree distribution of G_d. Let θ_i be the fraction of vertices with out-degree $i > 0$ and d be the average out-degree. Let the label of vertex u, $\mathcal{L}_v(u)$, be the out-degree of u. We assume that d is known; also assume that from an edge (u, v) we can query $\mathcal{L}_v(u)$. In random edge sampling the probability of sampling a vertex with out-degree i is proportional i: $\pi_i = i\theta_i/d$. On the other hand, random vertex sampling samples a vertex with out-degree i with probability θ_i. A straightforward calculation shows that the NMSE (equation (1)) of B randomly sampled *edges* with out-degree i is

$$\mathrm{NMSE}(i) = \sqrt{(1/\pi_i - 1)/B}, \quad i > 0. \tag{3}$$

Similarly, the $\mathrm{NMSE}(i)$ for random vertex sampling is

$$\mathrm{NMSE}(i) = \sqrt{(1/\theta_i - 1)/B}. \tag{4}$$

Now note that $\pi_i/\theta_i = i/d$, which means that $\pi_i > \theta_i$ if $i > d$ and $\pi_i < \theta_i$ if $i < d$. From equations (3) and (4) we see that random edge sampling more accurately estimates degrees larger than the average ($i > d$) while random vertex sampling more accurately estimates degrees smaller than the average ($i < d$). This means random edge sampling exhibits smaller NMSE when estimating the tail of the out-degree distribution.

Above we have seen that random edge sampling is more accurate than random vertex sampling in estimating the tail of the out-degree degree distribution. A similar result happens with the in-degree distribution and the degree distribution of undirected networks. The above analysis explains real world experiments [29]. Unfortunately, as discussed in Section 1, random edge sampling is rarely practical. In what follows we see that, if G is connected, random walks exhibit similar statistical properties to random edge sampling.

4. RANDOM WALK SAMPLING

In this section we review random walk (RW) sampling and estimation over a non-bipartite, connected, directed, symmetric graph G. Sampling G with a RW is straightforward. The random walker has a sampling budget B and starts at vertex $v_0 \in V$. For the sake of simplicity, unless stated otherwise, we consider that all queries to vertices have unit cost and that we have a fixed sampling budget B.

Let $\{(u_i, v_i)\}_{i=1}^{B}$ be the a sequence of edges sampled by a RW, where $u_i = v_{i-1}$, $i = 2, \ldots, B$. Note that edges may be sampled multiple times. We refer to (u_i, v_i) as the i-th sampled edge. At the i-th step a walker at vertex v_i chooses an outgoing edge (v_i, u_i) uniformly at random from the set of outgoing edges of v_i and adds (v_i, u_i) to the sequence of sampled edges. At step $i + 1$ the random walker starts at vertex u_i and the sampling continues until $i = B$.

The RW described here is the most common type of RW found in the literature [22]. Other types of random walks differ in the way in which outgoing edges are sampled. The Metropolis-Hastings RW [38] is an example of a random walk that samples *vertices* (not edges) uniformly at random. However, experiments estimating a variety of metrics indicate that Metropolis-Hastings RW is less accurate than the random walk described in this work [15, 29]. For more details about other types of RW please refer to [32, Chapter 7].

An important property of a RW is its ability to reach a unique stationary regime. A necessary condition for stationarity is that G must be symmetric, connected, and non-bipartite (the non-bipartite assumption can be relaxed in a lazy random walk [22]). In a stationary RW, the sequence of sampled edges is a stationary sequence. A sequence $X_1, X_2, \ldots$ of random variables is said to be stationary if for any positive integers n and k, the joint distribution of $(X_n, \ldots, X_{n+k})$ is independent of n. Once the RW reaches steady state, it also shares two important properties with random edge (RE) sampling. First, both RW and RE sample edges uniformly at random [22], which means that the probability that a vertex v is sampled is $\deg(v)/\mathrm{vol}(V)$. Second, both RW and RE obey the strong law of large numbers, as we see next.

4.1 Strong Law of Large Numbers

The following variation of the strong law of large numbers is a powerful tool to build (asymptotically) unbiased estimators of graph characteristics. We provide a trivial extension of a well known result [25, Theorem 17.2.1] to the case where we are interested in a subset of the graph edges. Let $E^\star \subseteq E$ be non-empty. Let (u_i, v_i) be the i-th RW sampled edge such that $(u_i, v_i) \in E^\star$; and let $B^\star(B)$ be the number of such samples, where B is the number of RW steps. $B^\star(B)$ is a random variable that represents the number of RW sampled edges that belong to $E^\star$. Note that $B^\star(B) \leq B$.

Theorem 4.1 (SLLN). *For any function f, where* $\sum_{(u,v) \in E^\star} |f(u, v)| < \infty$,

$$\lim_{B \to \infty} \frac{1}{B^\star(B)} \sum_{i=1}^{B^\star(B)} f(u_i, v_i) \to \frac{1}{|E^\star|} \sum_{\forall (u,v) \in E^\star} f(u, v),$$

almost surely, i.e., the event occurs with probability one.

PROOF. Let

$$h(u, v) = \begin{cases} 1 & \text{if } (u, v) \in E^\star, \text{ and} \\ 0 & \text{otherwise.} \end{cases}$$

As the RW is stationary (and edges are sampled uniformly)

$$\lim_{B \to \infty} \frac{\sum_{i=1}^{B} f(u_i, v_i) h(u_i, v_i)}{\sum_{i=1}^{B} h(u_i, v_i)} \to \frac{\sum_{\forall (u,v) \in E} f(u, v) h(u, v)}{\sum_{\forall (u,v) \in E} h(u, v)},$$

almost surely [25, Theorem 17.2.1]. The proof follows from noting that $B^\star(B) = \sum_{i=1}^{B} h(u_i, v_i)$ and that $h(u, v) = 0, \forall (u, v) \in E \backslash E^\star$. □

Theorem 4.1 allows us to construct estimators of graph characteristics that converge to their true values as the number of RW samples goes to infinity ($B \to \infty$). If we are trying to estimate vertex labels we set $E^\star = E$ and $B^\star = B$. In what follows we apply Theorem 4.1 to estimate graph characteristics; we also present four examples of estimators.

4.2 Estimators

An *estimator* is a function that takes a sequence of observations (sampled data) as input and outputs an estimate of a unknown population parameter (graph characteristic). In this section we see how we can estimate graph characteristics using the edges sampled by a RW.

We present estimators of the following four graph characteristics: the edge label density (the fraction of edges with a given label in the graph), the assortative mixing coefficient [27], the vertex label density, and the global clustering coefficient [34]. Designing these estimators is straightforward:

(1) First we find a function f that computes the characteristic of G using E;

(2) then we replace E with the sequence of edges sampled by a stationary RW.

In what follows we illustrate how to build an estimator of the edge label density.

4.2.1 Edge Label Density

We seek to estimate the fraction of *edges* with label $l \in \mathcal{L}_e$ in G_d among all edges (u, v) that have labels, i.e., $\mathcal{L}_e(u, v) \neq \emptyset$. Edge labels can be anything, from social networking labels to the amount of IP traffic over each link in a computer network. An edge label can be, for instance, a tuple $(\text{outdeg}(u), \text{indeg}(v))$ where $\text{outdeg}(u)$ is the out-degree of u and $\text{indeg}(v)$ is the in-degree of v in the original graph G_d.

For now we assume that we know E. Let $E^\star$ be the non-empty subset of E for which there are labels. Let p_l denote the fraction of edges in $E^\star$ with label l; it is clear that

$$p_l = \sum_{\forall (u,v) \in E^\star} \frac{\mathbf{1}(l \in \mathcal{L}_e(u, v))}{|E^\star|},$$

where

$$\mathbf{1}(l \in \mathcal{L}_e(u, v)) = \begin{cases} 1 & \text{if } l \in \mathcal{L}_e(u, v), \\ 0 & \text{otherwise.} \end{cases}$$

Let $B^\star(B)$ be the number of RW sampled edges that belong to $E^\star$ and (u_i, v_i) be the i-th of such edges. Replacing $E^\star$

with the edges in $E^\star$ sampled by a stationary RW gives the following estimator

$$\hat{p}_l \equiv \sum_{i=1}^{B^\star(B)} \frac{\mathbf{1}(l \in \mathcal{L}_e(u_i, v_i))}{B^\star(B)}. \tag{5}$$

It follows directly from Theorem 4.1 (with $f(u, v) = \mathbf{1}(l \in \mathcal{L}_e(u, v))$) that $\lim_{B \to \infty} \hat{p}_l \overset{\text{a.s.}}{\to} p_l$. Moreover, from the linearity of expectation, $E[\hat{p}_l] = p_l$ for all values of $B^\star(B) > 0$.

4.2.2 Assortative Mixing Coefficient

The assortative mixing coefficient [27] is a measure of the correlation of labels between two neighboring vertices. By appropriately assigning edge labels derived from vertex labels, we can use the density estimator of equation (5) to derive an estimator of the assortative mixing coefficient. In order to simplify our exposition, we restrict our analysis to the assortative mixing of vertex degrees in a directed graph (equation (25) of [27]). It is trivial to extend our analysis to other types of assortative mixing coefficients, e.g., equations (21) and (23) of [27].

Let $(\text{outdeg}(u), \text{indeg}(v))$ denote the label of a directed edge (u, v) in G that also exists in G_d; and let $E^\star$ be the set of all such edges ($E^\star = E_d$). Let p_{ij} denote the fraction of labeled edges with label (i, j). Let W_{out} (W_{in}) denote the maximum *observed* out-degree (in-degree) of G_d in the RW. The degree assortative mixing coefficient [27] of a directed graph can be estimated using

$$\hat{r} \equiv \frac{1}{\hat{\sigma}_{\text{in}} \, \hat{\sigma}_{\text{out}}} \sum_{i=0}^{W_{\text{out}}} \sum_{j=0}^{W_{\text{in}}} ij(\hat{p}_{ij} - \hat{q}_i^{\text{out}} \hat{q}_j^{\text{in}}),$$

where

$$\hat{p}_{ij} \equiv \sum_{k=1}^{B^\star(B)} \frac{\mathbf{1}(\text{outdeg}(u_k) = i \,, \, \text{indeg}(v_k) = j)}{B^\star(B)};$$

$$\hat{q}_i^{\text{out}} \equiv \sum_{k=0}^{W_{\text{in}}} \hat{p}_{ik} \quad ; \quad \hat{q}_j^{\text{in}} \equiv \sum_{k=0}^{W_{\text{out}}} \hat{p}_{kj} \quad ;$$

$$\hat{\sigma}_{\text{in}} = \sqrt{\sum_{i=0}^{W_{\text{out}}} j^2 \hat{q}_j^{\text{in}} - \left(\sum_{i=0}^{W_{\text{out}}} j \hat{q}_j^{\text{in}} \right)^2} \quad ; \quad \text{and}$$

$$\hat{\sigma}_{\text{out}} = \sqrt{\sum_{i=0}^{W_{\text{in}}} i^2 \hat{q}_i^{\text{in}} - \left(\sum_{i=0}^{W_{\text{in}}} i \hat{q}_i^{\text{out}} \right)^2}$$

where $\hat{\sigma}_{\text{in}}$ and $\hat{\sigma}_{\text{out}}$ are the standard deviation of the distribution $\hat{q}_i^{\text{in}}$ ($\hat{q}_j^{\text{out}}$). As the estimate $\hat{p}_{ij}$ (equation (5)) asymptotically converges almost surely to its true value, it is trivial to show that $\hat{q}_i^{\text{in}}$, $\hat{q}_j^{\text{out}}$, $\hat{\sigma}_{\text{in}}$, and $\hat{\sigma}_{\text{out}}$ also asymptotically converge almost surely to their true values. Thus, $\hat{r}$ asymptotically converges, almost surely, to the true assortative mixing coefficient of [27], as long as $\sigma_{\text{in}} > 0$ and $\sigma_{\text{out}} > 0$. This implies that $\hat{r}$ is an asymptotically unbiased estimator of the assortative mixing coefficient of G_d.

4.2.3 Vertex Label Density

Let $\mathcal{L}_v(v)$ be the set of labels associated with vertex v, $\forall v \in V$. The fraction of *vertices* with label l in G, θ_l, is

$$\theta_l = \frac{1}{|V|} \sum_{\forall(u,v)\in E} \frac{\mathbf{1}(l \in \mathcal{L}_v(v))}{\deg(v)}, \qquad (6)$$

as $G = (V, E)$ is directed and symmetric. By replacing E with a sequence of edges sampled by a stationary RW (here we have $E^\star = E$ and $B^\star = B$) and renormalizing, we arrive at the following estimator for θ_l

$$\hat{\theta}_l \equiv \frac{1}{SB} \sum_{i=1}^{B} \frac{\mathbf{1}(l \in \mathcal{L}_v(v_i))}{\deg(v_i)}, \qquad (7)$$

where $S = 1/B \sum_{i=1}^{B} 1/\deg(v_i)$. From Theorem 4.1 we have $\lim_{B\to\infty} S \to |V|/|E|$, almost surely. Using again Theorem 4.1 we have

$$\lim_{B\to\infty} \frac{1}{B} \sum_{i=1}^{B} \frac{\mathbf{1}(l \in \mathcal{L}_v(v_i))}{\deg(v_i)} \to \frac{1}{|E|} \sum_{\forall(u,v)\in E} \frac{\mathbf{1}(l \in \mathcal{L}_v(v))}{\deg(v)},$$

almost surely, which divided by $|V|/|E|$ yields equation (6). As S converges almost surely to $|V|/|E|$, we have $\lim_{B\to\infty} \hat{\theta}_l \to \theta_l$, almost surely. This also implies that $\hat{\theta}_l$ is an asymptotically unbiased estimator of θ_l.

4.2.4 Global Clustering Coefficient

In the literature the term *clustering coefficient* often refers to the local clustering coefficient [37]. In our example we estimate a different metric: the *global* clustering coefficient. In a social network the global clustering coefficient, C, is the probability that the friend of John's friend is also John's friend [34]. Let $V^\star$ be the set of vertices $v \in V$ with $\deg(v) > 1$. The global clustering coefficient of an undirected graph is defined as [34]

$$C \equiv \frac{1}{|V^\star|} \sum_{\forall v \in V} c(v), \qquad (8)$$

where

$$c(v) = \begin{cases} \Delta(v)/\binom{\deg(v)}{2} & \text{if } \deg(v) \geq 2 \\ 0 & \text{otherwise}, \end{cases}$$

where $\Delta(v) = |\{(u,w) \in E : (v,u) \in E \text{ and } (v,w) \in E\}|$ is the number of triangles that contain vertex v and $\binom{\deg(v)}{2}$ is the maximum number of triangles that a vertex v with degree $\deg(v)$ can belong to.

Note that finding $\Delta(v)$ for a given vertex $v \in V$ requires knowing all vertices within two hops of v, which can be a resource intensive task. To avoid the cost of computing Δ, we rewrite equation (8)

$$C = \frac{1}{|V^\star|} \sum_{\forall(v,u)\in E} \frac{f(v,u)}{\binom{\deg(v)}{2}}, $$

where $f(v,u)$ gives the number of shared neighbors between u and v.

Let (v_i, u_i) be the i-th sampled edge in a stationary RW and let

$$\hat{C} \equiv \frac{1}{SB} \sum_{i=1}^{B} \frac{f(v_i, u_i)}{\binom{\deg(v_i)}{2}} \frac{1}{\deg(v_i)},$$

where

$$S = \frac{1}{B} \sum_{i=1}^{B} \frac{1}{\deg(v_i)}.$$

Corollary 4.2. $\lim_{B\to\infty} \hat{C} \to C$, *almost surely.*

PROOF. From Theorem 4.1

$$\lim_{B\to\infty} S \to |V^\star|/|E|,$$

almost surely. Also from Theorem 4.1

$$\lim_{B\to\infty} \frac{1}{B} \sum_{i=1}^{B} \frac{f(v_i, u_i)}{\binom{\deg(v_i)}{2}} \frac{1}{\deg(v_i)} \to \frac{1}{|E|} \sum_{\forall(v,u)\in E} \frac{f(v,u)}{\binom{\deg(v)}{2}},$$

almost surely, which together with the almost sure convergence of S implies that $\lim_{B\to\infty} \hat{C} \to C$, almost surely. $\qquad\square$

Note that almost sure convergence implies that $\hat{C}$ is an asymptotically unbiased estimator of C.

4.3 Estimator Accuracy & Graph Structure

Sampling a graph using a RW is not without drawbacks. A random walker can get (temporarily) "trapped" inside a subgraph whose characteristics differ from those of the whole graph. Even if the random walker starts in steady state (i.e., is stationary), this scenario may increase the mean squared error of the estimates. If the random walker does not start in steady state, this scenario may cause an increase in the estimation bias as well as the mean squared error. Ideally, the random walker needs to mitigate the effect of these traps on the estimates.

The above two types of estimation errors are well documented in the literature and various solutions are available [14]. For instance, if the random walker does not start in a stationary regime (transient), it is common practice to discard the first w samples [14]. The value of w is called the *burn-in period*. There are two problems with this solution: (1) it only reduces the error related to the non-stationarity of the samples; (2) it is difficult to determine a good value for w if the sampling budget is small (compared to the size of the graph) and the size and structure of G are unknown.

A simple naive solution to the RW "trapping" problem (adopted in [15] to sample Facebook), is to sample the graph using multiple independent random walkers [14]. In what follows we see that this naive approach can lead to increased estimation errors. In Section 5 we propose a method to mitigate the random walk "trapping" problem using m *dependent* random walkers.

4.4 Multiple Independent Random Walkers

The main problem of estimating graph characteristics using a single walker is that the walker may get trapped inside a local neighborhood. But there is the question of what happens if we could start m independent random walkers (*MultipleRW*) at m independently sampled vertices in the graph. Note that when $m = 1$ we are back to sampling G using a single random walker, which we denote as *SingleRW*. Networks such as MySpace, Facebook, and Bittorrent admit random (uniform) vertex sampling at cost c higher than the cost of sampling the neighbors of a known vertex (which is what a RW does). In such networks random vertex sampling may help us start m random walkers at different parts of the

graph. While the value of c can be large, initializing m random walkers with uniformly sampled vertices costs (only) mc units of our sampling budget (where one unity of the budget is the cost of sampling a vertex in a RW).
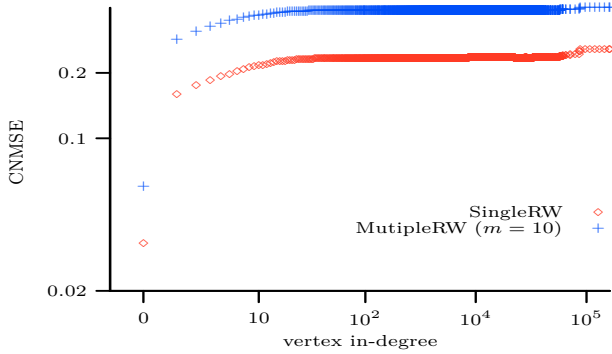

Figure 1: (Flickr) The log-log plot of the CNMSE of the in-degree distribution estimates with budget $B = |V|/10$.

Unfortunately, m independent random walkers starting at m randomly sampled vertices may decrease estimation accuracy. Consider the following experiment where each of the m random walkers (independently) performs $\lfloor B/m - c \rfloor$ steps. We seek to estimate the CCDF (complementary cumulative distribution function) of the in-degree of the Flickr graph (the Flickr dataset is summarized in Table 1). According to Table 1 the Flickr graph is disconnected. The goal of this simulation is to compare the estimation accuracy of SingleRW and MultipleRW when there are no disconnected components. For this we set $c = 1$. The sampling budget is $B = 171,525 = |V|/10$, which amounts to a sampling budget equivalent to 10% of the vertices in the graph. Figure 1 shows a log-log plot of the CNMSE, equation (2), of SingleRW and MultipleRW ($m = 10$) averaged over $10,000$ runs. Note that the estimates obtained by SingleRW are, on average, more accurate than the estimates obtained by MultipleRW. Increasing the sampling budget B does not reduce the gap. In Section 6 we see, over other real-world graphs, that when starting random walkers from uniformly sampled vertices, MultipleRW has higher estimation errors than SingleRW.

4.5 Disconnected Graph Example

The following example shows a situation in which both MultipleRW and SingleRW have large estimation errors. In this example we initialize MultipleRW with m randomly (uniformly) sampled vertices. We simplify our exposition by assuming that each MultipleRW walker takes B/m steps, where B (the sampling budget) is a multiple of m. Let $G = (V, E)$ be an undirected graph that has two large disconnected components $G_A = (V_A, E_A)$ and $G_B = (V_B, E_B)$. Let $|V_A| = |V_B|$ and $\mathrm{vol}(V_A) > \mathrm{vol}(V_B)$. When initial vertices are uniformly sampled, the probability that each MultipleRW walker (independently) starts in G_A (G_B) is $h_A = |V_A|/|V|$ ($h_B = |V_B|/|V|$). Recall that G_A and G_B are disconnected. For each random walker, after B/m ($B \gg 1$) RW steps, an edge $(u_A, v_A) \in E_A$ is sampled with probability $p_A \approx h_A/\mathrm{vol}(V_A)$. Similarly an edge $(u_B, v_B) \in E_B$ is sampled with probability $p_B \approx h_B/\mathrm{vol}(V_B)$. Thus $p_A < p_B$, i.e., the edges in G_B are sampled with higher probability than the edges in G_A. As our estimators assume that all edges are sampled with the same probability, this imbal-

ance between p_A and p_B has the potential to introduce large MSEs (and biases). Note that increasing m does not change p_A and p_B. Increasing B only mitigates this problem if G is connected and, in a loosely connected graph, only large values of B positively impact the MSE. Ideally we want a RW algorithm that does not rely on large sampling budgets B to achieve low estimation errors.

Now consider the same thought experiment where each random walker starts in G_A and G_B (independently) with probabilities $h_A = \mathrm{vol}(V_A)/\mathrm{vol}(V)$ and $h_B = \mathrm{vol}(V_B)/\mathrm{vol}(V)$, respectively. In this new scenario it is easy to see that $p_A = p_B = 1/\mathrm{vol}(V) = 1/\mathrm{vol}(V)$. Thus, we would like to start a RW at vertex v with probability $\deg(v)/\mathrm{vol}(V)$, $\forall v \in V$. Section 6 shows that in practice this approach can successfully mitigate estimation errors caused by disconnected components. Unfortunately, it is difficult to sample m mutually *independent* vertices with probabilities proportional to their degrees. In the case where G is connected, this is equivalent to jointly start m independent random walkers in steady state. In networks such as MySpace, Facebook, and Bittorrent it is unclear how one can (efficiently) sample vertices with probabilities proportional to their degrees.

We want an m-dimensional random walk that, in steady state, samples **edges uniformly** *at random* **but***, unlike MultipleRW, can benefit from starting its walkers at* **uniformly sampled vertices***.*

5. FRONTIER SAMPLING (FS)

In this section we present a new and promising approach to an m-dimensional random walk that benefits from starting its walkers at uniformly sampled vertices. *Frontier Sampling* (FS) performs m *dependent* random walks in the graph. We refer to m as the dimension of the FS random walk. Let c be the cost of randomly sampling a vertex. The FS algorithm, given in Algorithm 1 is a centrally coordinated sampling algorithm that maintains a list of m vertices representing m random walkers. This way FS is less likely to

---

Algorithm 1: Frontier Sampling (FS).

1: $n \leftarrow 0$ {n is the number of steps}
2: Initialize $L = (v_1, \ldots, v_m)$ with m randomly chosen vertices (uniformly)
3: **repeat**
4:    Select $u \in L$ with probability $\deg(u)/\sum_{\forall v \in L} \deg(v)$
5:    Select an outgoing edge of u, (u, v), uniformly at random
6:    Replace u by v in L and add (u, v) to sequence of sampled edges
7:    $n \leftarrow n + 1$
8: **until** $n \geq B - mc$

---

get stuck in loosely connected components than a single random walker. However, in Section 5.2 we see that the joint steady state distribution of FS is much closer to the uniform distribution than is the steady state distribution of m independent random walkers. Section 5.3 describes how the FS algorithm can be made fully distributed. In Section 6 we see that, if the initial set of random walk vertices is chosen uniformly at random, FS estimates are more accurate than both single and m independent random walkers.

Frontier Sampling: An m-dimensional Random Walk

FS shares many of the same statistical properties of a single random walker. The key insight behind Theorem 5.2 below is that the FS stochastic process is equivalent to the stochastic process of a single random walker over the m-th Cartesian power of G, $G^m = (V^m, E_m)$, where

$$V^m = \{(v_1, \ldots, v_m) \,|\, v_1 \in V \wedge \cdots \wedge v_m \in V\}$$

is the m-th Cartesian power of V and $\forall \mathbf{v}, \mathbf{u} \in V^m$, $(\mathbf{v}, \mathbf{u}) \in E_m$ if exists an index i such that $(v_i, u_i) \in E$ and $u_j = v_j$ for $j \neq i$.
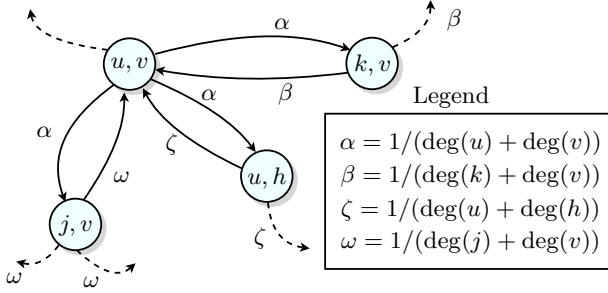
Figure 2: Illustration of the Markov chain associated to the Frontier sampler with dimension $m = 2$.

Lemma 5.1. *The Frontier sampling process is equivalent to the sampling process of a single random walker over G^m.*

PROOF. Consider the $(n-1)$-st step of FS. The reader may find Figure 2 helpful in following the proof. Let $L_n = (v_1, \ldots, v_m)$ be the state of FS before the n-th step. Clearly $L_n \in V^m$. Let $e(L_n)$ denote the collection of all edges associated to the vertices in L_n. We refer to $e(L_n)$ as the edge frontier at the n-th step. We describe the transition from state L_n to state L_{n+1} as follows (lines (4) and (5) of the FS algorithm): Select a vertex $v \in L_n$ with probability proportional to $\deg(v)$ and then replace vertex v in L_n with one of its neighbors (selected uniformly at random). This is equivalent to randomly sampling an edge from $e(L_n)$ with probability

$$p = \frac{1}{|e(L_n)|} = \frac{1}{\sum_{\forall v \in L_n} \deg(v)}.$$

Therefore, L_n transits to state L_{n+1} iff $(L_n, L_{n+1}) \in E_m$ and the transition probability from L_n to L_{n+1} is $1/|e(L_n)|$. Thus, the Markov chain that describes FS is equivalent to the Markov chain of a single random walker over G^m. □

Theorem 5.2. *Recall that G is a directed symmetric graph. If G is connected and non-bipartite, then in steady state FS has the following properties:*

(I) *edges are sampled uniformly at random and form a stationary sequence,*

(II) *has distribution, $L_\infty = (v_1, \ldots, v_m)$, equal to*

$$\frac{\sum_{i=1}^m \deg(v_i)}{m|V|^{m-1}\mathrm{vol}(V)} \, , \,$$

which is unique, and

(III) *the sequence of sampled edges satisfies the Strong Law of Large Numbers (Theorem 4.1).*

PROOF. The proof is found in our technical report [31]. □

In Section 4.5 we observed that, when starting multiple RWs, the MSE is reduced when the number of walkers inside each subgraph matches the number obtained when the graph is connected and all walkers are in steady state. In what follows we see that, in steady state, the average number of MultipleRW walkers in V_A is far from the average number obtained with m uniformly sampled vertices. In contrast, Section 5.2 shows that as $m \to \infty$, by uniformly sampling the starting vertices, FS starts in steady state with respect to the number of random walkers in any subset of vertices $V_A \subseteq V$.

5.1 MultipleRW Steady State v.s. Uniform Distribution

Consider a MultipleRW process with m walkers and let $K_{mw}(m)$ be a random variable that denotes the steady state number of MultipleRW random walkers in V_A. Let $\alpha_A = E[K_{mw}(m)]/E[K_{un}(m)]$ be the ratio between the steady state number of MultipleRW in V_A and the number of random walkers that start in V_A from uniformly sampled vertices. As all random walkers are independent we have

$$E[K_{mw}(m)] = \frac{m\,|V_A|d_A}{|V|d} \, .$$

It is also easy to see that

$$E[K_{un}(m)] = \frac{m\,|V_A|}{|V|}.$$

From the above we have

$$\alpha_A = E[K_{mw}(m)]/E[K_{un}(m)] = d_A/d.$$

Note that the value of α_A may be quite large or close to zero depending on both (1) the choice of V_A and (2) the average degree of G.

5.2 FS Steady State v.s. Uniform Distribution

Let $G = (V, E)$ be a connected graph and $V_A \subset V$ be a proper subset of V; define $V_B = V \backslash V_A$. Let $d_A = \mathrm{vol}(V_A)/|V_A|$, $d_B = \mathrm{vol}(V_B)/|V_B|$, and $d = \mathrm{vol}(V)/|V|$ be the average degrees of the vertices in V_A, V_B, and V, respectively. Consider a FS process with m walkers and let $K_{fs}(m)$ be a random variable that denotes the number of random walkers in V_A in steady state. Let $K_{un}(m)$ be a random variable that denotes the number of sampled vertices, out of m uniformly (randomly) sampled vertices from V, that belong to V_A. $K_{un}(m)$ has distribution $P[K_{un}(m) = k] = \binom{m}{k}p^k(1-p)^{m-k}$, $\forall k \geq 0$, where $p = |V_A|/|V|$. In this section we show that $K_{fs}(m)$ and $K_{un}(m)$ converge to the same limiting distribution, i.e.,

$$\lim_{m\to\infty} P[K_{fs}(m) = k] = \lim_{m\to\infty} P[K_{un}(m) = k], \; \forall k \geq 0. \quad (9)$$

Recall that the FS algorithm starts m random walkers at m uniformly sampled vertices (sampled independently). Let $V_A \subseteq V$. As m increases, eq. (9), the number of FS walkers that are initially selected to be in V_A approaches the steady state distribution (assuming G is connected).

Let $L \in V^m$ be the state of FS; from (Theorem 5.2) we have

$$P[L = (v_1, \ldots, v_m)] = \frac{\sum_{i=1}^m \deg(v_i)}{m|V|^{m-1}\mathrm{vol}(V)}.$$

Graph	Flickr	LiveJournal	YouTube	Internet RLT
Description	Social Net.	Social Net.	Social Net.	Internet tracert.
Type of graph	Directed	Directed	Directed	Directed
# of Vertices	$1,715,255$	$5,204,176$	$1,138,499$	$192,244$
Size of LCC	$1,624,992$	$5,189,809$	$1,134,890$	$609,066$
# of Edges	$22,613,981$	$77,402,652$	$9,890,764$	$609,066$
Average Degree	12.2	14.6	8.7	3.2
$w_{\max}$	2232	1029	3305	335
% of Original Graph	26.9%	95.4%	NA	NA

Table 1: Summary of the graph datasets used in our simulations. "Size of LCC" refers to the size of the largest connected component and $w_{\max}$ is the value of the largest vertex degree divided by the average degree.

In the following lemma we find the probability that $K_{fs}(m) = k$, $0 \le k \le m$.

Lemma 5.3.

$$P[K_{fs}(m) = k] = \frac{1}{md}\binom{m}{k}p^k(1-p)^{m-k}(k\,d_A + (m-k)d_B)\,,$$

where $p = |V_A|/|V|$ *and* $0 \le k \le m$.

PROOF. $P[K_{fs}(m) = k]$ is the sum of the probabilities $P[L = (v_1, \ldots, v_m)]$ over all states L in which exactly k vertices belong to V_A. Consider v_i, the i-th element of L. When $v_i \in V_A$, the contribution of v_i towards $P[K_{fs}(m) = k]$ is

$$|V_A|^{k-1}|V_B|^{m-k}\deg(v_i)/(m|V|^{m-1}\mathrm{vol}(V))\,;$$

when $v_i \in V_B$, the contribution of v_i towards $P[K_{fs}(m) = k]$ is

$$|V_A|^k|V_B|^{m-k-1}\deg(v_i)/(m|V|^{m-1}\mathrm{vol}(V))\,.$$

Summing over all elements in L and over all vertices yields

$$
\begin{aligned}
P[K_{fs}(m) = k] &= \binom{m}{k}\frac{|V_A|^k|V_B|^{m-k}}{m|V|^{m-1}\mathrm{vol}(V)} \\
&\times \left(\sum_{i=1}^{k}\sum_{\forall u_i \in V_A}\frac{\deg(u_i)}{|V_A|} + \sum_{j=1}^{m-k}\sum_{\forall t_j \in V_B}\frac{\deg(t_j)}{|V_B|}\right) \quad (10) \\
&= \frac{1}{m\,d}\binom{m}{k}p^k(1-p)^{m-k}(k\,d_A + (m-k)d_B)
\end{aligned}
$$

$\square$

The previous lemma gives the probability that a subset of vertices V_A has $K_{fs}(m) \in \{0, \ldots, m\}$ FS random walkers. The following theorem shows that $K_{fs}(m)$ and $K_{un}(m)$ converge to the same limiting distribution.

Theorem 5.4.
$\lim_{m \to \infty} P[K_{fs}(m) = k] = \lim_{m \to \infty} P[K_{un}(m) = k]$, $\forall k \ge 0$.

PROOF. From Lemma 5.3

$$P[K_{fs}(m) = k] = \frac{k\,d_A + (m-k)d_B}{m\,d}P[K_{un}(m) = k].\quad (11)$$

Note that if $k = mp$ we have

$$\frac{mpd_A + (m-mp)d_B}{m\,d} = \frac{m\frac{\mathrm{vol}(V_A)}{|V|} + m\frac{\mathrm{vol}(V_B)}{|V|}}{m\,d} = 1\,.\quad (12)$$

As $m \to \infty$, the probability mass of $P[K_{un}(m) = k]$ gets highly concentrated around the interval $k \in [mp - c\sqrt{m}, mp +$

$c\sqrt{m}]$, for large enough values of c. Let $k^-(m) = mp - z(m)\sqrt{mp(1-p)}$ and $k^+(m) = mp + z(m)\sqrt{mp(1-p)}$, where $z(m) = o(\sqrt[6]{m})$[1] is a slow increasing function of m. Note that

$$\frac{z(m)\sqrt{m}\sqrt{p(1-p)}(d_A - d_B)}{m\,d} = o(m^{4/6})/m$$

and, thus, eq. (12) yields

$$\lim_{m \to \infty}\frac{k^-(m)\,d_A + (m-k^-(m))d_B}{m\,d} = 1\quad (13)$$

and

$$\lim_{m \to \infty}\frac{k^+(m)\,d_A + (m-k^+(m))d_B}{m\,d} = 1.\quad (14)$$

All that is left to show is that $\lim_{m \to \infty} P[K_{un}(m) < k^-(m)] = 0$ and $\lim_{m \to \infty} P[K_{un}(m) > k^+] = 0$. Using an extension of the *de Moivre-Laplace* limit theorem shown in [12, pg. 193] yields

$$
\begin{aligned}
\lim_{m \to \infty} P[K_{un}(m) < k^-(m)] &= 0\,,\ \text{and} \\
\lim_{m \to \infty} P[K_{un}(m) > k^+(m)] &= 0.
\end{aligned}\quad (15)
$$

Putting together eqs. (11), (13), (14), and (15), with

$$\lim_{m \to \infty}\frac{k(m)\,d_A + (m-k(m))d_B}{m\,d} < \infty\,,\ k(m) = o(m)\,,$$

yields

$$\lim_{m \to \infty} P[K_{fs}(m) = k] = \lim_{m \to \infty} P[K_{un}(m) = k],\ \forall k \ge 0,$$

which concludes our proof. $\square$

We have seen as m gets larger, FS gets closer to starting in steady state with respect to the number of FS random walkers inside V_A, $\forall V_A \subset V$.

We have seen that if we initialize m random walkers with uniformly sampled vertices, FS starts closer to steady state than MultipleRW. In what follows we show that FS is well suited to be used in large scale (parallel, asynchronous) experiments without incurring in any coordination or communication costs between the random walkers.

5.3 Distributed FS

FS is well suited to be used in large scale (parallel, asynchronous) experiments. Let B be the budget of FS. In the distributed version of FS the budget is not directly related to the number of sampled vertices obtained by the algorithm.

---

[1] $f(m) = o(h(m))$ implies $\lim_{m \to \infty} f(m)/h(m) = 0$.

This is because distributed FS is achieved using multiple independent random walkers where the cost of sampling a vertex v is an exponentially distributed random variable with parameter $\deg(v)$. In what follows we show, using the Uniformization principle of Markov chains [8, Chapter 7.5] and the Poisson decomposition property, that FS can be made fully distributed.

Theorem 5.5. *A MultipleRW sampling process where the cost of sampling a vertex v is an exponentially distributed random variable with parameter $\deg(v)$ is equivalent to a FS process.*

PROOF. Consider the following Distributed FS (DFS) process. Let $\chi = \{L(\tau) \in V^m : \tau \in \mathbb{R}^\star\}$ be the Markov chain associated with a random walker over $G^m = (V^m, E_m)$, the m-th Cartesian power of G, with transition rate matrix

$$\mathbf{Q} = \mathbf{A} - \mathbf{D},$$

where $\mathbf{A}$ is the adjacency matrix of G^m, $\mathbf{A}_{i,j} \in \{0, 1\}$, $\forall i, j$, and $\mathbf{D}$ is a diagonal matrix with $\mathbf{D}_{i,i} = \sum_{\forall j} \mathbf{A}_{i,j}$. We observe this FS process over the interval $[0, B]$.

In the DFS process, the probability that the k-th random walker transitions out of vertex v_k at step $\tau + \Delta$ depends only on $\deg(v_k)$ and not on the state of $L(\tau)$. Thus, we can decompose the Poisson process describing a departure from the state $L'_n = (v_1, \ldots, v_m)$ into m independent stochastic processes, where the i-th process is a Poisson process with parameter $\lambda_i = \deg(v_i), i = 1, \ldots, m$. The above is equivalent to the stochastic process of a MultipleRW process with m random walkers and budget B, where the cost of sampling a vertex v is an exponentially distributed random variable with rate $\deg(v)$.

The DFS is equivalent to a FS process via the *Uniformization* property of Markov chains [8, Chapter 7.5]. The transition probability matrix of the Uniformized Markov chain (with unitary uniformization parameter) at the embedded transition points is

$$\mathbf{P} = I - \mathbf{D}^{-1}\mathbf{Q} = \mathbf{D}^{-1}\mathbf{A},$$

which is also the transition probability matrix of a FS process. $\square$

6. RESULTS

In this section we compare FS with SingleRW and MultipleRW. We also contrast FS with random vertex and edge sampling. The experiments consist of executing these sampling methods on a variety of real world graphs. The datasets used in the simulations are summarized in Table 1: "Flickr", "Livejournal", and "YouTube" are popular photosharing, blog (weblog), and video sharing websites, respectively. Users are represented as vertices of a graph. In these websites a user can subscribe to other user updates; an edge (u, v) exists between users u and v if user u subscribes to user v. At "Livejournal" and "YouTube" it is possible to query the incoming and outgoing edges of a given user. Further details of these three datasets can be found in [26]. "Internet RLT" is a router-level Internet graph collected from traceroute measurements of 23 monitors distributed over the world [13]. Note that some of these graphs contain disconnected components (subgraphs).

In the following simulations the starting vertex of each random walker is chosen uniformly at random from the set of all vertices. Our results show that FS estimates are consistently more accurate than their SingleRW and MultipleRW counterparts. Moreover, when restricted to the largest connected component, FS reaches steady state faster than SingleRW and MultipleRW in the simulations presented in our technical report [31].

6.1 Assortative Mixing Coefficient

In our first experiment we treat the graphs in Table 1 as undirected graphs. In-degrees and out-degrees are represented as vertex labels and the assortative mixing coefficient is obtained using the estimator described in Section 4.2.2.

In our experiment we average the estimates and calculate their mean squared error (MSE) over 100 runs. The sampling budget is $|V|/100$ for all graphs. Let $\hat{r}$ denote the estimated value of r. Table 2 shows a summary of the relative bias of $\hat{r}$ $(1 - E[\hat{r}]/r)$ and $\hat{r}$'s NMSE with respect to the true value of r. We observe that FS is consistently more accurate than both MultipleRW and SingleRW. If we focus on Flickr, the FS bias is 7 fold smaller than the bias of both MultipleRW and SingleRW. In addition FS's NMSE is one order of magnitude smaller than the NMSEs of MultipleRW and SingleRW. The Internet graph ("Internet RLT") is the only graph we studied that shows little difference between FS and MultipleRW.

We also perform an extreme experiment that focuses on the impact of loosely connected components on the assortative mixing estimates. Consider a graph that consists of two instances of a random undirected Barabási-Albert [5] graph, G_A and G_B, with 5×10^5 vertices each and average degrees 2 and 10, respectively, joined by a single edge connecting the two smallest degree vertices in G_A and G_B (ties are resolved arbitrarily). Henceforth, we use G_{AB} to denote the above graph. It is worth noting that over the G_{AB} graph, SingleRW consistently finds $\hat{r} = 0$ over all 100 runs. This is because SingleRW only estimated the assortative mixing of either subgraph A or subgraph B, which are both zero. Over G_{AB} MultipleRW performs almost as bad as SingleRW while FS is able to accurately estimate r.

6.2 In- and Out-degree Distribution Estimates

We now focus on estimating the in-degree distribution. Let $\theta = \{\theta_i\}_{\forall i \in \mathcal{L}}$ denote the in-degree distribution, where θ_i is the fraction of vertices with in-degree i. In our simulations we estimate $\gamma_i = \sum_{k=i+1}^{\infty} \theta_k$, the CCDF of θ, using equation (7). We choose to estimate the CCDF instead of the density because the CCDF is the plot of choice when it comes to displaying degree distributions. Each simulation consists of 10,000 runs (sample paths) used to compute the empirical CNMSE (equation (2)). The CNMSE is used to compare the accuracy of the estimates obtained from FS (dimension $m \in \{10, 1000\}$), SingleRW, and MultipleRW ($m \in \{10, 1000\}$ walkers). For the sake of conciseness, we restrict our presentation to a handful of representative results.

Consider first two representative results from the Flickr graph, whose in-degree CCDF (complementary cumulative distribution function) log-log plot is shown in Figure 3. The sampling budget is $B = 17,152 = |V|/100$, which amounts to sampling 1% of the vertices. In the first simulation, we are restricted to the *Largest Connected Component* (LCC) (which contains 94% of the vertices). The objective is to test if FS can outperform SingleRW and MultipleRW even

Graph	r	FS		MultipleRW		SingleRW	
		Bias	|NMSE|	Bias	|NMSE|	Bias	|NMSE|
Flickr	0.007	8%	1.08	752%	7.65	-619%	27.32
LiveJournal	0.07	-0.5%	0.11	-12%	0.16	1%	0.17
Internet RLT	0.17	3%	0.33	2%	0.32	17%	0.44
Youtube	-0.03	0.001%	0.02	2%	0.03	-1%	0.1
G_{AB}	0.08	0.01%	0.12	70%	0.72	100%	1.00

Table 2: Assortative mixing coefficient estimate bias and the module of the estimator NMSE ; r is the true value of the global clustering coefficient and these values are estimated value over 100 runs. The sampling budget is $|V|/100$ for all graphs.
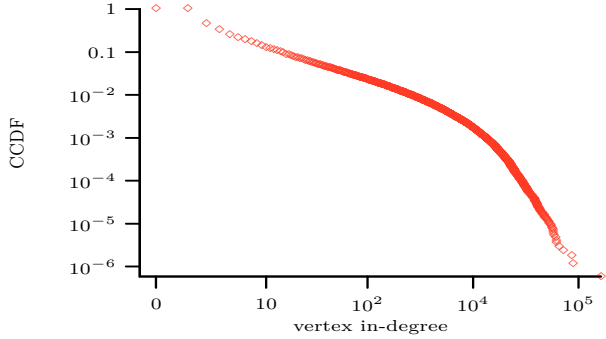
Figure 3: (Flickr) Log-log plot of the in-degree CCDF.
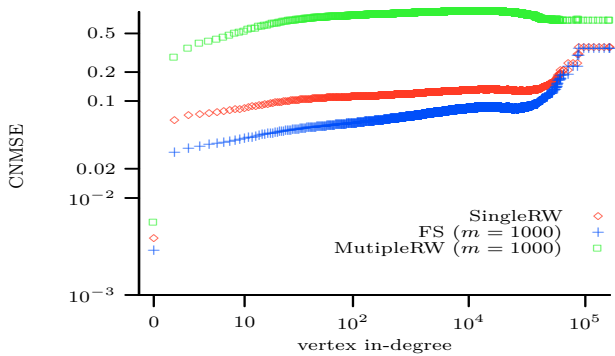
Figure 4: (LCC of Flickr) The log-log plot of the CNMSE of the in-degree distribution estimates with budget $B = |V|/100$.
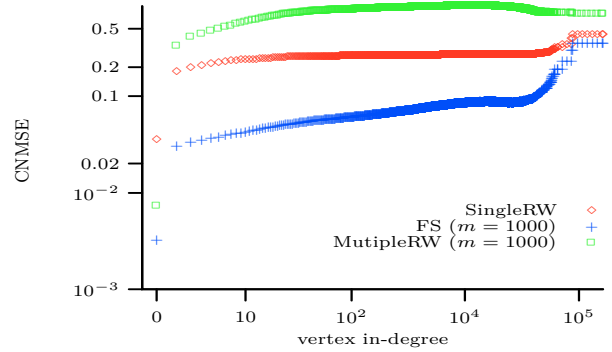

Figure 5: (Flickr) The log-log plot of the CNMSE of the in-degree distribution estimates with budget $B = |V|/100$.
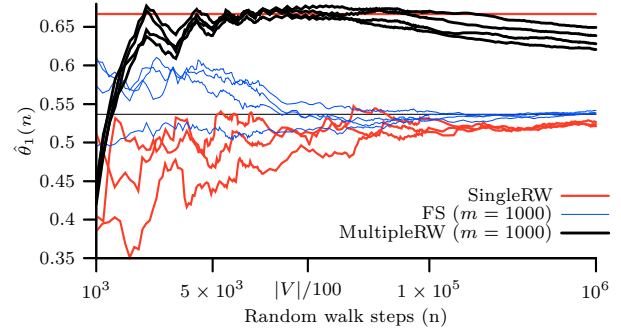

Figure 6: (LCC of Flickr) Four sample paths of $\hat{\theta}_1$ ($\theta_1 = 0.53$) as a function of the number of steps n (horizontal axis in log scale).

when there are no disconnected components. Figure 4 shows a log-log plot of the CNMSE of FS ($m = 1000$), SingleRW, and MultipleRW ($m = 1000$). In this experiment FS outperforms both SingleRW and MultipleRW. It is interesting to note that the estimates obtained by SingleRW are more accurate than the estimates obtained by MultipleRW. Now consider the complete Flickr graph. Figure 5 shows a log-log plot of the CNMSE of the in-degree distribution estimates. Contrasting the plots shown in Figures 4 and 5 we see that the gap between FS and both SingleRW and MultipleRW has significantly increased, favoring FS.

To better understand the differences between these sampling methods, Figure 6 focuses on four runs (sample paths) of the simulation over the complete Flickr graph. Figure 6 plots the evolution of $\hat{\theta}_1$ (the estimate of θ_1) as a function of n (the number of steps in the random walk). At each run of the simulator both FS and MultipleRW start at the same initial set of vertices (chosen using random vertex sampling). Figure 6 shows that all four FS sample paths (runs) quickly converge to the value of θ_1. For SingleRW, three of the four runs start inside the LCC. These runs do not converge to the

value of θ_1 as some vertices with in-degree one lie outside the LCC. In one of the runs, SingleRW starts in a small disconnected component and, thus, grossly overestimates the value of θ_1. For a similar reason, i.e., walkers starting at small disconnected components, MultipleRW grossly overestimates the value of θ_1. The MultipleRW rapid increase of $\hat{\theta}_1$ at around $n = 10^3$ steps needs further investigation. It may be due to the transient of the random walk (discussed in Section 4.4). Even when $n \gg 1$ (not shown in Figure 6) the MultipleRW estimate is unable to converge to θ_1. Modifying both SingleRW and MultipleRW methods to cope with disconnected components is an interesting open problem.

For the sake of conciseness, we omit the results of the simulations over the remaining graphs (Table 1) as they are similar to the results observed over the Flickr graph. However, consider the *out-degree* distribution estimates of Livejournal. Figure 7 shows a log-log plot of the CCDF of the out-degrees. The log-log plot of the CNMSE is shown in Figure 8 for FS ($m = 100$), SingleRW, and MultipleRW ($m = 100$) with sampling budget $B = |V|/10$. From Figure 8 we see that estimates of vertices with small out-degrees
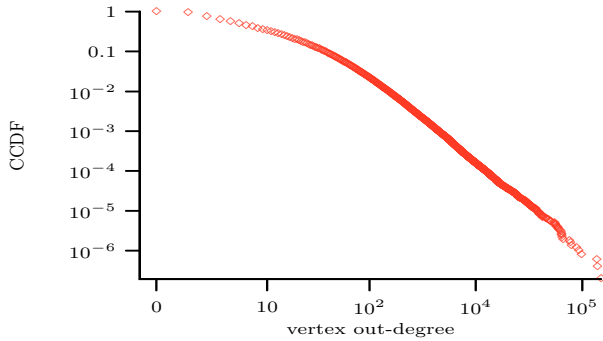
Figure 7: (Livejournal) Log-log plot of the out-degree CCDF.

in FS are up to one order of magnitude more accurate than those obtained from both SingleRW and MultipleRW.
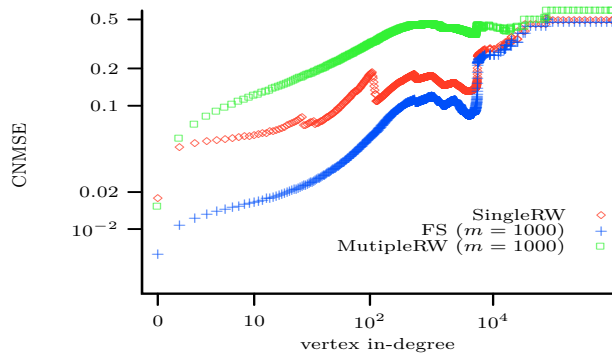

Figure 8: (Livejournal) The log-log plot of the CNMSE of the out-degree distribution estimation with sampling budget $B = |V|/100$ (CNMSE over $10,000$ runs).
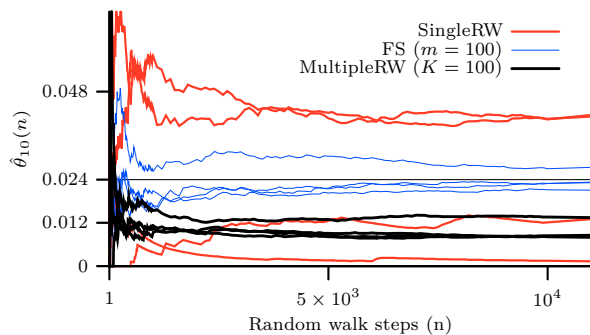

Figure 9: (G_{AB} graph) Four paths of $\hat{\theta}_{10}$ as a function of the number of steps n ($\theta_{10} = 0.024$).

The next experiment focuses on studying the impact of loosely connected components on the degree distribution estimates. For this we use the two Barabási-Albert joined graphs G_{AB} presented in Section 4.2.2. The experiment consists of estimating the degree distribution of G_{AB} using FS ($m = 100$), SingleRW, and MultipleRW ($m = 100$). Again, both FS and MultipleRW start at the initial set of vertices in each simulation (chosen uniformly at random). In this experiment the hypothesis is that, for small sampling budgets, each random walker will see the degree distribution of either G_A or G_B but not the degree distribution of G_{AB}.

Moreover, as the starting vertex of each random walker is chosen uniformly at random, G_A, which has the same number of vertices as G_B but $1/5$ of the edges, receives more random walkers than its *per edge* "share". Consequently, MultipleRW oversamples G_A.

Figure 9 shows the results of four simulation runs and plots the evolution of the estimates of θ_{10} ($\hat{\theta}_{10}$) as a function of the number of steps. In this simulation note that: (1) FS quickly converges to a value that is close to the correct value; (2) two out of the four SingleRW runs overestimate θ_{10} and the remaining two underestimate it; (3) three out of the four MultipleRW runs converge to the same, incorrect, fraction (underestimating the true value of θ_{10}). FS is designed to be robust to disconnected or loosely connected components. All of the FS runs quickly converge to a good estimates of θ_{10}. Figure 10 also shows that the CNMSE for FS is consistently lower than the CNMSE for SingleRW and MultipleRW.
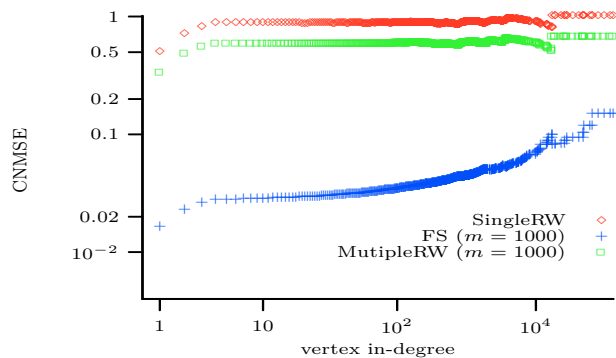

Figure 10: (G_{AB} graph) The log-log plot of the CNMSE of the degree distribution estimation with sampling budget $B = |V|/100$ (CNMSE over $10,000$ runs).

6.3 FS v.s. Stationary MultipleRW & SingleRW

We now compare FS with SingleRW and MultipleRW, when the latter two start in steady state. Figure 11 shows the results (over the Flickr graph) of the same simulation scenario used to obtain the results in Figure 5, except that now MultipleRW and SingleRW both start in steady state. While SingleRW has improved slightly (most notably at the tail errors), the benefit of starting in steady state is most felt by the MultipleRW method. In this simulation we see that the large estimation errors of MultipleRW in the previous simulations were due to the starting vertices being sampled uniformly at random. It is interesting to observe that MultipleRW starting in steady state and FS have similar estimation errors.

6.4 FS v.s. Random Independent Sampling

In Section 3 we showed that, if the degrees of two neighboring vertices are independent, random edge sampling is more accurate than random vertex sampling when it comes to estimating the tail of the degree distribution. In this section we observe this to be true over large real world graphs; we also observe that the accuracy of FS closely matches the accuracy of random edge sampling. In the following simulations we estimate the in-degree distribution. Random edge sampling uses the estimator $\hat{\theta}_i$, equation (7) (the estimator used for sampled vertices is trivial).
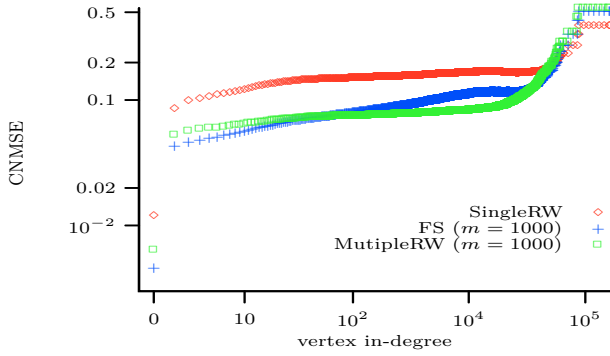
Figure 11: (Flickr) The log-log plot of the CNMSE of the in-degree distribution estimation of MultipleRW and SingleRW starting in steady state; sampling budget $B = |V|/100$ (NMSE over 10,000 runs).

In our first simulation we set the sampling cost of random vertex sampling to one and random edge sampling has cost two (as each edge samples two vertices). The sampling budget is $B = |V|/100$. We label this simulation "100% hit ratio" to indicate the unitary cost of randomly sampling vertices. Figure 12 shows a log-log plot of the NMSE (not the CNMSE) of our simulation over the (complete) Flickr graph. Here we use the NMSE (instead of the CNMSE) in order to be able to compare our results with the ones presented in equations (3) and (4). The vertical line indicates the average in-degree. Note that random edge sampling is more (less) accurate than random vertex sampling at estimating in-degrees larger (smaller) than the average in-degree, as predicted by equations (3) and (4) of our model in Section 3. We also observe that the accuracy of FS ($m = 1000$) closely matches the accuracy of random edge sampling.
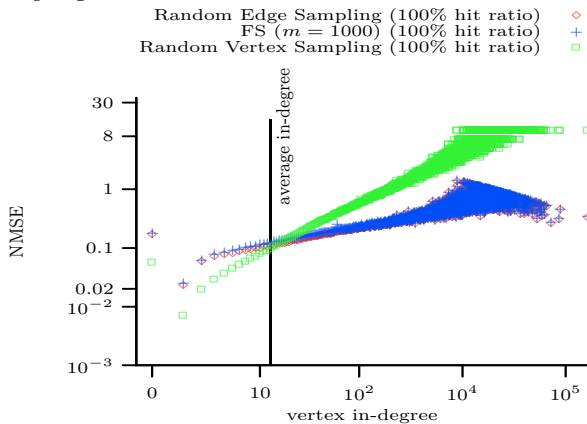

Figure 12: (Flickr) The log-log plot shows the NMSE of the in-degree distribution estimation with budget $B = |V|/100 = 18612$ (CNMSE over 10,000 runs).

Some complex networks exhibit a sparse user-id space. In this scenario a fraction of the sampling budget B can be spent querying invalid users-ids. Motivated by recent experiments over the MySpace network [30], the following experiment assumes that only 10% of the user-ids are valid, i.e., on average only one in every ten randomly sampled vertices are valid. We denote this value (10%) to be the *hit*
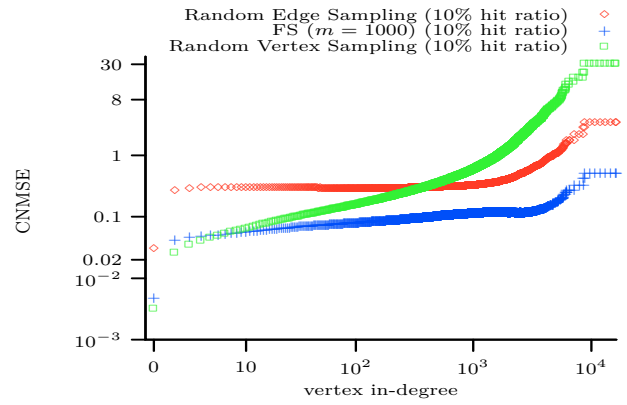

Figure 13: (Livejournal) The log-log plot shows the CNMSE of the in-degree distribution estimation with budget $B = |V|/100 = 52844$ (CNMSE over 10,000 runs).

ratio. For random edge sampling we assume a *hit ratio* of 1% (the choice of 1% is arbitrary). Figure 13 shows a log-log plot of the CNMSE of our simulation over the (complete) Livejournal graph with sampling budget $B = |V|/100 = 52844$. We observe that FS ($m = 1000$), which samples $m = 1000$ random vertices and (on average) crawls $B - 10\,m$ vertices, outperforms random edge sampling. Also note that FS estimates are more accurate than the estimates obtained from random vertex sampling for all but the three smallest in-degrees. This indicates that FS is more robust to low hit ratios than random vertex and edges sampling.
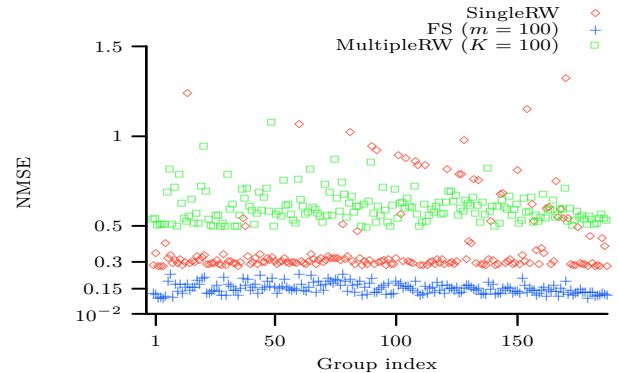

Figure 14: (Flickr) The NMSE of the density estimates of the most popular groups in the Flickr graph.

6.5 Density of Special Interest Groups

In a variety of complex networks, e.g. on-line social networks, each vertex (user) is associated with multiple labels that represent group affiliations, e.g. user interests, user geolocation, among others. For example, in the Flickr graph 21% of the users belong to one or more special interest groups [26]. Let $\mathcal{L}$ denote the set of groups in the Flickr graph and θ_l denote the fraction of vertices that belong to group $l \in \mathcal{L}$. In the simulations we estimate θ_l using FS ($m = 100$), SingleRW, and MultipleRW ($m = 100$) with budget $B = |V|/100$. Figure 14 shows the NMSE (from 10,000 runs) of the most popular 200 groups ordered in decreasing popularity. FS is clearly superior to both SingleRW and MultipleRW. Even when restricting the random walks to the largest connected component, FS still noticeably outperforms MultipleRW ($m = 100$) and SingleRW.

Graph	B	C	$E[\hat{C}]$ (NMSE)		
			FS	SingleRW	MultipleRW
Flickr	1%	0.14	0.13 (0.04)	0.12 (0.33)	0.16 (0.18)
LiveJournal	1%	0.16	0.16 (0.02)	0.16 (0.02)	0.17 (0.06)

Table 3: Global clustering coefficient estimates. C is the true value of the global clustering coefficient and $\hat{C}$ is its estimated value.

6.6 Global Clustering Coefficient Estimates

In our last set of experiments we evaluate the accuracy of estimating the global clustering coefficient using FS, SingleRW, and MultipleRW. Our simulations show a small difference between FS ($m = 1000$), SingleRW, MultipleRW ($m = 1000$). Let C be the clustering coefficient and $\hat{C}$ denote its estimated value. Table 3 presents the empirical value of $E[\hat{C}]$ and the empirical NMSE of the clustering coefficient, given by

$$\text{NMSE} = \frac{\sqrt{E[(\hat{C} - C)^2]}}{C},$$

over $10,000$ runs of FS, SingleRW, and MultipleRW over the Flickr and Livejournal graphs. From the results of Table 3 we see that FS accurately estimates the global clustering coefficient and has smaller error than both SingleRW and MultipleRW.

7. RELATED WORK

This section is devoted to review the related literature. FS can be classified as a Markov Chain Monte Carlo (MCMC) method. Other MCMC-based methods have been applied to characterize complex networks. Applications include, but are not limited to estimating characteristics of a population [36] (e.g. estimation of HIV seroprevalence among drug users [24]), content density in peer-to-peer networks [16, 23, 29, 35], uniformly sampling Web pages from the Internet [17, 33], and uniformly sampling Web pages from a search engine's index [4]. The above literature is mostly concerned with random walks that seek to sample vertices uniformly (also known as Metropolized Random Walks or Metropolis-RW) [16, 17, 33, 4, 35]. The accuracy of RW and Metropolis-RW (MRW) is compared in [15, 29], and in a variety of experiments RW estimates are shown to be consistently more accurate than or equal to MRW estimates.

The above literature does not consider the use of multiple random walks to address the problem of estimating characteristics of disconnected or loosely connected graphs. While multiple independent random walkers have been used as a convergence test in the literature, our simulations in Section 6 show that independent walkers are not suited to sample loosely connected graphs when the starting vertices are selected uniformly at random.

A number of real complex networks are known to have disconnected or loosely connected components. A large body of MCMC literature is dedicated to overcome the locality problem described in Section 4.3. However, the literature either assumes that the graph is very structured, e.g., a 2 dimensional lattice, or that the graph is completely known. These assumptions make the solutions inapplicable to our problem. A comprehensive list of MCMC methods and their characteristics can be found in [32].

Projecting a RW onto a higher dimensional space has been

used in [9] to make the Markov chain associated to the random walker nonreversible, which can speed up the mixing of the original RW. Unfortunately, it is unclear if this method can be successfully used to estimate characteristics of complex networks.

In networks that cannot be crawled (e.g., the Internet topology), samples must be obtained along shortest paths, and vertex degrees cannot be queried, [1] shows that observed vertex degrees are biased. Our work, however, assumes a graph can be crawled and vertex degrees queried. Our scenario admits a RW with an unbiased estimator. Multiple random walks also find other applications besides the one presented in this work. They are used to collect Web data [10], search P2P networks [6, 38], and decrease the time to discover "new wireless nodes" [2]. Dependent multiple random walks are also used in percolation theory [3].

8. DISCUSSION AND FUTURE WORK

In this work we presented a new and promising random walk-based method (*Frontier sampling*) that mitigates the estimation errors caused by subgraphs that "trap" a random walker. Frontier sampling (FS) uses multiple (m) mutually *dependent* random walkers starting from vertices sampled uniformly at random. The FS samples are shown to be the projection (onto the original graph) of a special type of m-dimensional (single) random walker. Simulations over real world graphs in Section 6 show that Frontier sampling (FS) is more robust than single and multiple independent random walkers (starting out of steady state) to estimate in-degree distributions and the fraction of users that belong to a social group. We also present evidence, using an analytical argument (also substantiated by simulations), that random walks (in particular, FS) are better suited to estimate the tail (all degrees greater than the average) of degree distributions than random vertex sampling. FS can also be made fully distributed without incurring in any coordination or communication costs.

The ideas behind FS can have far reaching implications, from estimating characteristics of dynamic networks to the design of new MCMC-based approximation algorithms.

9. ACKNOWLEDGMENTS

10. REFERENCES

[1] Dimitris Achlioptas, Aaron Clauset, David Kempe, and Cristopher Moore. On the bias of traceroute sampling: Or, power-law degree distributions in regular graphs. *J. ACM*, 56(4):1–28, 2009.

[2] Chen Avin and Bhaskar Krishnamachari. The power of choice in random walks: An empirical study. *Comput. Netw.*, 52(1):44–60, 2008.

[3] P. Balister, B. Bollobás, and A. Stacey. Dependent percolation in two dimensions. *Prob. Theory and Related Fields*, 117(4):495–513, 2000.

[4] Ziv Bar-Yossef and Maxim Gurevich. Random sampling from a search engine's index. *J. ACM*, 55(5):1–74, 2008.

[5] A.-L. Barabási and R. Albert. Emergence of scaling in random networks. *Science*, 286:509–512, 1999.

[6] Nabhendra Bisnik and Alhussein A. Abouzeid. Optimizing random walk search algorithms in p2p networks. *Computer Networks*, 51(6):1499–1514, 2007.

[7] S. Boccaletti, V. Latora, Y. Moreno, M. Chavez, and D.-U. Hwang. Complex networks: Structure and dynamics. *Physics Reports*, 424(4-5):175–308, 2006.

[8] Christos G. Cassandras and Stephane Lafortune. *Introduction to Discrete Event Systems.* Springer-Verlag, Inc., 2006.

[9] Fang Chen, László Lovász, and Igor Pak. Lifting Markov chains to speed up mixing. In *Proc. of STOC*, pages 275–281, 1999.

[10] Junghoo Cho and Hector Garcia-Molina. Parallel crawlers. In *Proc. of the WWW*, pages 124–135, 2002.

[11] Nathan Eagle, Alex S. Pentland, and David Lazer. Inferring friendship network structure by using mobile phone data. *PNAS*, 106(36):15274–15278, August 2009.

[12] W. Feller. *An Introduction to Probability Theory and its Applications*, volume 1. John Wiley & Sons, New York, 3rd edition, 1968.

[13] Coperative Association for Internet Data Analysis. CAIDA's Internet topology data kit #0304, 2003.

[14] Charles J. Geyer. Practical Markov Chain Monte Carlo. *Statistical Science*, 7(4):473–483, 1992.

[15] Minas Gjoka, Maciej Kurant, Carter T. Butts, and Athina Markopoulou. A walk in Facebook: Uniform sampling of users in online social networks. In *Proc. of the IEEE Infocom*, March 2010.

[16] Christos Gkantsidis, Milena Mihail, and Amin Saberi. Random walks in peer-to-peer networks: algorithms and evaluation. *Perform. Eval.*, 63(3):241–263, March 2006.

[17] Monika R. Henzinger, Allan Heydon, Michael Mitzenmacher, and Marc Najork. On near-uniform url sampling. In *Proceedings of the WWW*, pages 295–308, 2000.

[18] Marlom A. Konrath, Marinho P. Barcellos, and Rodrigo B. Mansilha. Attacking a swarm with a band of liars: evaluating the impact of attacks on bittorrent. In *P2P '07: Proceedings of the Seventh IEEE International Conference on Peer-to-Peer Computing*, pages 37–44, Washington, DC, USA, 2007. IEEE Computer Society.

[19] Jure Leskovec and Christos Faloutsos. Sampling from large graphs. In *Proc. of the KDD*, pages 631–636, 2006.

[20] Jure Leskovec, Kevin J. Lang, Anirban Dasgupta, and Michael W. Mahoney. Statistical properties of community structure in large social and information networks. In *Proc. of the WWW*, pages 695–704, 2008.

[21] David A. Levin, Yuval Peres, and Elizabeth L. Wilmer. *Markov Chains and Mixing Times.* AMS, 2009.

[22] L. Lovász. Random walks on graphs: a survey. *Combinatorics*, 2:1–46, 1993.

[23] Laurent Massoulié, Erwan Le Merrer, Anne-Marie Kermarrec, and Ayalvadi Ganesh. Peer counting and sampling in overlay networks: random walk methods. In *Proc. of the PODC*, pages 123–132, 2006.

[24] Courtney McKnight, Don Des Jarlais, Heidi Bramson, Lisa Tower, Abu S. Abdul-Quader, Chris Nemeth, and Douglas Heckathorn. Respondent-driven sampling in a study of drug users in New York City: Notes from the field. *Journal of Urban Health*, 83(6):154–159, 2006.

[25] Sean Meyn and Richard L. Tweedie. *Markov Chains and Stochastic Stability.* Cambridge University Press, 2009.

[26] Alan Mislove, Massimiliano Marcon, Krishna P. Gummadi, Peter Druschel, and Bobby Bhattacharjee. Measurement and Analysis of Online Social Networks. In *Proc. of the IMC*, October 2007.

[27] M. E. J. Newman. Assortative mixing in networks. *Phys. Rev. Lett.*, 89(20):208701, Oct 2002.

[28] M. E. J. Newman. The structure and function of complex networks. *SIAM Review*, 45(2):167–256, 2003.

[29] Amir H. Rasti, Mojtaba Torkjazi, Reza Rejaie, Nick Duffield, Walter Willinger, and Daniel Stutzbach. Respondent-driven sampling for characterizing unstructured overlays. In *Proc. of the IEEE Infocom*, pages 2701–2705, April 2009.

[30] Bruno Ribeiro, William Gauvin, Benyuan Liu, and Don Towsley. On MySpace account spans and double Pareto-like distribution of friends. In *IEEE Infocom 2010 Network Science Workshop*, Mar 2010.

[31] Bruno Ribeiro and Donald Towsley. Estimating and sampling graphs with multidimensional random walks. arXiv:1002.1751v2, 2010.

[32] Christian P. Robert and George Casella. *Monte Carlo Statistical Methods.* Springer-Verlag, 2nd edition, 2005.

[33] Paat Rusmevichientong, David M. Pennock, Steve Lawrence, and Lee C. Giles. Methods for sampling pages uniformly from the world wide web. In *AAAI Fall Symposium on Using Uncertainty Within Computation*, pages 121–128, 2001.

[34] Thomas Schank and Dorothea Wagner. Approximating clustering-coefficient and transitivity. *Journal of Graph Algorithms and Applications*, 9(2):265–275, 2004.

[35] Daniel Stutzbach, Reza Rejaie, Nick Duffield, Subhabrata Sen, and Walter Willinger. On unbiased sampling for unstructured peer-to-peer networks. *IEEE/ACM Trans. Netw.*, 17(2):377–390, 2009.

[36] Erik Volz and Douglas D. Heckathorn. Probability based estimation theory for Respondent-Driven Sampling. *Journal of Official Statistics*, 2008.

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

[38] Ming Zhong and Kai Shen. Random walk based node sampling in self-organizing networks. *SIGOPS Oper. Syst. Rev.*, 40(3):49–55, 2006.