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.

Keywords

Multidimensional Random Walks, Graph sampling, Estimation, Degree distribution, Assortative mixing, Global clustering coefficient, Frontier Sampling

1. INTRODUCTION

A number of recent studies [7, 11, 14, 18, 19, 25, 29, 28, 34] (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, 27]. 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 [23], or the average number of copies of a file in a peer-to-peer (P2P) network [15].

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 [29]). 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 can be impractical when edges cannot be randomly queried (e.g., online social networks like Facebook [14], MySpace [29], 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 [25], and Bittorrent [17]). 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 [17].
- 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

IMC'10, November 1–3, 2010, Melbourne, Australia.

by moving a particle (walker) from a vertex to a neighboring vertex (through 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 over graphs such as Twitter, LiveJournal [25], YouTube [25], Facebook [14], MySpace [29], P2P networks [28], 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 [20]) 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 K independent random walkers, for any $K > 0$. 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 random vertex and random edge sampling. We show analytically that the tail of power law graphs is better estimated using random edge sampling than using random vertex sampling. Using simulations over real world networks (in Section 6.4) we observe that FS accuracy is comparable to the accuracy of random edge sampling. These results help explain recent empirical results [28]; (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 of our model. 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.

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 a symmetric directed graph on-the-fly 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 a connected graph. As G is symmetric, we denoted $\deg(v)$ to be the in-degree or the out-degree of $v \in V$ as they are equal.

The graph G is also labeled. Let $\mathcal{L}$ be a finite set of vertex or edge labels (whether $\mathcal{L}$ is a set of vertex or edge labels will be clear from the context). Each edge $(u, v) \in E$ is associated to a set of labels $\mathcal{L}(u, v) \subseteq \mathcal{L}$. For instance, the label of edge (u, v) can be the in-degree of v in the original graph G_d. Similarly, we can associate a set of labels to each vertex, $\mathcal{L}(v)$, $\forall v \in V$. Whether we are referring to edge or vertex labels will be clear from the context.

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

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

For the sake of simplicity, 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, unless stated otherwise.

3. VERTEX V.S. EDGE SAMPLING

Here we consider a simple estimation problem. We use this example 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. We assume that d is known and that a sampled edge (u, v) only provides the out-degree of u. In random edge sampling the probability of sampling a vertex with out-degree i is proportional to the out-degree: $\pi_i = i\,\theta_i/d$. On the other hand, random vertex sampling samples a vertex with out-degree i with probability θ_i. A simple calculation shows that the NMSE (equation (1)) of B randomly sampled *edges* with out-degree i is

$$\mathrm{NMSE}(i) = \sqrt{(1/\pi_i - 1)/B}\,, \quad i > 0. \tag{2}$$

Similarly, the NMSE(i) for random vertex sampling is

$$\mathrm{NMSE}(i) = \sqrt{(1/\theta_i - 1)/B}\,. \tag{3}$$

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 (2) and (3) 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. This characteristic of random edge sampling is also known as importance sampling estimation [30, Chapter 3.3].

Above we have seen that random edge sampling is more accurate than random vertex sampling in estimating the tail of the degree distribution. Our analytical conclusion agrees with real world experiments [28]. 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 and connected graph G. Sampling G with a RW is a simple task. 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 unitary 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. An edge (u, v) is sampled by a RW if both vertices u and v are visited by the walker. Note that we allow edges to be sampled multiple times. We refer to (u_i, v_i) as the i-th sampled edge. At the n-th step a walker at vertex v chooses an outgoing edge (v, u) uniformly at random from the outgoing edges of v and adds (v, u) to the sequence of sampled edges. At step $n + 1$ the random walker starts at vertex u and the sampling continues until $n = B$.

The RW described here is the most common type of RW found in the literature [21]. Other types of random walks differ in the way in which outgoing edges are sampled. The Metropolis-Hastings RW [14, 33, 36] is an example. However, experiments estimating a variety of metrics using the Metropolis-Hastings RW [14, 33, 36], which mimics random *vertex* sampling, indicate that the Metropolis-Hastings RW has lower accuracy than the random walk described in this work [14, 28]. For more details about other types of RW please refer to [30, 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 [21]). 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. Stationarity is a natural generalization of random sampling where the assumption of independence is dropped. Once the RW reaches steady state, it also shares two important properties with random edge sampling. Both sample edges uniformly at random [21] and both 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 (asymptoticaly) unbiased estimators. We provide a trivial extension of a well known result [24, Section 17.2] to the case where interested in a subset of the graph edges. Let $E^\star$ be a non-empty subset of E. Let (u_i, v_i) be the i-th RW sampled edge such that $(u_i, v_i) \in E^\star$; and let $B^\star$ be the number of such samples. $B^\star$ is the number of RW sampled edges that belong to $E^\star$. Note that $B^\star \leq B$, where B is the number of RW steps.

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

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

where "a.s." denotes "almost sure" converge, i.e., the event happens with probability one. Moreover, $\lim_{B\to\infty} B^\star = \infty$ for any $E^\star$.

Proof. The proof that $\lim_{B\to\infty} B^\star = \infty$ for any $E^\star$ is trivial as the RW is ergodic. The remainder of the proof borrows many elements from the proof in [24, Section 17.2]. As the RW is stationary, the edges are sampled uniformly at random and thus, from the linearity of expectation,

$$E\left[\frac{1}{B^\star}\sum_{i=1}^{B^\star} f(u_i, v_i)\right] = \frac{1}{B^\star}\sum_{i=1}^{B} E[f(u_i, v_i)] = \sum_{\forall (u,v)\in E^\star} \frac{f(u, v)}{|E^\star|}\,.$$

Now consider the $B^\star$-th sampled edge, $(u_{B^\star}, v_{B^\star})$. Let $l_{B^\star}$ be the number of times that $(u_{B^\star}, v_{B^\star})$ is visited by the RW and let $\tau_{uv}(k)$ be the time of the $(k+1)$-st visit to $(u_{B^\star}, v_{B^\star})$. Let

$$S_j(f) = \sum_{n=\tau_{uv}(j)+1}^{\tau_{uv}(j+1)} f(u_n, v_n).$$

By the strong Markov property the random variables $\{S_j$:

$j \geq 0\}$ are i.i.d. with the same average. As

$$\sum_{j=0}^{l_{B^\star}-1} S_j = \sum_{i=1}^{B^\star} f(u_i, v_i) - \sum_{k=1}^{\tau_{uv}(0)} f(u_k, v_k)$$

and because the chain is ergodic we have

$$\lim_{B^\star \to \infty} \sum_{k=1}^{\tau_{uv}(0)} f(u_k, v_k)/B^\star = 0\,.$$

The proof follows directly from the application of the Strong Law of Large Numbers for i.i.d. rv's ($\{S_j : j \geq 0\}$). □

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 ($B \to \infty$). 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.

Here we present estimators of four graph characteristics, namely: the edge label density (the fraction of edges with a given label in the graph), the assortative mixing coefficient [26], the vertex label density, and the global clustering coefficient [32]. 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 a 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 $l \in \mathcal{L}$ in G among all edges that have labels. 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 a non-empty subset of E. Let p_l denote the fraction of edges in $E^\star$ with label l; it is clear that

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

where

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

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

$$\hat{p}_l \equiv \sum_{i=1}^{B^\star} \frac{\mathbf{1}(l \in \mathcal{L}(u,v))}{B^\star}\,, \tag{4}$$

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

4.2.2 Assortative Mixing Coefficient

The assortative mixing coefficient [26] 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 (4) 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 [26]). It is trivial to extend our analysis to other types of assortative mixing coefficients, e.g., equations (21) and (23) of [26].

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 out-degree (in-degree) of G_d. The degree assortative mixing coefficient [26] 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

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

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

and $\hat{\sigma}_{\text{in}}$ ($\hat{\sigma}_{\text{out}}$) are the standard deviation of the distribution $\hat{q}_i^{\text{in}}$ ($\hat{q}_j^{\text{out}}$). As the estimate $\hat{p}_{ij}$ (equation (4)) 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, $\hat{r}$ also asymptotically converges, almost surely, to the true assortative mixing coefficient of [26], as long as $\sigma_{\text{in}} > 0$ and $\sigma_{\text{out}} > 0$. This also implies that $\hat{r}$ is an asymptotically unbiased estimator of the assortative mixing coefficient of G_d.

4.2.3 Vertex Label Density

Let $\mathcal{L}(v)$ be the set of labels associated with vertex v, $\forall v \in V$. We seek to calculate, θ_l, the fraction of *vertices* with label l in G. It is given as

$$\theta_l = \frac{1}{|V|} \sum_{\forall (u,v) \in E} \frac{\mathbf{1}(l \in \mathcal{L}(v))}{\deg(v)}\,, \tag{5}$$

as $G = (V, E)$ is directed and symmetric. It is trivial to verify that θ_l is the fraction of vertices with label l. Replacing E with a sequence of edges sampled by a stationary RW and renormalizing we arrive at the following estimator

$$\hat{\theta}_l \equiv \frac{1}{S\,B} \sum_{i=1}^{B} \frac{\mathbf{1}(l \in \mathcal{L}(u_i, v_i))}{\deg(v_i)}\,, \tag{6}$$

where $S = 1/B \sum_{i=1}^{B} 1/\deg(v_i)$. Note that here $E^\star = E$ and $B^\star = B$. From Theorem 4.1 we have $\lim_{B\to\infty} S \overset{\text{a.s.}}{\to} |V|/|E|$.

Using again Theorem 4.1 we have that

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

which divided by $|V|/|E|$ gives back equation (5). As S converges almost surely to $|V|/|E|$, we have that $\lim_{B\to\infty} \hat{\theta}_l \overset{\text{a.s.}}{\to} \theta_l$. 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 [35]. 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 [32]. Let $V^\star$ be the set of vertices $v \in V$ with $\deg(v) \geq 2$. The global clustering coefficient of a undirected graph is defined as [32]

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

where

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

$\Delta(v) = |\{(u,w) \in E : (v,u) \in E \text{ and } (v,w) \in E\}|$ is the number of triangles that contain vertex v and $\binom{\deg(v)}{2}$ is the number of triples of vertex v. We estimate C from a sequence of edges sampled by a stationary RW:

$$\hat{C} = \frac{1}{S\,B}\sum_{i=1}^{B} \frac{c(v_i)}{\deg(v_i)},$$

where

$$S = \frac{1}{B}\sum_{i=1}^{B} \frac{\mathbf{1}(\deg(v_i) \geq 2)}{\deg(v_i)}.$$

Note that from sampled edge (u_i, v_i) we only use vertex v_i; also note that $E^\star = E$ and $B^\star = B$.

Corollary 4.2. $\lim_{B\to\infty} \hat{C} \overset{a.s.}{\to} C$, *i.e., $\hat{C}$ converges almost surely to C.*

Proof. From Theorem 4.1

$$\lim_{B\to\infty} S \overset{\text{a.s.}}{\to} \frac{|V^\star|}{|E|}.$$

Also from Theorem 4.1

$$\lim_{B\to\infty} \frac{1}{B}\sum_{i=1}^{B} \frac{c(v_i)}{\deg(v_i)} \overset{\text{a.s.}}{\to} \frac{1}{|E|} \sum_{\forall (u,v)\in E} \frac{c(v)}{\deg(v)} = \frac{1}{|E|} \sum_{\forall v\in V} c(v),$$

which together with the almost sure convergence of S implies that $\lim_{B\to\infty} \hat{C} \overset{\text{a.s.}}{\to} C$. □

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. 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 [13]. For instance, if the random walker does not start in a stationary regime (transient), it is common practice to discard the first w samples [13]. 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 [14] to sample Facebook), is to sample the graph using multiple independent random walkers [13]. In what follows we see that this naive approach can lead to increased estimation errors. In Section 5 we see how to mitigate the random walk "trapping" problem with m *dependent* random walkers.

4.4 Multiple Independent Random Walkers

The problem with a single walker is that it may get traped inside *one* local neighborhood. But what if we could start m random walkers (*MultipleRW*) at different parts of the graph? If $m = 1$ we are back to sampling G using a single random walker, which we denote as *SingleRW*. Some complex networks admit random (uniform) vertex sampling at cost c (e.g., MySpace, Facebook, Bittorrent). In this complex networks random vertex sampling may help us start m random walkers in different parts of the graph. While the value of c can be high compared to the cost of RW sampling, initializing m random walkers costs (only) mc units of the budget.

In what follows we see that starting m random walkers from m randomly sampled vertices may not decrease the estimation error. To simplify our exposition we assume that if B is the sampling budget, each walker takes B/m steps. Let $(V_1, \ldots, V_m)$ be the state of m independent random walkers in steady state. It is easy to verify that for any value of m edges are sampled with probability $1/|E|$.

Consider the following thought experiment. Let G 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 $|E_A| \gg |E_B|$. We start two independent random walkers (MultipleRW with $m = 2$) from two randomly (uniformly) sampled vertices. Let p_A (p_B) be the probability that the random walker samples an edge (u, v) in G_A (G_B) after $B \gg 1$ steps. It is easy to see that $p_A \approx |V_A|/(|V_A| + |V_B|) \times 1/|E_A|$ and $p_B \approx |V_B|/(|V_A| + |V_B|) \times 1/|E_B|$. Thus, $p_A \ll p_B$, which means that the edges in G_B are oversampled while the edges in G_A are undersampled. As our estimators assume that all edges are sampled uniformly, this imbalance between p_A and p_B may introduce biases and higher MSEs. Note that increasing B can only mitigate the problem if G is connected.

Now consider the same thought experiment where a random walker starts in G_A (G_B) with probability $|E_A|/(|E_A| + |E_B|)$ ($|E_B|/(|E_A| + |E_B|)$). In this new scenario it is easy to

see that $p_A = p_B = 1/(|E_A| + |E_B|)$. Thus, starting vertices of MultipleRW in a component in proportion to the number of edges of the component is a key factor to determine the accuracy of MultipleRW. Section 6 shows that in practice this approach can successfully mitigate estimation errors caused by disconnected components. Unfortunately, starting m mutually *independent* random walkers in a component in proportion to the number of edges of the component is a hard task. In complex networks such as MySpace, Facebook, and Bittorrent it is unclear how this can be achieved.

Ideally we want to start the m random walkers from m uniformly sampled vertices. In a graph where uniform vertex sampling is close to the steady state of MultipleRW, we should still be able to mitigate estimation errors caused by loosely connected components. Unfortunately, as we see next, the steady state of MultipleRW can be far from uniform. In Section 6 we see that in practice starting MultipleRW uniformly may even increase the estimation errors when compared to SingleRW also starting from a uniformly chosen vertex.

MultipleRW Steady State v.s. Uniform Distribution

Here we see that the joint distribution of m randomly sampled vertices differs considerably from the steady state distribution of MultipleRW. Our metric of distance is the total variation distance. The total variation distance is commonly used in the literature of random walks to compare steady state distributions. The total variation distance between two probability distributions is the largest possible difference between the probabilities that the two distributions assign to the same event. The steady state distribution of MultipleRW with m walkers, $(v_1, \ldots, v_m) \in V^m$, is

$$\frac{\prod_{i=1}^{m} \deg(v_i)}{(|E|)^m},$$

as the steady state of the i-th random walker is $\deg(v_i)/|E|$ and there are m independent random walkers.

The total variation distance, δ_{mw}, between the MutipleRW steady state and the uniform distribution is

$$\delta_{mw} \equiv \sup_{(v_1,\ldots,v_m)\in V^m} \left| \frac{\prod_{i=1}^{m} \deg(v_i)}{|E|^m} - \frac{1}{|V|^m} \right|.$$

Note that the largest difference happens when all m walkers are either at the largest or at the lowest degree vertex. Let $W_{\min}$ and $W_{\max}$ denote the minimum and maximum degrees of a vertex in G, respectively; and let $w_{\min} = W_{\min}/d$ and $w_{\max} = W_{\max}/d$. Then,

$$\delta_{mw} = \frac{1}{|V|^m} \max\left(w_{\max}^m - 1, 1 - w_{\min}^m\right). \quad (7)$$

Also note that $w_{\max} \geq 1$, $w_{\min} \leq 1$, and $m \geq 1$. When $w_{\max}$ is large we have that the distance

$$\delta_{mw} \approx \left(\frac{w_{\max}}{|V|}\right)^m,$$

converges to zero as $m \gg 1$ but it is $w_{\max}^m$ times larger than the uniform vertex sampling probability. Thus, the steady state of MultipleRW can be $w_{\max}^m$ times larger than uniform.

In real world graphs $w_{\max}$ is likely to be fairly large and m fairly small (due to the cost of performing random vertex sampling). Table 1 shows the values of $w_{\max}$ of some real world graphs (these graphs are described in details in Section 6). Our Flickr graph has $w_{\max} = 2232$ and our Livejournal graph has $w_{\max} = 1029$. In Section 6 we see that, over real world graphs, when starting at randomly sampled vertices MultipleRW performs poorly even when compared to SingleRW. However, the above analysis motivates the following question:

> **Q:** *Is there a multidimensional random walk that, in steady state, samples* **edges** *uniformly at random* **but** *its joint distribution is close to random* **vertex** *sampling?*

5. FRONTIER SAMPLING

In this section we present a new and promising approach to address the above question. *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 is simple:

Frontier Sampling (FS) is a centrally coordinated sampling algorithm that maintains a list of m vertices representing m random walkers. This way FS is less likely to get stuck in loosely connected components than a single random walker. However, in Section 5.1 we see that unlike m independent random walkers, the joint steady state distribution of FS is much closer to the uniform distribution. Section 5.2 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.

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** $n \geq B - mc$

Algorithm 1: Frontier Sampling (FS).

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 $j \neq i$.

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 1 helpful in following the proof. Let $L_n = (v_1, \ldots, v_m)$ be the state of FS before the n-th step. Clearly

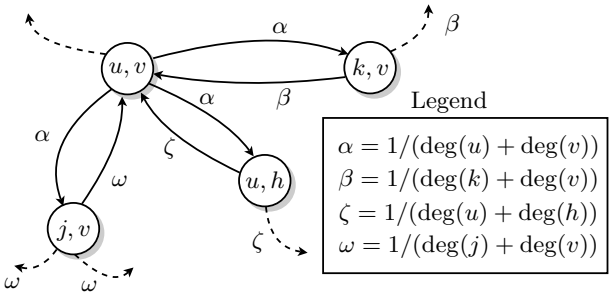

Figure 1: Illustration of the Markov chain associated to the Frontier sampler with dimension $m = 2$.

$L_n \in V^m$. 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 is able to transition 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. *If G is connected and non-bipartite, then FS is stationary and has a unique stable distribution where:*

(I) edges are sampled with probability $1/|E|$,

(II) sampled edges form a stationary sequence,

(III) its steady state distribution, $L_\infty = (v_1, \ldots, v_m)$, is

$$\frac{\sum_{i=1}^m \deg(v_i)}{m|V|^{m-1}|E|}, \text{ and}$$

(IV) the sequence satisfies the Strong Law of Large Numbers (Theorem 4.1).

Proof. The proof is found in Appendix A. □

In what follows we compare the joint steady state distribution of FS with the uniform distribution.

5.1 FS Steady State v.s. Uniform Distribution

In what follows we show that FS steady state distribution gets closer to uniform as m (the number of walkers) gets larger. Again we use the total variation distance as our metric to compare distributions. The steady state distribution of FS with m walkers, $(v_1, \ldots, v_m) \in V^m$, is (Theorem 5.2)

$$\frac{\sum_{i=1}^m \deg(v_i)}{m|V|^{m-1}|E|}.$$

It is important to note that FS and MultipleRW share the same state space, V^m.

The variation distance between the FS steady state and the uniform distribution is

$$\delta_{fs} \equiv \sup_{(v_1,\ldots,v_m)\in V^m} \left| \frac{\sum_{i=1}^m \deg(v_i)}{m|V|^{m-1}|E|} - \frac{1}{|V|^m} \right| = \\ = \frac{1}{m|V|^m} \sup_{(v_1,\ldots,v_m)\in V^m} \left| \sum_{i=1}^m \deg(v_i)/d - m \right|, \quad (8)$$

where $d = |E|/|V|$ is the average degree. It is easy to see that the largest difference happens when all m walkers are either at the largest or at the lowest degree vertex. Following Section 4.4, let $W_{\min}$ and $W_{\max}$ denote the minimum and maximum degrees of a vertex in G, respectively; and let $w_{\min} = W_{\min}/d$ and $w_{\max} = W_{\max}/d$. Now we simplify equation (8):

$$\delta_{fs} = \frac{1}{m|V|^m} \sup_{(v_1,\ldots,v_m)\in V^m} \left| \sum_{i=1}^m \deg(v_i)/d - m \right| = \\ = \frac{1}{|V|^m} \max\left(w_{\max} - 1, 1 - w_{\min}\right).$$

Compare the above equation with our MultipleRW δ_{mw} in equation (7). With $m = 1$ we are in the single random walker case and $\delta_{fs} = \delta_{\mathrm{mw}}$. With $m > 1$, the inequality is strict $\delta_{fs} < \delta_{\mathrm{mw}}$. Now consider $w_{\max} \geq 2$ (a reasonable assumption for most complex networks). In this scenario the distance

$$\delta_{fs} \approx \frac{w_{\max}}{|V|^m},$$

is $w_{\max}$ times larger than the uniform vertex sampling probability while for MultipleRW it is $w_{\max}^m$ (as seen in Section 4.4), which diverges as m gets larger. In the Flickr graph (detailed in Table 1) we observe that δ_{mw} is approximately 2232^{m-1} times larger than δ_{fs}. Also note that, different from the single random walker case, the absolute difference quickly goes to zero as m gets larger. This is because the probability that all random walkers are at the largest degree vertex goes to zero as m gets larger.

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.

5.2 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. 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 5.3. *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 a Markov chain

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.

associated to 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 with $\mathbf{A}_{i,j} \in \{0,1\}$, $\forall i,j$ 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. □

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 [25]. "Internet RLT" is a router-level Internet graph collected from traceroute measurements of 23 monitors distributed over the world [12]. 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 Appendix B.

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.

We also perform an extreme experiment that focuses on studying the impact of loosely connected components over 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, this graph is referred to as G_{AB}.

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. Table 2 shows a summary of the relative bias of the estimates. We observe that FS bias is often much smaller than the biases of MultipleRW and SingleRW. In the Flickr graph the relative bias of MultipleRW and SingleRW is 6 to 7 larger than the value of the assortative mixing coefficient, r. It is worth mentioning that in the G_{AB} graph, SingleRW estimates $\hat{r} = 0$ over all 100 runs. This is because SingleRW only estimated the assortative mixing of either subgraph A or subgraph B, which are both zero. In the same scenario MultipleRW does almost as bad as SingleRW while FS is able to accurately estimate r.

		Bias in respect to r		
Graph	r	FS	MultipleRW	SingleRW
Flickr	0.007	8%	752%	−619%
LiveJournal	0.07	−0.5%	−12%	1%
G_{AB}	0.08	0.01%	70%	100%
Internet	0.17	3%	2%	17%
Youtube	−0.03	0.001%	2%	−1%

Table 2: Assortative mixing coefficient estimate bias; 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.

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 (6). We choose to estimate the CCDF instead of the density because the CCDF is the plot of choice when it comes to display degree distributions. Each simulation consists of $10,000$ runs (sample paths) used to compute the empirical NMSE (equation (1)) of the CCDF (which we denote the CNMSE to avoid confusion with the NMSE of θ_i). 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.

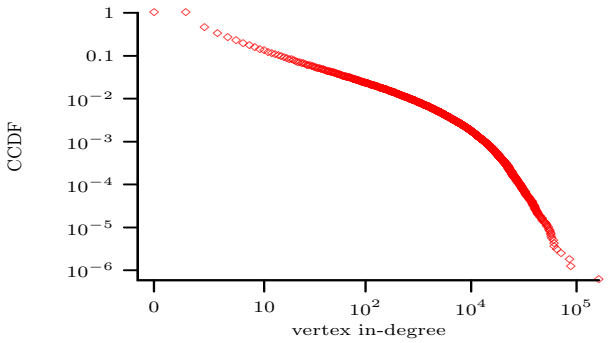

Figure 2: (Flickr) Log-log plot of the in-degree CCDF.

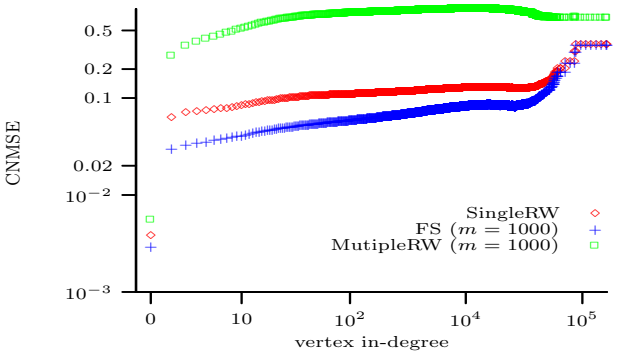

Figure 3: (LCC of Flickr) The log-log plot of the CNMSE of the in-degree distribution estimates with budget $B = |V|/100$.

Consider first two representative results from the Flickr graph, whose in-degree CCDF (complementary cumulative distribution function) log-log plot is shown in Figure 2. The sampling budget is $B = 18,123 = |V|/100$, 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 when there are no disconnected components. Figure 3 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 4 shows a log-log

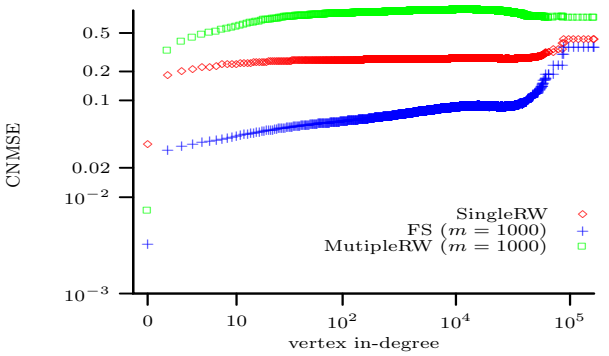

Figure 4: (Flickr) The log-log plot of the CNMSE of the in-degree distribution estimates with budget $B = |V|/100$.

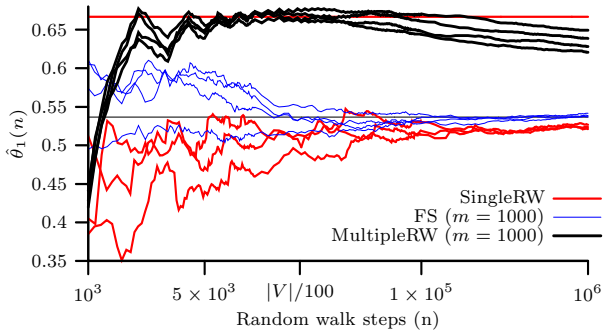

Figure 5: (LCC of Flickr) Four sample paths of $\hat{\theta}_1$ ($\theta_1 = 0.53$) as a function of the number of steps n (horizontal axis in log scale).

plot of the CNMSE of the in-degree distribution estimates. Contrasting the plots shown in Figures 3 and 4 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 5 focuses on four runs (sample paths) of the simulation over the complete Flickr graph. Figure 5 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 5 shows that all four FS sample paths (runs) quickly converge to the value of θ_1. For SingleRW, three out 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 jump 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 5) 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 6 shows a log-log plot of the CCDF of the out-degrees. The log-log plot of the CNMSE is shown in Figure 7 for FS

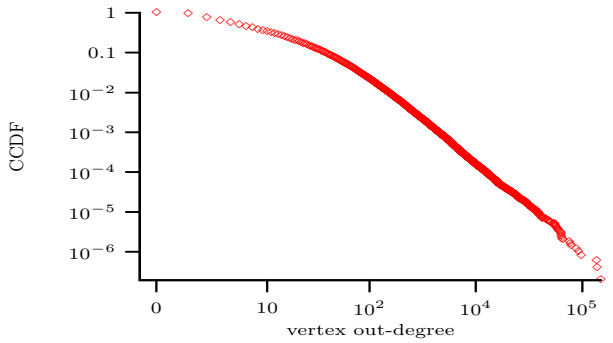

Figure 6: (Livejournal) Log-log plot of the out-degree CCDF.

($m = 100$), SingleRW, and MultipleRW ($m = 100$) with sampling budget $B = |V|/10$. From Figure 7 we see that estimates of vertices with small out-degrees in FS are up to one order of magnitude more accurate than those obtained from both SingleRW and MultipleRW.

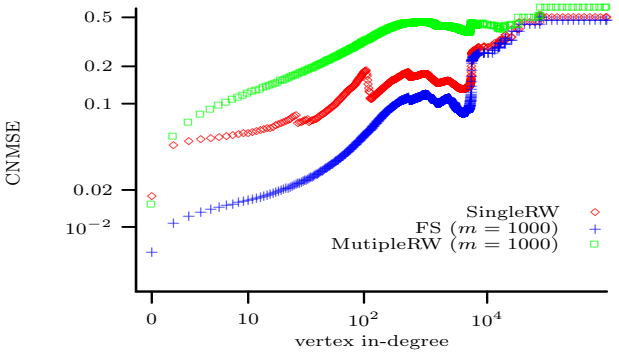

Figure 7: (Livejournal) The log-log plot of the CNMSE of the out-degree distribution estimation with sampling budget $B = |V|/100$ (CNMSE over 10,000 runs).

The next experiment focuses on studying the impact of loosely connected components over the degree 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 8 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

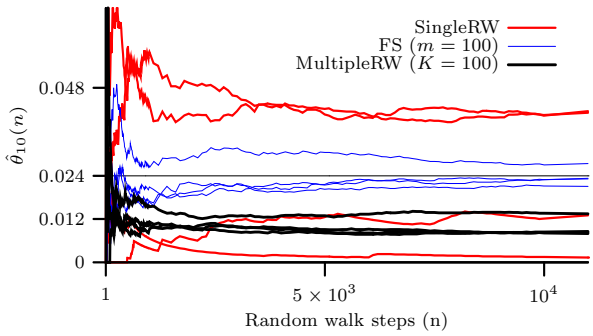

Figure 8: (G_{AB} graph) Four paths of $\hat{\theta}_{10}$ as a function of the number of steps n ($\theta_{10} = 0.024$).

(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 9 also shows that the CNMSE for FS, SingleRW, and MultipleRW, that of FS is consistently lower.

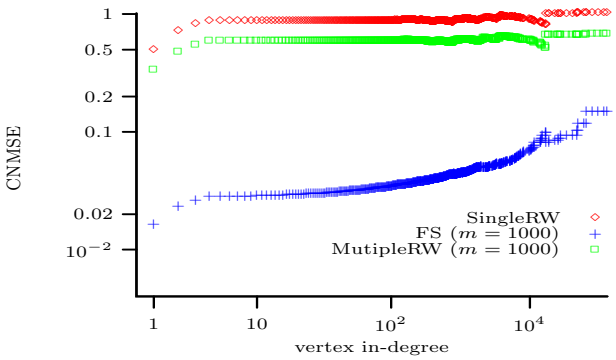

Figure 9: (G_{AB} graph) The log-log plot of the CNMSE of the degree distribution estimation with sampling budget $B = |V|/100$ (CNMSE over 10,000 runs).

6.3 FS v.s. Stationary SingleRW and MultipleRW

We now compare FS with SingleRW and MultipleRW, when the latter two start in steady state. Figure 10 shows the results (over the Flickr graph) of the same simulation scenario used to obtain the results in Figure 4, 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 Frontier v.s. Random 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 sec-

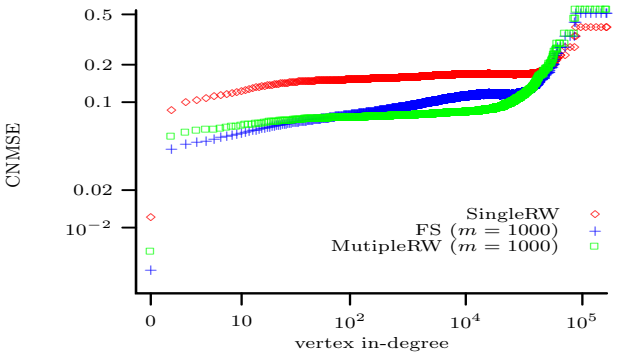

Figure 10: (Flickr) The log-log plot of the CNMSE of the in-degree distribution estimation of MultipleRW and SingleRW starting in steady state; sampling budget $B = |V|/100$ (NMSE over 10,000 runs).

tion 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 (6) (the estimator used for sampled vertices is trivial).

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 11 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 (2) and (3). 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 (2) and (3) of our model in Section 3. We also observe that the accuracy of FS ($m = 1000$) closely matches the accuracy of random edge sampling.

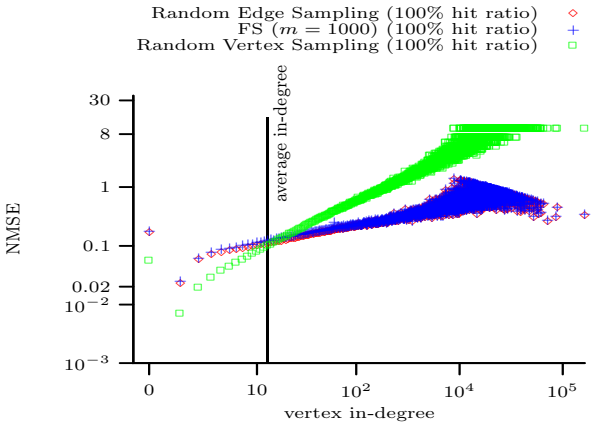

Figure 11: (Flickr) The log-log plot shows the NMSE of the in-degree distribution estimation with budget $B = |V|/100 = 18612$ (CNMSE over 10,000 runs).

Some complex networks exhibit a sparse user-id space.

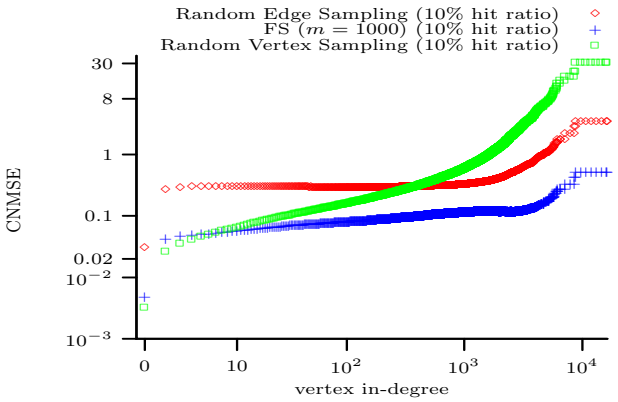

Figure 12: (Livejournal) The log-log plot shows the CNMSE of the in-degree distribution estimation with budget $B = |V|/100 = 52844$ (CNMSE over 10,000 runs).

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 [29], the following experiment assumes that only 10% of the user-ids are valid, i.e., in average only one in every ten randomly sampled vertices are valid. We denote this value (10%) to be the *hit ratio*. For random edge sampling we assume a *hit ratio* of 1% (the choice of 1% is arbitrary). Figure 12 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 (in 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.

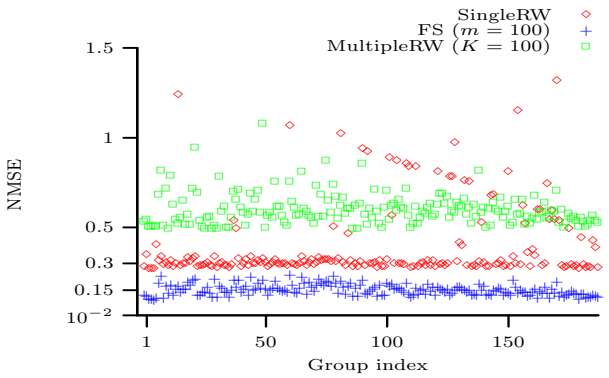

Figure 13: (Flickr) The NMSE of the density estimates of the most popular groups in the Flickr graph.

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 [25]. 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 13 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	Budget(B)	C	$\hat{C} \pm \sqrt{MSE}$
G_{AB}	$\|V\|/10$	10^{-4}	$10^{-4} \pm 10^{-5}$
Flickr	$\|V\|/20$	0.05	0.05 ± 10^{-3}
LiveJournal	$\|V\|/50$	0.06	0.06 ± 0.01

Table 3: Frontier sampling: 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 little difference between FS ($m = 100$), SingleRW, MultipleRW ($m = 100$). Table 3 presents the average and the root mean squared error ($\sqrt{MSE}$) over 100,000 runs of the estimates obtained using FS over three graphs. From the results of Table 3 we see that FS accurately estimates the global clustering coefficient.

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 are applied to characterize complex networks. Applications include, but are not limited to estimating: characteristics of a population [34] (e.g. estimation of HIV seroprevalence among drug users [23]), content density in peer-to-peer networks [15, 22, 28, 33], uniformly sampling Web pages from the Internet [16, 31], 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) [15, 16, 31, 4, 33]. The accuracy of RW and Metropolis-RW (MRW) is compared in [14, 28], 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 [30].

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, 36], 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

We would like to thank Weibo Gong for many helpful discussions and Alan Mislove for kindly making available some of the data used in this paper. This research was sponsored by the ARO under MURI W911NF-08-1-0233, and the U.S. Army Research Laboratory and the U.K. Ministry of Defence and was accomplished under Agreement Number W911NF-06-3-0001. The views and conclusions contained in this document are those of the author(s) and should not be interpreted as representing the official policies, either expressed or implied, of the U.S. Army Research Laboratory, the U.S. Government, the U.K. Ministry of Defence or the U.K. Government. The U.S. and U.K. Governments are authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation hereon.

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

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

[14] 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.

[15] 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.

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

[17] 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.

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

[19] 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.

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

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

[22] 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.

[23] 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.

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

[25] 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.

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

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

[28] 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.

[29] Bruno Ribeiro, William Gauvin, Benyuan Liu, and Don Towsley. On MySpace account spans and double Pareto-like distribution of friends. Technical Report UM-CS-2010-001, UMass Amherst, Jan 2010.

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

[31] 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.

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

[33] 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.

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

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

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

Graph	B (sampling budget)	Sampling prob. error FS	MRW	SRW
Internet RLT	100	17%	257%	156%
YouTube	20	43%	236%	216%
Hep-Th	20	36%	1510%	781%

Table 4: Relative worst-case difference between the steady state and the transient edge sampling probabilities after $B - K$ steps. Frontier edge sampling probabilities are closer to steady state in all graphs. Legend: (FS) = Frontier sampling ($K = 10$), (SRW) = Single ($K = 1$) Random Walker, and (MRW) = Multiple ($K = 10$) Random Walkers.

APPENDIX

A. PROOF OF THE FRONTIER SAMPLING THEOREM

In what follows we restate Theorem 5.2 and present a proof.

Theorem. *If G is connected and non-bipartite, then FS is asymptotically stationary and has a unique stable distribution where: (1) edges are sampled with probability $1/|E|$, (2) sampled edges form a stationary sequence, and (3) the sequence satisfies the Strong Law of Large Numbers (Theorem 4.1).*

PROOF. Consider the $(n-1)$-st step of Frontier sampling. The reader may find Figure 1 helpful in following the proof. Let $L_n = (v_1, \ldots, v_m)$ be the state of Frontier sampling before the n-th step. Clearly $L_n \in V^m$. In what follows 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 frontier sampling algorithm): Select a vertex $v \in L_n$ with probability proportional to $\deg(v)$ and then replace element 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

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

Thus, L_n is able to transition 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, we conclude that Frontier sampling is 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-ary Cartesian product 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 $j \neq i$. Note that $|E_m| = m|V|^{m-1}|E|$.

Now we need to prove that the distribution of L_∞ is stable and unique. For this we only need to show that the random walk over G^m is ergodic. A random walk (Markov chain) is ergodic when it is aperiodic and recurrent non-null. Recall that the random walk over G is ergodic. The probability that Frontier sampling transitions from $L_n \in V^m$ to $L_{n+1} \in V^m$ such that L_n and L_{n+1} only differ in their i-th element is always greater than zero, otherwise there is an infinite increasing degree sequence in the vertices of G. But this is not possible as the random walk over G is recurrent non-null (an infinite increasing degree sequence would be a sink in the random walk over G). Thus, any finite sequence of transitions $\{L_{n+w}\}_{w=1}^{\Delta}$ that only updates its i-th element has probability greater than zero. Thus, as the sequence $\{L_{n+w}\}_{w=1}^{\Delta}$ is also a single random walk over G, it is aperiodic for any chosen $i = 1, \ldots, m$. Thus, a random walker over G^m must also be aperiodic. We can use the same argument to show that the random walk over G^m is recurrent non-null. As random walk over G^m is ergodic, we have that $L^\star$ is distributed according to the steady state distribution of a random walk over G^m

$$P[L_\infty = (v_1, \ldots, v_m)] = \frac{\sum_{i=1}^m \deg(v_i)}{2m|V|^{m-1}|E|},$$

where $L_\infty \equiv \lim_{n\to\infty} L_n$, which is unique and stable (similar to a single random walker as seen in Section 4).

The rest of the proof is straightforward. Each edge in G^m is actually an edge in G. As each edge in G is copied m times into G^m, we have that edges in G are also sampled uniformly at random in a random walk over G^m. As Frontier sampling is a random walk over G^m, its samples form a stationary sequence and follow the Strong Law of Large Numbers seen in Theorem 4.1. The same is true for the sequence of sampled vertices. □

B. CONVERGENCE TO UNIFORM EDGE SAMPLING

A question we seek to answer with our simulations is how fast these random walk methods converge to their stationary edge sampling probabilities. In this simulation we set $K \in \{1, 10\}$ (number of independent random walkers), $m = 10$ (Frontier sampling dimension) and restrict our analysis to the largest connected component of the three graphs in our datasets with the smallest number of vertices (in order to speed the computation): "Internet RLT", "YouTube", and "Hep-th". Let $p_{u,v}^{(B)}$ denote the probability that a random walker, whose initial vertex is chosen uniformly at random, samples edge (u, v) at its the end of its sampling budget B. To measure the convergence to the stationary edge sampling probability, we use the largest relative difference between the stationary sampling probability $1/|E|$ and $p_{u,v}^{(B)}$:

$$\max_{(u,v)\in E} 1 - \frac{p_{u,v}^{(B)}}{1/|E|}.$$

Table 4 presents a Monte Carlo estimate of this relative difference. The 95% confidence interval of the Monte Carlo simulation is ±1%. Our estimates show that the difference between the transient and the stationary edge sampling probabilities of independent random walkers are between 5 and 42 times larger than the difference of Frontier sampling. This means that Frontier sampling converges faster to stationarity edge sampling probability.