A Walk in Facebook: Uniform Sampling of Users in Online Social Networks

Minas Gjoka, Maciej Kurant, Carter T. Butts, Athina Markopoulou
California Institute for Telecommunications and Information Technology (CalIT2)
University of California, Irvine

Abstract—Our goal in this paper is to develop a practical framework for obtaining an unbiased sample of users in an online social network (OSN), by crawling its social graph. Such a sample allows to estimate any user property and some topological properties as well. To this end, we consider and compare several candidate crawling techniques. Two approaches that can produce unbiased samples are the Metropolis-Hasting random walk (MHRW) and a re-weighted random walk (RWRW). Both have pros and cons, which we demonstrate through a comparison to each other as well as to the ground-truth. In contrast, using Breadth-First-Search (BFS) or a simple Random Walk (RW) leads to substantially biased results. Second, and in addition to offline performance assessment, we introduce online formal convergence diagnostics to assess sample quality during the data collection process. We show how these diagnostics can be used to effectively determine when a random walk sample is of adequate size and quality. Third, as a case study, we apply the above methods to sample Facebook users. We collect the first, to the best of our knowledge, unbiased sample of Facebook users. We make it publicly available and we use it to characterize several key properties of Facebook.

Index Terms—Measurements, online social networks, Facebook, sampling, crawling, random walks, convergence diagnostics.

I. INTRODUCTION

Online Social Networks (OSNs) have recently emerged as a new Internet killer-application. The adoption of OSNs by Internet users is off-the-charts with respect to almost every metric. In November 2010, Facebook, the most popular OSN, counted more than 500 million members; the total combined membership in the top five OSNs (Facebook, QQ, Myspace, Orkut, Twitter) exceeded 1 billion users. Putting this number into context, the population of OSN users is approaching 20% of the world population and it is more than 50% of the world’s Internet users. According to Nielsen, users all over the world now spend over 110 billion minutes on social media sites per month, which accounts for 22% of all time spent online, surpassing even email messaging as the most preferred online activity. According to Alexa, a well-known traffic analytics website, Facebook is the second most visited website on the Internet (the first being Google) with each user spending 30 minutes on average per day on the site (more than the time spent on Google). Four of the top five OSNs are also contained in Alexa’s top 15 websites in regard to traffic rankings. Clearly, OSNs in general, and Facebook in particular, have become an important phenomenon on the Internet, which is worth studying.

OSN data are of interest to multiple disciplines and can be used, for example, to design viral marketing strategies, to model the spread of influence through social networks, to conduct low cost surveys in large scale, to detect hidden community structures etc. From a networking/systems perspective, there are several reasons to study and better understand OSNs. One motivation is to optimize content delivery to users. Indeed, the aforementioned statistics show that OSNs already play an increasingly important role in generating and re-directing Internet traffic. OSN operators can optimize data storage in the cloud to reduce response times, e.g., by designing middleware that takes into account the characteristics of the social graph to achieve data locality while minimizing replication. Network operators can exploit OSNs to optimize content delivery by predicting user demand and pre-fetching and caching content, as proposed in [4, 5] performed a measurement study of the network-level effects of popular third party applications on Facebook. Another use of OSNs is to create algorithms that can exploit trusted or influential users, e.g., to thwart unwanted communication while not impeding legitimate communication [6]; to utilize social trust for collaborative spam filtering [7]; or to enable online personas to cost-effectively obtain credentials [8].

The immense interest generated by OSNs has given rise to a number of measurement and characterization studies that attempt to provide a first step towards their understanding. Only a very small number of these studies are based on complete datasets provided by the OSN operators [9, 10]. A few other studies have collected a complete view of specific parts of OSNs; e.g., [11] collected the social graph of the Harvard university network. However, the complete dataset is typically unavailable to researchers, as most OSNs are unwilling to share their company’s data even in an anonymized form, primarily due to privacy concerns.

Furthermore, the large size and access limitations of most OSN services (e.g., login requirements, limited view, API query limits) make it difficult or nearly impossible to fully crawl the social graph of an OSN. In many cases, HTML
scraping is necessary, which increases the overhead multifold. Instead, it would be desirable to obtain and use a small but representative sample.

Therefore, sampling techniques become essential for estimation of OSN properties, in practice. While sampling can, in principle, allow precise inference from a relatively small number of observations, this depends critically on the ability to draw a sample with known statistical properties. The lack of a sampling frame (i.e., a complete list of users, from which individuals can be directly sampled) for most OSNs makes principled sampling especially difficult. To evade this limitation, our work focuses on sampling methods that are based on crawling of friendship relations - a fundamental primitive in any OSN.

Our goal in this paper is to provide a framework for obtaining an asymptotically unbiased sample of OSN users by crawling the social graph. We provide practical recommendations for appropriately implementing the framework, including: the choice of crawling technique; the use of online convergence diagnostics; and the implementation of high-performance crawlers. We then apply our framework to an important case-study - Facebook. More specifically, we make the following three contributions.

Our first contribution is the comparison of several candidate graph-crawling techniques in terms of sampling bias and efficiency. First, we consider Breadth-First-Search (BFS) - the most widely used technique for measurements of other OSNs [9], [12] and Facebook [13]. BFS sampling is known to introduce bias towards high degree nodes, which is highly non-trivial to characterize analytically [14], [15]. Second, we consider Random Walk (RW) sampling, which also leads to bias towards high degree nodes, but at least its bias can be quantified by Markov Chain analysis and thus can be corrected via appropriate re-weighting (RWRW) [16], [17]. Then, we consider the Metropolis-Hastings Random Walk (MHRW) that can directly yield a uniform stationary distribution of users. This technique has been used in the past for P2P sampling [18], recently for a few OSNs [19], [20], but not for Facebook. Finally, we also collect a sample (UNI) that represents the ground truth, i.e., a uniform sample of Facebook userIDs, selected by a rejection sampling procedure from Facebook's 32-bit ID space. Such ground truth is in general unavailable, and our ability to use it as a basis of comparison is therefore a valuable asset of this study. We compare all sampling methods in terms of their bias and convergence properties. We find that MHRW and RWRW are both able to collect asymptotically unbiased samples, while BFS and RW result in significant bias in practice. We also compare MHRW to RWRW, via analysis, simulation and experimentation and discuss their pros and cons. The former provides a sample ready to be used by non-experts, while the latter is more efficient for all practical purposes.

Our second contribution is that we introduce, for the first time in this context, the use of formal convergence diagnostics (namely Geweke and Gelman-Rubin) to assess sample quality in an online fashion. These methods allow us to determine, in the absence of a ground truth, when a sample is adequate for subsequent use, and hence when it is safe to stop sampling, which is a critical issue in implementation.

Our third contribution is that we apply our framework to an important case-study - Facebook. To the best of our knowledge, this is the first time that all the aforementioned techniques have been applied to and compared on Facebook. We crawl Facebook's web front-end, which is highly non-trivial due to various access limitations, and we provide guidelines for the practical implementation of high-performance crawlers. We obtain the first unbiased sample of Facebook users, which we make publicly available [21]; we have received approximately 200 requests for this dataset in the last six months. Finally, we use the collected datasets to characterize several key properties of Facebook, including user properties (e.g., privacy settings) and topological properties (e.g., the node degree distribution, clustering and assortativity, and connectivity between regional networks).

Obtaining a uniform sample of OSN users is interesting on its own right, as it allows to estimate any user property, such as age, privacy settings or any other user attribute. We note that degree distribution is a specific user (node) property that can be estimated from a uniform sample of users and happens to contain information about the topology. In addition, uniform sampling of users is a first step towards estimating topological properties or the topology itself. For example, in Section VI we collect and study the egonets (i.e., the one-hop neighborhood) of all sampled users and use them to estimate the clustering coefficient and assortativity, which are topological properties. In Section VI-C we use the sample of users to estimate the topology at the coarser granularity of countries.

The structure of the rest of paper is as follows. Section II discusses related work. Section III describes the sampling methodology, including the assumptions and limitations, the candidate crawling techniques and the convergence diagnostics. Section IV describes the data collection process, including the implementation of high-performance crawlers, and the collected data sets from Facebook. Section V evaluates and compares all sampling techniques in terms of efficiency (convergence of various node properties) and quality (bias) of the obtained sample. Section VI provides a characterization of some key Facebook properties, based on the MHRW sample. Section VII concludes the paper. The appendices elaborate on the following important points: (A) the uniform sample obtained via userID rejection sampling, referred to as “ground truth”; (B) the lack of temporal dynamics in Facebook, in the timescale of our crawls and (C) comparison of the sampling efficiency of MHRW vs. RWRW via analysis, simulation and experimentation.

II. RELATED WORK

Broadly speaking, there are two types of work most closely related to this paper: (i) sampling techniques, focusing on the quality and efficiency of the sampling technique itself and (ii) characterization studies, focusing on the properties of online
social networks based on a collected sample. In this section we review this related literature and we position our work with respect to it.

A. Graph sampling techniques

Sampling techniques can be roughly classified into two categories: graph traversal techniques and random walks. In graph traversal techniques, each node in the connected component is visited exactly once, if we let the process run until completion. These methods vary in the order in which they visit the nodes; examples include Breadth-Search-First (BFS), Depth-First Search (DFS), Forest Fire (FF) and Snowball Sampling (SBS) [22]. BFS, in particular, is a basic technique that has been used extensively for sampling OSNs in past research [9], [12], [13], [23], [24]. One reason for this popularity is that an (even incomplete) BFS sample collects a full view (all nodes and edges) of some particular region in the graph. However, BFS has been shown to lead to a bias [25], [26] towards high degree nodes in various artificial and real-world topologies. Our work also confirms the bias of BFS when sampling Online Social Networks. It is also worth noting that BFS and its variants lead to samples that not only are biased but also do not have provable statistical properties. Efforts to analytically compute and correct the bias for random graphs with a given degree distribution include the works by Achlioptas et al. [27] and by Kurant et al. [14].

Ye et al. [28] perform crawling simulations on previously collected samples of online social graphs and examine the efficiency, sensitivity and bias of four graph traversal methodologies. Their findings agree with our results with regards to the bias of BFS. Compared to [28], we perform online crawling of large scale online social networks and we thoroughly address the challenge of crawling bias by using principled methods.

Random walks on graphs are well-studied topic; see [29] for an excellent survey. They have been used for sampling the World Wide Web (WWW) [30], [31], peer-to-peer networks [17], [18], [32], and other large graphs [33]. In [30], Henzinger et al. use a random walk with jumps to achieve near-uniform sampling of URLs in the WWW. Their setting is different since the URL graph is directed and random jumps are needed to avoid entrapment in a region of the web. Gkantsidis et al. [32] simulate various peer-to-peer topologies and show that random walks outperform flooding (BFS) with regards to searching for two cases of practical interest. They also argue that random walks simulate uniform sampling well with a comparable number of samples. Leskovec et al. [33] explore several sampling methods and compare them in terms of various graph metrics; their evaluations in static and dynamic graphs show that random walks perform the best.

Random walks are also biased but their bias can be analyzed using classical results from Markov Chains, and corrected either at the end or during the data collection. In the context of peer-to-peer sampling, Rasti et al. [17] have applied re-weighting at the end to yield unbiased estimators of peer properties. A re-weighted random walk is considered as a special case of Respondent-Driven Sampling (RDS) [34]. if sampling is with replacement and exactly one neighbor is selected in every step [16]. Alternatively, the random walk can be modified during the collection using the Metropolis rules so as to achieve, by design, any desired stationary distribution [35], [36]. In our case, we would like this target distribution to be the uniform. This algorithm, known as Metropolis-Hasting Random Walk (MHRW) has been applied to peer-to-peer networks by Stutzbach et al. in [18]: they use a Metropolized Random Walk with Backtracking (MRWB), modified to deal with peer churn, to select a representative sample of peers in a peer-to-peer network and demonstrate its effectiveness, through simulations over artificially generated graphs as well as with measurements of the Gnutella network. In [17], [19], Rasti et al. provide a comparison of Re-Weighted Random Walk (or RDS in their terminology) with Metropolis-Hasting Random Walk.

Compared to the aforementioned papers, our work is most closely related to the random walk techniques. We obtain unbiased estimators of user properties of Facebook using MHRW and RWRW and we compare the two through experiments and analysis; BFS and RW (without re-weighting) are used mainly as baselines for comparison. We accompany the basic crawling techniques with formal, online convergence diagnostic tests using several node properties, which, to the best of our knowledge, has not been done before in measurements of such systems. The closest to such formal diagnostics is the work by Latapy et al. [37] which studies how the properties of interest evolve when the sample grows to practically detect steady state. We also implement multiple parallel chains, which have also been recently used in [17] but started at the same node (while we start from different nodes, thus better utilizing the multiple chains). We demonstrated that random walks, whose bias can be analyzed and corrected, are able to estimate properties of users in online social networks remarkably well in practice. We also find that correcting for the bias at the end (RWRW), rather than during the walk (MHRW), appears more efficient, for all practical purposes, due to faster mixing in the Markov chain - a finding that agrees with [17].

We apply the measurement techniques to online social networks, instead of peer-to-peer and other complex networks, and we study characteristics specific to that context. We are the first to perform unbiased sampling of large scale OSNs. Krishnamurthy et al. [20] ran a single Metropolis Random Walk, inspired by [18], on Twitter as a way to verify the lack of bias in their main crawl used throughout their paper. However, the Metropolis algorithm was not the main focus of their paper and Twitter is a directed graph which requires different treatment. Parallel to our work, Rasti et al. [19] also applied similar random walk techniques to collect unbiased samples of Friendster.

Previous work on the temporal dynamics of social networks includes [19], [38–41]. Kumar et al. [38] studied the structure and evolution of Flickr and Yahoo! 360, from datasets provided by their corresponding operators. Backstrom et al. [39] presented different ways in which communities in social networks grow over time and [40] proposed a method for modeling relationships that change over time in a social network. Willinger et al. [41] proposed a multi-scale approach to study dynamic social graphs at a coarser level of granularity;
Section VI-C presents such an attempt at one time point. Rasti et al. [19] evaluated the performance of random walks in dynamic graphs via simulations and show that there is a tradeoff between number of parallel samplers, churn and accuracy. In our work, we assume that the social graph remains static during the duration of the crawl, which we show in Appendix B to be a good approximation in Facebook. Therefore, we do not consider dynamics, which are essential in other sampling contexts.

A unique asset of our study is the collection of true uniform sample of OSN users through rejection sampling of userIDs (UNI), which served as ground truth in this paper; see Section III-C. It is important to note that UNI yields a uniform sample of users regardless of the allocation policy of userIDs by the OSN; see Appendix A for a proof of this nonintuitive result. UNI is a star random node sampling scheme; this is different from the induced subgraph random node sampling schemes presented in [33], [42].

B. Characterization studies of OSNs

Several papers have measured and characterized properties of OSNs. In [43], Krishnamurthy presents a summary of the challenges that researchers face in collecting data from OSNs. In [9], Ahn et al. analyze three online social networks; one complete social graph of Cyworld obtained from the Cyworld provider, and two small samples from Orkut and Myspace crawled with BFS. Interestingly, in our MHRW sample we observe a multi-scaling behavior in the degree distribution, similarly with the complete Cyworld dataset. In contrast, the crawled datasets from Orkut and Myspace in the same paper were reported to have simple scaling behavior. We believe that the discrepancy is due to the bias of the BFS-sampling they used. In [12], [23], Mislove et al. studied the properties of the social graph in four popular OSNs: Flickr, LiveJournal, Orkut, and YouTube. Their approach was to collect the large Weakly Connected Component, also using BFS; their study concludes that OSNs are structurally different from other complex networks.

[11], [13], [44] are closely related to our study as they also study Facebook. Wilson et al. [13] collect and analyze social graphs and user interaction graphs in Facebook between March and May 2008. In terms of methodology, their approach is what we refer to as a Region-Constrained BFS. They exhaustively collect all open user profiles and their list of friends in the 22 largest regional networks (out of the 507 available). First, such Region-Constrained BFS might be appropriate to study particular regions, but it does not provide Facebook-wide information, which is the goal of our study. Second, it seems that the percentage of users in the social graph retrieved in [13] is 30%-60% less than the maximum possible in each network. In terms of findings, some noteworthy differences from [13] are that we find larger values of the degree-dependent clustering coefficient, significantly higher assortativity coefficient, and that the degree distribution does not obey a power law. In terms of results, the main conclusion in [13] is that the interaction graph should be prefered over social graphs in the analysis of online social networks, since it exhibits more pronounced small-world clustering. In our work, we collect a representative sample of the social graph which can allow us to fetch a representative sample of user profiles Facebook-wide, appropriate for user interaction analysis. However interaction graphs such as those presented in [13], [24], [46] are out of the scope of this paper.

[11] and [44] have also made publicly available and analyzed datasets corresponding to university networks from Facebook with many annotated properties for each student. In contrast, we collect a sample of the global Facebook social graph.

Other works that have measured properties of Facebook include [24], [45], [47], [48]. In [45], Krishnamurthy et al. examine the usage of privacy settings in Myspace and Facebook, and the potential privacy leakage in OSNs. Compared to that work, we have one common privacy attribute, “View friends”, for which we observe similar results using our unbiased sample; we also have additional privacy settings and the one-hop neighborhood for every node, which allows us to analyze user properties conditioned on their privacy awareness. Bonneau et al. [47] demonstrate that many interesting user properties can be accurately approximated just by crawling “public search listings”.

Finally, there is a large body of work on the collection and analysis of datasets for platforms or services that are not pure online social networks but include social networking features, e.g., user ratings, video categories, etc.) could be exploited to improve the caching strategy.

C. Comparison to our prior and ongoing work.

The conference version of this work appeared in [52]. This paper is significantly revised and extended to include the following materials. (i) A detailed discussion of the uniform userID rejection sampling, which is used as ground truth in this work; see Section II-C and Appendix A. (ii) An empirical validation of the assumption that the social graph is static in the time scales of the crawl; see Appendix B. (iii) A detailed comparison of MHRW and RWRR methods and the finding that RWRR is more efficient for all practical purposes; see Section V-A for an experimental comparison on Facebook and Appendix C for a comparison via analysis and simulation.
(iv) An extended section on the characterization of Facebook based on a representative sample; see Section [VII] for additional graphs on node properties and topological characteristics, new results on privacy settings, and a coarser granularity view of the Facebook social graph at the country level. (iv) A comprehensive review of related work in Section [II]

This work focuses on providing a practical sampling framework (e.g., choosing a crawling method, utilizing online convergence diagnostics, implementation issues) of a well-connected OSN social graph.

III. SAMPLING METHODOLOGY

A. Scope and Limitations

Scope and Assumptions. We consider OSNs, whose social graph can be modeled as an undirected graph \( G = (V, E) \), where \( V \) is a set of nodes (users) and \( E \) is a set of edges. We make the following assumptions.

A1 We focus on the friendship graph, which is undirected: edges correspond to mutual friendship relations.

A2 We are interested only in the publicly declared friendships, which, under default privacy settings, are available to any logged-in user.

A3 We assume that the friendship graph is well connected, which is the case in Facebook but not necessarily in other OSNs, such as Last.fm [53]. We are not interested in isolated users, i.e., users without any declared friends.

A4 We assume that the friendship graph remains static during the duration of our crawling. This is validated in our Facebook experiments; the reader is referred to Appendix B for details. We developed high performance crawlers that are able to obtain a converged sample in one day, a time scale much shorter than the timescale in which Facebook changes. Therefore, accounting for temporal dynamics is out of the scope of this paper.

A5 The OSN should support crawling, e.g., through some mechanism such as an API call or HTML scraping. This allows us, once we visit a node, to observe its one-hop neighbors and decide whether to sample them or not.

This is a very mild assumption in practice, as exploring the friendship relations is a fundamental building block of any OSN.

Goal and applications. Our goal is to obtain a uniform sample of OSN nodes (users) by crawling the social graph. This is interesting on its own right, as it allows to estimate any user property, such as age, privacy settings etc. It also allows us to estimate some specific topological properties such as node degree distribution, clustering and assortativity, as in Section [VI-A]. The last two properties are computed based on the one-hop neighborhood of nodes. Furthermore, as we show in Section [VI-C] node sampling can be used to estimate the topology at coarser granularity, e.g., at the country, regional or university-level. Therefore, a random sample of nodes, obtained using our methodology, is a useful building block towards the goal of topology estimation.

Limitations. We would like to emphasize that although our collected samples can be used for estimating user properties, it is not directly applicable for reconstructing the topology. For example, a typical erroneous use of our datasets is to use the nodes and edges in the sample, possibly together with their neighbors (nodes and edges in the egonets) to reconstruct the Facebook social graph; a topology reconstructed this way in not representative. Therefore, if topology and not users are of interest, careful use of our techniques is needed along the lines of the previous paragraph.

We also note that, although all our assumptions are true in Facebook, this may not be the case in all OSNs. Therefore, our methodology should be applied with caution. If some of the assumptions do not hold, pre-processing may be needed or the methodology may not be applicable. For example, A1 is necessary for the analysis, but does not hold in networks such as Twitter [20] where edges are directed; some proposals for “symmetrizing” the relation may be considered [53], [54]. A2 limits the scope to OSN users with open profile. If A3 does not hold, we may be able to consider several relations whose union leads to a connected graph [53]. A4 depends both on the temporal dynamics of the graph and on the efficient implementation of the crawlers, but it is critical that it holds; diagnostic tests should be applied to ensure it applies. A5 is typically true in OSNs; even if crawling is not explicitly available, there may be ways to infer the one-hop neighborhood, e.g., through co-membership to groups, shared partners or various interactions.

B. Sampling Methods

The crawling of the social graph starts from an initial node and proceeds iteratively. In every operation, we visit a node and discover all its neighbors. There are many ways in which we can proceed, depending on which neighbor we choose to visit next. In this section, we describe the sampling methods implemented and compared in this paper.

1) Breadth First Search (BFS): At each new iteration the earliest explored but not-yet-visited node is selected next. As this method discovers all nodes within some distance from the starting point, an incomplete BFS is likely to densely cover only some specific region of the graph.

2) Random Walk (RW): In the classic random walk [29], the next-hop node \( w \) is chosen uniformly at random among the neighbors of the current node \( v \). I.e., the probability of moving from \( v \) to \( w \) is

\[
P_{v,w}^{RW} = \begin{cases} \frac{1}{k_v} & \text{if } w \text{ is a neighbor of } v, \\ 0 & \text{otherwise.} \end{cases}
\]

The random walk is [29] inherently biased. Assuming a connected graph and aperiodicity, the probability of being at the particular node \( v \) converges to the stationary distribution \( \pi_v^{RW} = \frac{k_v}{2|E|} \), i.e., the classic RW samples nodes w.p. \( \pi_v^{RW} \sim k_v \). This is clearly biased towards high degree nodes; e.g., a node with twice the degree will be visited by RW twice more often. In Section [IV] we show that several other node properties are correlated with the node degree and thus estimated with bias by RW sampling.

4“Observing” a node (e.g., via a user’s “Friends list”) is inexpensive in practice, while “visiting” a node is typically associated with a cost.
3) Re-Weighted Random Walk (RWRW): A natural next step is to crawl the network using RW, but to correct for the degree bias by an appropriate re-weighting of the measured values. This can be done using the Hansen-Hurwitz estimator \[55\] as first shown in \[16, 56\] for random walks and also later used in \[17\]. Consider a stationary random walk that has visited \(V = v_1, ..., v_n\) unique nodes. Each node can belong to one of \(m\) groups with respect to a property of interest \(A\), which might be the degree, network size or any other discrete-valued node property. Let \((A_1, A_2, ..., A_m)\) be all possible values of \(A\) and corresponding groups: \(\cup^m A_i = V\). E.g., if the property of interest is the node degree, \(A_i\) contains all nodes \(u\) that have degree \(k_u = i\). To estimate the probability distribution of \(A_i\), we need to estimate the proportion of nodes with value \(A_i, i = 1, ..m:\)

\[
\hat{p}(A_i) = \frac{\sum_{u \in A_i} 1/k_u}{\sum_{u \in V} 1/k_u}
\]

Estimators for continuous properties can be obtained using related methods, e.g. kernel density estimators.

4) Metropolis-Hastings Random Walk (MHRW): Instead of correcting the bias after the walk, one can appropriately modify the transition probabilities so that it converges to the desired uniform distribution. The Metropolis-Hastings algorithm \[35\] is a general Markov Chain Monte Carlo (MCMC) technique \[36\] for sampling from a probability distribution \(\pi\) that is difficult to sample from directly. In our case, we would like to sample nodes from the uniform distribution \(\mu_v = \frac{1}{|V|}\). This can be achieved by the following transition probability:

\[
P^{\text{MHRW}}_{v \rightarrow w} = \begin{cases} 
\frac{1}{|E|} \cdot \min(1, \frac{k_w}{k_v}) & \text{if } w \text{ is a neighbor of } v, \\
1 - \sum_{y \neq v} P^{\text{MHRW}}_{v \rightarrow y} & \text{if } w = v, \\
0 & \text{otherwise.}
\end{cases}
\]

It can be shown that the resulting stationary distribution is \(\pi^{\text{MHRW}} = \frac{1}{|V|}\), which is exactly the uniform distribution we are looking for. \(P^{\text{MHRW}}_{v \rightarrow w}\) implies the following algorithm, which we refer to simply as MHRW in the rest of the paper:

\[
\text{\texttt{v \leftarrow initial node.}}
\]

\[
\text{\texttt{while stopping criterion not met do}}
\]

\[
\text{\texttt{Select node } w \text{ uniformly at random from neighbors of } v.}
\]

\[
\text{\texttt{Generate uniformly at random a number } 0 \leq p \leq 1.}
\]

\[
\text{\texttt{if } p \leq \frac{k_w}{k_v} \text{ then}}
\]

\[
\text{\texttt{\hspace{1cm} v \leftarrow w.}}
\]

\[
\text{\texttt{else}}
\]

\[
\text{\texttt{\hspace{1cm} Stay at } v}
\]

\[
\text{\texttt{end if}}
\]

\[
\text{\texttt{end while}}
\]

In every iteration of MHRW, at the current node \(v\) we randomly select a neighbor \(w\) and move there w.p. \(\min(1, \frac{k_w}{k_v})\). We always accept the move towards a node of smaller degree, and reject some of the moves towards higher degree nodes. This eliminates the bias towards high degree nodes.

C. Ground Truth: Uniform Sample of UserIDs (UNI)

Assessing the quality of any graph sampling method on an unknown graph, as it is the case when measuring real systems, is a challenging task. In order to have a “ground truth” to compare against, the performance of such methods is typically tested on artificial graphs. This has the disadvantage that one can never be sure that the results can be generalized to real networks that do not follow the simulated graph models and parameters.

Fortunately, Facebook was an exception during the time period we performed our measurements. We capitalized on a unique opportunity to obtain a uniform sample of Facebook users by generating uniformly random 32-bit userIDs, and by polling Facebook about their existence. If the userID exists, we keep it, otherwise we discard it. This simple method is a textbook technique known as rejection sampling \[57\] and in general it allows to sample from any distribution of interest, which in our case is the uniform. In particular, it guarantees to select uniformly random userIDs from the allocated Facebook users regardless of their actual distribution in the userID space, even when the userIDs are not allocated sequentially or evenly across the userID space. For completeness, we re-derive this property in Appendix \[VIII\]. We refer to this method as “UNI”, and use it as a ground-truth uniform sampler.

Although UNI sampling solves the problem of uniform node sampling in Facebook, crawling remains important. There are two necessary conditions for UNI to work. First, such an operation must be supported by the system. Facebook currently allows to verify the existence of an arbitrary userID and retrieve her list of friends; however, Facebook may remove this option in the future for privacy reasons. Second, the userID space must not be sparsely allocated for this operation to be efficient. During our data collection (April-May 2009) the number of Facebook users (~ 200 \(\times\) 10^6) was comparable to the size of the userID space (2^62 ~ 4.3 \(\times\) 10^3), resulting in about one user retrieved per 22 attempts on average. If the userID were 64 bit long or consisting of strings of arbitrary length, UNI would had been infeasible.

In summary, we were fortunate to be able to obtain a uniform sample of userIDs, which we then used as a baseline for comparison (our “ground truth”) to show that our results perfectly agree with it. However, crawling friendship relations is a fundamental primitive available in all OSNs and, we believe, the right building block for designing sampling techniques in OSNs, in the long run. For details on UNI, please see Appendix \[VIII\].

D. Convergence

1) Using Multiple Parallel Walks: Multiple parallel walks are used in the MCMC literature \[36\] to improve convergence. Intuitively, if we only have one walk, the walk may get trapped in cluster while exploring the graph, which may lead to erroneous diagnosis of convergence. Having multiple parallel walks reduces the probability of this happening and allows for more accurate convergence diagnostics. We note

\[\text{[To mention a few such cases in the same time frame: Orkut had a 64bit userID and hi5 used a concatenation of userID+Name. Interestingly, within days to weeks after our measurements were completed, Facebook changed its userID allocation space from 32 bit to 64 bit \[55\]. We speculate that the main reason for such a change was to allocate more userID space but we do not preclude security reasons behind this change i.e. to hinder efforts of data collection. Section \[VIII\] contains more information about userID space usage in Facebook In April 2009.]}\]
that the advantage of multiple random walks is achieved when there is no fixed budget in the number of samples that would lead to many short walks, which is true in our case. An additional advantage of multiple parallel walks, from an implementation point of view, is that it is amenable to parallel implementation from different machines or different threads in the same machine. Some coordination is then required to increase efficiency by not downloading information about nodes that have already been visited by independent walks.

We implemented each of the considered crawling algorithms with several parallel walks. Each walk starts from a different node in $V_0 \subset V$, $|V_0| \geq 1$ ($|V_0| = 28$ in our case) and proceeds independently of the others. The initial nodes $V_0$ are randomly chosen in different networks. For a fair comparison, we compare multiple MHRWs to multiple RWs and multiple BFSs, all starting from the same set of initial nodes $V_0$.

2) Detecting Convergence with Online Diagnostics: Valid inferences from MCMC are based on the assumption that the samples are derived from the equilibrium distribution, which is true asymptotically. In order to correctly diagnose when convergence to equilibrium occurs, we use standard diagnostic tests developed within the MCMC literature [59]. In particular, we would like to use diagnostic tests to answer at least the following questions:

- How many of the initial samples in each walk do we need to discard to lose dependence from the starting point (or burn-in)?
- How many samples do we need before we have collected a representative sample?

A standard approach is to run the sampling long enough and to discard a number of initial burn-in samples proactively. From a practical point of view, however, the burn-in comes at a cost. In the case of Facebook, it is the consumed bandwidth (in the order of gigabytes) and measurement time (days or weeks). It is therefore crucial to assess the convergence of our MCMC sampling, and to decide on appropriate settings of burn-in and total running time.

Given that during a crawl we do not know the target distribution, we can only estimate convergence from the statistical properties of the walks as they are collected. Here we present two standard convergence tests, widely accepted and well documented in the MCMC literature [59]. In particular, we would like to use diagnostic tests to answer at least the following questions:

- How many of the initial samples in each walk do we need to discard to lose dependence from the starting point (or burn-in)?
- How many samples do we need before we have collected a representative sample?

A standard approach is to run the sampling long enough and to discard a number of initial burn-in samples proactively. From a practical point of view, however, the burn-in comes at a cost. In the case of Facebook, it is the consumed bandwidth (in the order of gigabytes) and measurement time (days or weeks). It is therefore crucial to assess the convergence of our MCMC sampling, and to decide on appropriate settings of burn-in and total running time.

Given that during a crawl we do not know the target distribution, we can only estimate convergence from the statistical properties of the walks as they are collected. Here we present two standard convergence tests, widely accepted and well documented in the MCMC literature, Geweke [59] and Gelman-Rubin [60], described below. In theory, convergence diagnostics should be used to indicate lack of convergence. In practice however, negative results with regards to non-convergence provide a degree of confidence that the sample has converged. In Section V-A we apply these tests on several node properties, including the node degree, userID, network ID and membership in a specific network; please see Section V-A3 for details. Below, we briefly outline the rationale of these tests and we refer the interested reader to the references for more details.

**Geweke Diagnostic.** The Geweke diagnostic [59] detects the convergence of a single Markov chain. Let $X$ be a single sequence of samples of our metric of interest. Geweke considers two subsequences of $X$, its beginning $X_a$ (typically the first 10%), and its end $X_b$ (typically the last 50%). Based on $X_a$ and $X_b$, we compute the $z$-statistic:

$$z = \frac{E(X_a) - E(X_b)}{\sqrt{\text{Var}(X_a) + \text{Var}(X_b)}}$$

With increasing number of iterations, $X_a$ and $X_b$ move further apart, which limits the correlation between them. As they measure the same metric, they should be identically distributed when converged and, according to the law of large numbers, the $z$ values become normally distributed with mean 0 and variance 1. We can declare convergence when all values fall in the $[-1, 1]$ interval.

**Gelman-Rubin Diagnostic.** Monitoring one long sequence of nodes has some disadvantages. For example, if our chain stays long enough in some non-representative region of the parameter space, we might erroneously declare convergence. For this reason, Gelman and Rubin [60] proposed to monitor $m > 1$ sequences. Intuitively speaking, the Gelman-Rubin diagnostic compares the empirical distributions of individual chains with the empirical distribution of all sequences together: if these two are similar, we declare convergence. The test outputs a single value $R$ that is a function of means and variances of all chains. With time, $R$ approaches 1, and convergence is declared typically for values smaller than 1.02.

Finally, we note that even after convergence is diagnosed by the above tests and initial samples are discarded, strong correlation of consecutive samples in the walk may affect sequential analysis. This is typically addressed by thinning, i.e., keeping only one every $r$ samples. In our approach, instead of thinning, we do sub-sampling of nodes after burn-in, which has essentially the same effect.

**IV. DATA COLLECTION**

**A. User properties of interest**

Fig. 1 summarizes the information collected when visiting the “show friends” web page of a sampled user $u$, which we refer to as basic node information.

**Name and userID.** Each user is uniquely defined by her userID, which is a 32-bit number. Each user presumably provides her real name. The names do not have to be unique.

**Friends list.** A core idea in social networks is the possibility to declare friendship between users. In Facebook, friendship

---

\[ \text{Facebook changed to 64-bit user ID space after May 2009} \]
is always mutual and must be accepted by both sides. Thus the social network is undirected.

**Networks.** Facebook uses the notion of “networks” to organize its users. There are two types of networks. The first type is **regional** (geographical) networks. There are 507 predefined regional networks that correspond to cities, regions, and countries around the world. A user can freely join any regional network but can be a member of only one regional network at a time. Changes are allowed, but no more than twice every 6 months (April 2009). The second type of networks contain user affiliations with colleges, workplaces, and high schools and have stricter membership criteria: they require a valid email account from the corresponding domain i.e. to belong in the UC Irvine network you have to provide a “@uci.edu” email account. On the other hand, a user can belong to many such networks.

**Privacy settings** $Q_v$. Each user $u$ can restrict the amount of information revealed to any non-friend node $w$, as well as the possibility of interaction with $w$. These are captured by four basic binary privacy attributes, as described in Table I. We refer to the resulting 4-bit number as privacy settings $Q_v$ of node $v$. By default, Facebook sets $Q_v = 1111$ (allow all).

**Friends of $u$**. The “show friends” web page of user $u$ exposes network membership information and privacy settings for each listed friend. Therefore, we collect such information for all friends of $u$, at no additional cost.

**Profiles**. Much more information about a user can potentially be obtained by viewing her profile. Unless restricted by the user, the profile can be displayed by her friends and users from the same network. In this work, we do not collect any profile, even if it is open/publicly available. We study only the basic node information shown in Fig 1.

**Ego Networks**. The sample of nodes collected by our method enables us to study many features of FB users in a statistically unbiased manner. However, more elaborate topological measures, such as clustering coefficient and assortativity, cannot be estimated based purely on a single-node view. For this reason, after finishing the BFS, RW, MHRW crawls, we decided to also collect a sample of **extended ego nets** for a sub-sample of the MHRW dataset, which is representative of the whole Facebook population.

In the social network literature [22], the ego net consists of the ego, all alters (neighbors), and all edges among alters. Here we collect the egonet plus all observed edges of the alters, since there is no additional sampling cost. We call it the extended ego net and we illustrate the concept in Fig 2. We use the extended egonet sample differently from the Random Node Neighbor (RNN) sample presented in [23]. We are interested in estimating properties of the ego nodes only whereas RNN [23] looks at the induced subgraph of all sampled nodes and is interested in estimating properties of the ego and all alters. Therefore, the sample of extended egonets, that we collect in this work, is expected to capture very well community structure properties (i.e., clustering coefficient) of the whole Facebook population.

Collecting an egonet requires sampling 100 nodes per node (ego) on average, which is impossible to do for all visited nodes. For this reason, we collect the ego-nets only for ~37K nodes, randomly selected from all nodes in MHRW (considering all 28 walks, after the 6000 ‘burn-in’ period). This sub-sampling has the side advantage that it eliminates the correlation of consecutive nodes in the same crawl, as discussed in Section II-ID2.

### B. Crawling Process

In order to implement these ideas to sample real OSNs, we had to implement high-performance distributed crawlers that explore users in the social graph in a systematic and efficient way.

1) **Challenges**: There are several practical challenges we faced while crawling the social graph of OSNs.

A first challenge is the defense mechanisms used by OSNs against automated data mining. Most OSNs require a logged-in account to expose information and might monitor the download rate during crawling. Mislove et al. [12] reported rate limits per IP while crawling Orkut and in our crawls we experienced banned accounts, probably due to excessive traffic. As expected, data access is subject to the individual
user privacy settings but we observed that, for the same access level, API calls are usually more restrictive compared to HTML scraping. Thus, to crawl Facebook’s social graph we performed HTML scraping.

Part of the allure of online social networks is the interactive experience with rich content that they offer to users. Such interactive web sites are typically developed by using client-side application logic (usually written in Javascript). This technology, also referred to as AJAX programming, enables asynchronous loading of web content. Therefore, traditional web crawlers need modifications to be able to extract the full list of URLs when crawling OSNs. In our case, we needed to customize crawling for Facebook and extract relevant information with regular expressions.

Facebook is changing daily by (i) the registration of new users, (ii) addition of new relations (iii) removal of existing relations (iv) deletion of user accounts. Even though the growth of Facebook is considerable in absolute numbers (more than 400K new users per day), it is quite small as a percentage of the total population (less than 0.25% per day), during the duration of our crawls. We validated this assumption based on advertised statistics on the growth of Facebook as well as based on our own measurements. For details the reader is referred to Appendix B. However, the longer the data collection time, the larger the effect of user growth in the collected sample. To address this challenge we built high performance crawlers that perform distributed data fetching and take advantage of the specifics of each crawling methodology. The ability to collect data in the order of hours to days allows us to consider the Facebook friendship graph as static in the timescale of our crawls.

2) Implementation: Fig. 3 depicts an overview of our distributed crawling process. Below we describe the system-level implementation.

First, we use a large number of machines with limited memory (100 Mbytes-1GBytes RAM) and disk space (up to 5Gbytes), to parallelize our crawling and shorten the data collection time. We have up to three layers of parallelism in each machine. Each crawling machine runs one or more crawling processes. Each crawling process shares one user account between multiple crawling threads within it. Each crawling thread fetches data asynchronously where possible. The determination of the number of processes per machine and threads per process depends on the defenses that the OSN has implemented and the crawling technique used. The algorithm (e.g. BFS, MHRW, etc) is implemented inside each crawling thread. The data, obtained from the OSN either through API calls or data scraping, is extracted and dumped in a text format for offline processing.

Second, we use one machine as coordinator server that supports the following operations. (i) It contains a mechanism to rate-limit the number of connections or amount of bandwidth over the whole cluster. We can use it so as not to cause a high load to the crawled OSN. (ii) It keeps track of already fetched users to avoid fetching duplicate data. As a result, the crawling process continues in a faster pace, since each request to the OSN returns new information. Additionally, we can increase on-demand the number of crawling machines without redundant data fetching. (iii) It maintain a data structure, which stores the crawl frontier i.e., a queue for BFS. (iv) It has the ability to cache data for re-use by crawling machines i.e., for random walks which perform sampling with replacement. Our ability to parallelize the crawling process depends on the selected methodology. For example, for BFS, we use in the coordinator a queue, that maintains the crawl frontier, and a hash table, that makes sure we do not re-fetch the same data; for RW, we use in the coordinator a size constrained hash table, that caches data that might be later re-used.

Third, we use a user account server to handle the authentication (login account or API account) per crawling process. When a crawling process is initiated, it requests an unused account from the user account server. The crawling process is activated only if a valid account is available. Furthermore, a crawling process has the ability to detect an invalidated account and request from the user account server to erase it and to return a new valid account. The user account server contains the accounts available for use by the crawling processes. The assumption is that the crawling administrator will populate the user account server with valid accounts at the beginning of the crawl and maintain a number of valid accounts comparable to the number of crawling processes during the crawl.

3) Invalid users: There are two types of users that we declare as invalid. First, if a user $u$ decides to hide her friends and to set the privacy settings to $Q_u = \ast\ast\ast\ast\ast\ast$, the crawl cannot continue. We address this problem by backtracking to the previous node and continuing the crawl from there, as if $u$ was never selected. Second, there exist nodes with degree $k_v = 0$; these are not reachable by any crawls, but we stumble upon them during the UNI sampling of the userID space. Discarding both types of nodes is consistent with our assumptions (A2, A3), where we already declared that we exclude such nodes (either not publicly available (A2) or isolated (A3)) from the graph we want to sample.

4) Execution of crawls: In Section III-D1 we discussed the advantages of using multiple parallel walks both in terms of convergence and implementation. We ran $|V_0| = 28$ different independent crawls for each crawling methodology, namely...
MHRW, BFS and RW, all seeded at the same initial, randomly selected nodes $V_0$. We let each independent crawl continue until we determine convergence per walk and per average crawl for the random walk methods. Eventually we collected exactly 81K samples for each independent crawl. It is a value much larger than required for convergence but it allowed us to perform sub-sampling and to collect extended egonets. We count towards this value all repetitions, such as the self-transitions of MHRW, and returning to an already visited state (RW and MHRW). In addition to the $28 \times 3$ crawls (BFS, RW and MHRW), we ran the UNI sampling until we collected 984K valid users, which is comparable to the 957K unique users collected with MHRW.

C. Description of Datasets

Information about the datasets we collected is summarized in Tables II and III. This information refers to all sampled nodes, before discarding any burn-in. The MHRW crawl contains 957K unique nodes, which is less than the $28 \times 81K = 2.26M$ iterations in all 28 random walks; this is because MHRW may reject a node transition and instead sample the same node during a walk. The number of rejected nodes in all MHRW walks, without repetitions, adds up to 645K nodes. In the BFS crawl, we observe that the overlap of nodes between the 28 different BFS instances is very small: 97% of the nodes are unique, which also confirms that the random seeding chose different areas of Facebook. In the RW crawl, 97% of the nodes are unique.

Table III shows that the percentage of common users between the MHRW, RW, BFS and UNI datasets is very small, as expected. The largest observed, but still objectively small, overlap is between RW and BFS and is probably due to the common starting points selected.

To collect the UNI dataset, we checked $\sim 18.5M$ user IDs picked uniformly at random from [1, 232]. Out of them, only 1,216K users existed. Among them, 228K users had zero friends; we discarded these isolated users to be consistent with our problem statement. This results in a set of 984K valid users with at least one friend each. Considering that the percentage of zero degree nodes is unusually high, we manually confirmed that 200 of the discarded users have indeed zero friends.

To analyze topological characteristics of the Facebook population, we collected $\sim 37K$ egonets that contain basic node information (see Fig 1) for $\sim 5.8M$ unique neighbors. Table I contains a summary of the egonet dataset, including properties that we analyze in Section VI.

Overall, from (i) the multiple crawls, namely BFS, RW, MHRW, UNI and (ii) the ego networks of a sub-sample of the Metropolis walk, we collected 11.6 million unique nodes with basic node information. This results in $\sim 172$ million unique users (including the sampled nodes and the neighbors in their egonets) for which we have basic privacy and network membership information becomes immense. This is a very large sample by itself, especially given that Facebook had reported having close to 200 million active users during the time of these measurements. We use this information in Section VI to analyze privacy related properties.

V. Evaluation of Sampling Techniques

In this section, we evaluate all candidate crawling techniques (namely BFS, RW and RWRW, MHRW), in terms of their efficiency (convergence) and quality (estimation bias). In Section V-A we study the convergence of the random walk methods, with respect to several properties of interest. We find a burn-in period of 6K samples, which we exclude from each independent crawl. The remaining 75K x 28 sampled nodes is our main sample dataset; for a fair comparison we also exclude the same number of burn-in samples from all datasets. In Section V-B we examine the quality of the estimation based on each sample. In Section V-C we summarize our findings and provide practical recommendations.

A. Convergence analysis

There are several crucial parameters that affect the convergence of a MCMC. In this section, we study these parameters by (i) applying formal convergence tests and (ii) using simple, yet insightful, visual inspection of the related traces and histograms.

1) How to count: Counting samples in BFS is trivial since nodes are visited at most once. However, in the random walks, nodes can be revisited and repetitions must be included in the sample in order to ensure the desired statistical properties. For RW, the same user cannot be immediately visited twice, but non-consecutive repetitions are possible. In practice, that happens infrequently in the RW sample (as can be seen from the number of unique nodes in Table III). On the other hand, MHRW repeatedly samples some (typically low degree) nodes, a property which is intrinsic to its operation. For instance, if some node $v_i$ has only one neighbor $v_{hi}$, then the chain stays at (repeatedly samples) $v_i$ for an average of $k_{v_i}$ iterations ($k_v$ is the degree of node $v$). Where $k_{v_i}$ is large (e.g., $O(10^2)$ or more), the number of repetitions may be locally large. While counter intuitive, this behavior is essential for convergence to the uniform distribution. In our MHRW sample, roughly 45% of the proposed moves are accepted. As a result, a typical MHRW visits fewer unique nodes than a RW or BFS sequence of the same length. This raises the question: what is a fair way to compare the results of MHRW with RW and BFS? Since queries are only made for new nodes, if $k_{v_i} = 1$ and MHRW stays at $v_i$ for some $t > 1$ iterations when crawling an OSN, the bandwidth consumed is equal in cost to one iteration (assuming that we cached the visited neighbor of $v_i$). This suggests that an appropriate practical comparison should be based not on the total number of iterations, but rather the number of visited unique nodes. In our subsequent comparisons, we will denote RW and MHRW indices as “RW-Fair” and “MHRW-Fair” when we compare using the number of visited unique nodes, as this represents the methods in terms of equivalent bandwidth costs.

2) Convergence Tests:

I. Burn-in. For the random walk based methods, a decision we have to make is about the number of samples to discard to lose dependence from the initial seed point. Since there is a cost for every user we sample, we would like to choose this value using formal convergence diagnostics so as not to
waste resources. Here, we apply the convergence diagnostics presented in Section III-D2 to several properties of the sampled nodes and choose as burn-in the maximum period from all tests.

The Geweke diagnostic is applied in each of the 28 walks separately and compares the difference between the first 10% and the last 50% of the walk samples. It declares convergence when all 28 values fall in the $[-1, 1]$ interval. Fig. 4 presents the results of the Geweke diagnostic for the user properties of node degree and regional network membership. We start at 50 iterations and plot 150 points logarithmically spaced. We observe that after approximately 500–2000 iterations we have a z-score strictly between $[-1, 1]$. We also see that RWRW and MHRW show similar results in regard to the Geweke diagnostic.

The Gelman-Rubin R diagnostic analyzes all the 28 walks at once by summarizing the difference of the between-walk variances and within-walk variances. In Fig 5 we plot the R score for the following metrics (i) number of friends (or node degree) (ii) networkID (or regional network) (iii) privacy settings $Q_v$ (iv) membership in specific regional networks, namely Australia, New York and Colombia. The last user property is defined as follows: if the user in iteration $i$ is a member of network $x$ then the metric is set to 1, otherwise it is set to 0. We can see that the R score varies a lot in the initial hundred to thousand iterations for all properties. To pick an example, we observe a spike between iterations 1,000 and 2,000 in the MHRW crawl for the New York membership. This is most likely the result of certain walks getting trapped within the New York network which is particularly large. Eventually, after 3000 iterations all the R scores for the properties of interest drop below 1.02, the typical target value used for convergence indicator.

We declare convergence when all tests have detected it. The Gelman-Rubin test is the last one at 3K nodes. To be even safer, in each independent walk we conservatively discard 6K nodes, out of 81K nodes total. In the remainder of the evaluation, we work only with the remaining 75K nodes per independent chain for RW, RWRW and MHRW.

**II. Total Running Time.** Another decision we have to make is about the number of iterations for which we run the random walks, or the walk length, excluding the burn-in samples. This length should be appropriately long to ensure that we are at equilibrium. Here, we utilize multiple ways to analyze the collected samples, so as to increase our confidence that

| (left:) Collected datasets by different algorithms during April-May 2009. The crawling algorithms (MHRW, RW and BFS) consist of 28 parallel walks each, with the same 28 starting points. UNI is the uniform sample of user IDs. (right:) The overlap between different datasets is small. | (right:) Collected datasets by different algorithms during April-May 2009. The crawling algorithms (MHRW, RW and BFS) consist of 28 parallel walks each, with the same 28 starting points. UNI is the uniform sample of user IDs. (right:) The overlap between different datasets is small. |
|---|---|---|---|---|
| Total number of valid users | MHRW | RW | BFS | UNI |
| Total number of unique users | 28×81K | 28×81K | 28×81K | 884K |
| Total number of unique neighbors | 957K | 2.19M | 2.20M | 984K |
| Crawling period | 04/18-04/23 | 05/03-05/08 | 04/30-05/03 | 04/22-04/30 |
| Avg Degree | 40 | 234 | 208 | 38 |
| Median Degree | 0.5 | 318 | 123 | 0.41 |
| Median Degree | 95.2 | 120.1M | 323 | 15.1K |
| Number of overlapping users | MHRW ⊖ RW | 16.2K | MHRW ⊖ BFS | 15.1K |
| Number of overlapping users | MHRW ⊖ Uniform | 58.4M | RW ⊖ BFS | 64.2K |
| Number of overlapping users | RW ⊖ Uniform | 2.20M | BFS ⊖ Uniform | 15.1K |

![Table III](image-url)
the collected samples are appropriate for further statistical analysis.

First, we apply formal convergence diagnostics that allow us to assess convergence online by indicating approximate equilibrium. Fig 6 shows the Geweke z-score for the number of friends (top) and the Gelman-Rubin R score for five different properties (bottom). These results are obtained after discarding the burn-in samples (0K..6K). They show that convergence is attained with at least 3k samples per walk, similar to the section in which we determined the burn-in. This is an indication that the Facebook social graph is well connected and our random walks achieved good mixing with our initial selection of random seeds.

Second, we perform visual inspection to check the convergence state of our sampling by plotting for each walk the running mean of a user property against the iteration number. This is a practical way to detect steady state, extensively used in the case of crawling OSNs, the main bottleneck is the time and bandwidth necessary to perform a single transition, rather than storage and post-processing of the extracted information. Therefore we did not apply thinning to our basic crawls.

However, we applied another idea (sub-sampling), that has

\[ \frac{1}{K} \sum_{v \in V} \hat{e}(v) \]
a similar effect with thinning, when collecting the second part of our data - the egonets. Indeed, in order to collect the information on a single egonet, our crawler had to visit the user and all its friends, an average $\sim 100$ nodes. Due to bandwidth and time constraints, we could fetch only 37K egonets. In order to avoid correlations between consecutive egonets, we collected a random sub-sample of the MHRW (post burn-in) sample, which essentially introduced spacing among sub-sampled nodes.

### IV. Comparison to Ground Truth

Finally, we compare the random walk techniques (RWRW, MHRW) in estimating the degree distribution of Facebook, in terms of the Kullback-Leibler (KL) divergence. We observe that (i) RWRW converges faster than MHRW and approximates UNI better at the end; and (ii) RWRW-Fair is also more efficient than MHRW-Fair. The “Fair” versions of the algorithms count the real bandwidth cost of contacting a previously unseen neighbor, which is also more efficient than MHRW-Fair. The “Fair” versions of the algorithms are approximations of UNI better at the end; and (ii) RWRW-Fair is also more efficient than MHRW-Fair. The “Fair” versions of the algorithms count the real bandwidth cost of contacting a previously unseen neighbor, and also more efficient than MHRW-Fair. The “Fair” versions of the algorithms are more efficient than MHRW-Fair.

Let us focus on two of these metrics of interest, namely node degree and sizes of geographical network and study their convergence in more detail. The results for both metrics and all three methods are shown in Fig. 10. We expected node degrees to not depend strongly on geography, while the relative size of geographical networks to strongly depend on geography. This implies that (i) the degree distribution will converge fast to a good uniform sample even if the walk has poor mixing and stays in the same region for a long time; (ii) a walk that mixes poorly will take long time to barely reach the networks of interests, not to mention producing a reliable network size estimate. In the latter case, MHRW will need a large number of iterations before collecting a representative sample.

The results presented in Fig. 10 confirm our expectation. MHRW performs much better when estimating the probability of a node having a given degree, than the probability of a node belonging to a specific regional network. For example, one MHRW crawl overestimates the size of “New York, NY” by roughly 100%. The probability that a perfect uniform sampling makes such an error (or larger) is $\sum_{i=0}^{\infty} \binom{n}{i} p^i (1-p)^{n-i} \approx 4.3 \times 10^{-13}$, where we took $i_0 = 1$, $n = 81K$, $p = 0.006$. Even given such single-walk deviations, however, the multiple-walk average for the MHRW crawl provides an excellent estimate of the true population size.

### B. Unbiased Estimation

This section presents the main results of this chapter. First, the MHRW and RWRW methods perform very well: they estimate two distributions of interest (node degree, regional network size) essentially identically to the UNI sampler. Second, the baseline algorithms (BFS and RW) deviate substantively from the truth and lead to misleading estimates.

#### 1) Node degree distribution

In Fig. 11, we present the degree distributions estimated by MHRW, RWRW, RW, and BFS. The average MHRW crawl’s pdf, shown in Fig. 11(a) is virtually identical to UNI. Moreover, the degree distribution found by each of the 28 chains separately are almost identical. In contrast, RW and BFS shown in Fig. 11(c) and (d) introduce MCMC has converged at least with respect to it. The distribution of the node degree is also typically heavy tailed, and thus is slow to converge.

- We also used several additional metrics (e.g. network ID, user ID and membership to specific networks), which are uncorrelated to the node degree and to each other, and thus provide additional assurance for convergence.

#### 3) The choice of metric matters

MCMC is typically used to estimate some user property/metric, i.e., a function of the underlying random variable. The choice of this metric can greatly affect the convergence time. We chose the metrics in the diagnostics of the previous section, guided by the following principles:

- We chose the node degree because it is one of the metrics we want to estimate; therefore we need to ensure that the

![Fig. 7. Average node degree $\overline{k}_v$ observed by each crawl, as a function of the number of iterations (or running mean).](image1)

![Fig. 9. Efficiency of the random walk techniques (RWRW, MHRW) in estimating the degree distribution of Facebook, in terms of the Kullback-Leibler (KL) divergence. We observe that (i) RWRW converges faster than MHRW and approximates UNI better at the end; and (ii) RWRW-Fair is also more efficient than MHRW-Fair. The “Fair” versions of the algorithms count the real bandwidth cost of contacting a previously unseen neighbor, either for sampling (in RW) or to learn its degree (in MHRW), based on our measurements.](image2)
As a side observation we can also see that the true degree distribution, thus leading to wrong information. Re-weighting the simple RW corrects for the bias and results to RWRW, which performs almost identical to UNI, as shown in Fig 11b. As a side observation we can also see that the true degree distribution clearly does not follow a power-law.

2) Regional networks: Let us now consider a geography-dependent sensitive metric, i.e., the relative size of regional networks. The results are presented in Fig. 10 (right). BFS performs very poorly, which is expected due to its local coverage. RW also produces biased results, possibly because of a slight positive correlation that we observed between network size and average node degree. In contrast, MHRW and RWRW perform very well albeit with higher variance, as already discussed in Section V-A3.

3) The userID space: Finally, we look at the distribution of a property that is completely uncorrelated from the topology of Facebook, namely the user ID. When a new user joins Facebook, it is automatically assigned a 32-bit number, called userID. It happens before the user specifies its profile, joins networks or adds friends, and therefore one could expect no correlations between userID and these features. In other words, the degree bias of BFS and RW should not affect the usage of userID space. Therefore, at first sight we were very surprised to find big differences in the usage of userID space discovered by BFS, RW and MHRW. We present the results in Fig 12.

Note that the userID space is not covered uniformly, probably for some historical reasons. BFS and RW are clearly

![Figure 10](image1.png)

![Figure 11](image2.png)
shifted towards lower userIDs. The origin of this shift is probably historical. The sharp steps at $2^{20} \approx 0.5e9$ and at $2^{30} \approx 1.0e9$ suggest that Facebook was first using only 29 bit of userIDs, then 30, and now 31. As a result, users that joined earlier have the smaller userIDs. At the same time, older users should have higher degrees on average. If our reasoning is correct, userIDs should be negatively correlated with node degrees. This is indeed the case, as we show in the inset of Fig 12. This, together with the degree bias of BFS and RW, explains the shifts of userIDs distributions observed in the main plot in Fig 12. In contrast to BFS and RW, MHRW performed extremely well with respect to the userID metric.

C. Findings and Practical Recommendations

1) Choosing between methods: First and most important, the above comparison demonstrates that both MHRW and RWRW succeed in estimating several Facebook properties of interest virtually identically to UNI. In contrast, commonly used baseline methods (BFS and simple RW) fail, i.e., deviate significantly from the truth and lead to substantively erroneous estimates. Moreover, the bias of BFS and RW shows up not only when estimating directly node degrees (which was expected), but also when we consider other metrics seemingly uncorrelated metrics (such as the size of regional network), which end up being correlated to node degree. This makes the case for moving from “1st generation” traversal methods such as BFS, which have been predominantly used in the measurements community so far [9], [12], [13], to more principled, “2nd generation”, sampling techniques whose bias can be analyzed and/or corrected for. The random walks considered in this paper, RW, RWRW and MHRW, are well-known in the field of Monte Carlo Markov Chains (MCMC). We apply and adapt these methods to Facebook, for the first time, and we demonstrate that, when appropriately used, they perform remarkably well on real-world OSNs.

2) Adding convergence diagnostics and parallel crawls: A key ingredient of our implementation - not previously employed in network sampling - was the use of formal online convergence diagnostic tests. We tested these on several metrics of interest within and across chains, showing that convergence was obtained within a reasonable number of iterations. We believe that such tests can and should be used in field implementations of walk-based sampling methods to ensure that samples are adequate for subsequent analysis. Another key ingredient of our implementation, which we recommend, was the use of parallel crawlers/chains (started from several random independent starting points, unlike [17], [19] who use a single starting point), which both improved convergence and decreased the duration of the crawls.

3) MHRW vs. RWRW: Both MHRW and RWRW achieved a uniform sample. When comparing the two, RWRW is slightly more efficient, i.e., needed less samples for the same accuracy. This is consistent with the findings in [17], [19]. This is partly due to the fact that MHRW requires a large number of rejections during the initial sampling process; and partly due to slower mixing, in practice, as it avoids high degree nodes. In this section, we present an empirical comparison based on the Facebook experiments. In Appendix C, we provide a more in-depth comparison via analysis and simulation.

However, when choosing between the two methods there are additional trade-offs to consider. First, MHRW yields an asymptotically uniform sample, which requires no additional processing for subsequent analysis. By contrast, RWRW samples are heavily biased towards high-degree nodes, and require use of appropriate re-weighting procedures to generate correct results. For the creation of large data sets intended for general distribution (as in the case of our Facebook sample), this “ready-to-use” aspect of MHRW has obvious merit. A second advantage of MHRW is the ease of online testing for convergence to the desired target (uniform) distribution. In contrast, in RWRW, we test for convergence on a different distribution and then re-weight, which can introduce distortion. Finally, simple re-weighting is difficult or impossible to apply in the context of many purely data-analytic procedures such as multidimensional scaling or hierarchical clustering. Simulated Importance Resampling [63] provides a useful alternative for RWRW samples, but suffers from well-known problems of asymptotic bias (see [64] for a discussion. This is of less concern for applications such as moment estimation, for which re-weighting is both simple and effective.

Ultimately, the choice of RWRW versus MHRW is thus a trade-off between efficiency during the initial sampling process (which favors RWRW in all practical cases) and simplicity/versatility of use for the resulting data set (which favors MHRW). For our present purposes, these trade-offs
favor MHRW, and we employ it here for producing a uniform ready-to-use sample of users.

VI. FACEBOOK CHARACTERIZATION

In this section, we use the unbiased sample of 1M nodes, collected through MHRW, and the sub-sample of 37K extended egonets to study some features of Facebook. In contrast to previous work, which focused on particular regions [11], [44] or used larger but potentially biased samples [12], [13], our results are representative of the entire Facebook graph.

A. Topological Characteristics

We first focus on purely topological aspects of the graph of Facebook.

1) Degree distribution: Node degree is just one example of a node (user) property, which also happens to be an important topological characteristic. In Fig. 11(a,b), we present the node degree distribution of Facebook. Differently from previous studies of crawled datasets in online social networks in [9], [12], [13], we observe that node degree is not a power law. Instead, we can identify two regimes, roughly $1 \leq k < 300$ and $300 \leq k \leq 5000$, each of which can be approximated by a power-law with exponents $\alpha_{k<300} = 1.32$ and $\alpha_{k\geq300} = 3.38$, respectively. We note, however, that the regime $300 \leq k \leq 5000$ covers only slightly more than one decade. This behavior is suggestive of multistage “vetting” models of network formation.

2) Assortativity: Depending on the type of complex network, nodes may tend to connect to similar or different nodes. For example, in many social networks high degree nodes tend to connect to other high degree nodes [65]. Such networks are called assortative. In Fig. 13, we plot the node degree vs. the degrees of its neighbors. We observe a positive correlation, which implies assortative mixing and is in agreement with previous studies of similar social networks. We can also summarize this plot by calculating the Pearson correlation coefficient, or assortativity coefficient which is $r = 0.233$. This value is higher than $r' = 0.17$ reported in [13]. A possible explanation is that the Region-Constrained BFS used in [13] stops at regional network boundaries and thus misses many connections to, typically high-degree, nodes outside the network.

3) Clustering coefficient: In social networks, it is likely that two friends of a user are also friends of each other. The intensity of this phenomenon can be captured by the clustering coefficient $C_v$ of a node $v$, defined as the relative number of connections between the nearest neighbors of $v$. The clustering coefficient of a network is just an average value $C$ over all nodes. We find the average clustering coefficient of Facebook to be $C = 0.16$, similar to that reported in [13]. Since the clustering coefficient tends to depend strongly on the node’s degree $k_v$, we looked at $C_v$ as a function of $k_v$. Fig. 14 shows a larger range in the degree-dependent clustering coefficient ([0.05, 0.35]) than what was found in [13] ([0.05, 0.18]).

B. Privacy awareness

Our crawls collected, among other user properties, the privacy settings $Q_v$ for each node $v$. $Q_v$ consists of four bits, each corresponding to one privacy attribute. By default, Facebook sets these attributes to ‘allow’, i.e., $Q_v = 1111$ for a new node $v$. We call users who change these default settings as ‘privacy aware’ users, and we denote by $PA$ the level of privacy awareness of a user $v$, i.e., privacy aware users have $PA = P(Q_v \neq 1111)$.

We studied the privacy awareness in Facebook and we report some of our findings. First, we present the distribution of privacy settings among Facebook users in Fig. 15 which shows that about 84% of users leave the settings unchanged, i.e., $P(Q_v = 1111) \simeq 0.84$. The remaining 16% of users modified the default settings, yielding $PA = 0.16$ across the entire Facebook. The two most popular modifications are

Fig. 13. Assortativity - correlation between degrees of neighboring nodes. The dots represent the degree-degree pairs (randomly subsampled for visualization only). The line uses log-binning and takes the average degree of all nodes that fall in a corresponding bin.

Fig. 14. Clustering coefficient of Facebook users as function of their degree.

Fig. 15. The distribution of the privacy settings among ~172M Facebook users. Value $Q_v = 1111$ corresponds to default settings (privacy not restricted) and covers 84% of all users.
TABLE IV
REGIONAL NETWORKS WITH RESPECT TO THEIR PRIVACY AWARENESS

| Network | PA | Network | PA |
|---------|----|---------|----|
| Iceland | 0.08 | ... | ... |
| Denmark | 0.11 | ... | ... |
| Provo, UT | 0.23 | Hamilton, ON |
| Ogden, UT | 0.23 | Calgary, AB |
| Slovenia | 0.23 | Iran |
| Plymouth | 0.23 | India |
| Eastern Idaho, ID | 0.23 | Egypt |
| Indonesia | 0.24 | United Arab Emirates |
| Western Colorado, CO | 0.24 | Palestine |
| Quebec City, QC | 0.25 | Vancouver, BC |
| Salt Lake City, UT | 0.26 | Lebanon |
| Ontario, CO | 0.27 | Turkey |
| Lancaster, PA | 0.27 | Toronto, ON |
| Boise, ID | 0.28 | Kuwait |
| Portsmouth | 0.29 | Jordan |
| ... | ... | ... | 0.30 | Saudi Arabia |

Q_v = 1011 (“hide my photo”) and Q_v = 1101 (“hide my friends”), each applied by about 7% of users. Second, the privacy awareness PA of Facebook users depends on many factors, such as the geographical location, node degree or the privacy awareness of friends. In Table IV we classify the regional networks with respect to PA of their members. In Fig. 16 we show the effect of node degree on the privacy settings of a user. We found that low degree nodes tend to be very concerned about privacy, whereas high degree nodes hardly ever bother to modify the default settings. This clear trend makes sense in practice: to protect her privacy, a privacy concerned user would carefully select her Facebook friends, e.g., by avoiding linking to strangers. At the other extreme, there are users who prefer to have as many ‘friends’ as possible, which is much easier with unrestricted privacy attributes. Finally, we found that the privacy awareness of a user is positively correlated with the privacy awareness of her friends. We observe a clear positive correlation in Fig. 17.

C. Estimation of the Country-to-Country friendship graph

Here we demonstrate an application of our user sampling techniques that allows us to obtain a coarser granularity view of the Facebook social graph at the regional network level. One of the user properties we collected was the affiliation of every user with regional networks. For the ease to illustration, we aggregate cities, regions, and states to countries and we estimate the country-to-country topology G^C = (C, E^C), based on (i) the country affiliation of individual users in the sample and (ii) their social links to users with other country affiliations. We create a graph G^C = (C, E^C), whose nodes C are countries and whose edges E^C are weighted connections between countries, based on the friendship relations on the -fine granularity - social graph. More specifically, the weight w(A, B) of edge (A, B) represents the probability that two nodes, selected uniformly at random from countries A, B ∈ C, are connected in G^C.

The country-to-country topology can be the first step towards answering questions about how the formation of social links between users is affected by physical distance, language, political situation, history etc. In Fig 18 we present a small part of the estimated country-to-country graph, which contains only the 300 strongest edges between the largest 50 countries in Facebook. The edge weight between two countries is indicated by the line thickness and color intensity. Some interesting observations include the strong cliques formed between Middle Eastern countries and South-East-Asian countries, the many diverse but strong relations of Australia and New Zealand, and the inwardness of the United States.

VII. CONCLUSION

In this paper, we developed a framework for unbiased sampling of users in an OSN by crawling the social graph and we provided recommendations for its implementation in practice. We made the following three contributions. We compared several candidate techniques in terms of bias (BFS and RW where significantly biased while MHRW and RWRW provided unbiased samples) and efficiency (through experimentation and analysis, we found RWRW to be the most efficient in practice, while MHRW has the advantage of providing a ready-to-use sample). We introduced the use of formal
online diagnostic tests for declaring burn-in and convergence. This is in addition to the offline comparison of all crawling methods against ground truth (uniform sampling of user IDs via rejection sampling). Finally, we applied these methods to obtain the first unbiased sample of Facebook users, and we used it to characterize several key user and topological properties of Facebook. The datasets are accessible and have been downloaded approximately 200 times over the last 6 months. We also provided guidelines for implementing high performance crawlers for sampling OSNs.

REFERENCES

[1] “Nielsen statistics,” http://blog.nielsen.com/nielsenwire/online_mobile/social-mobile-appliances-for-facebook/June 2010.

[2] “Alexa traffic statistics for facebook,” http://www.alexaw.com/siteinfo/facebook.com June 2010.

[3] J. Pujol, V. Erramilli, G. Sigano, X. Yang, N. Laoutaris, P. Chhabra, and P. Rodriguez, “The little engine (s) that could: Scaling online social networks,” ACM SIGCOMM Computer Communication Review, vol. 40, no. 4, pp. 375–386, 2010.

[4] S. Agarwal, “Social networks as Internet barometers for optimizing content delivery networks,” in Advanced Networks and Telecommunication Systems (ANTS), 2009 IEEE 3rd International Symposium on. IEEE, 2010, pp. 1–3.

[5] A. Nazir, S. Raza, D. Gupta, C. Chuah, and B. Krishnamurthy, “Network level footprints of facebook applications,” in Proceedings of the 9th ACM SIGCOMM conference on Internet measurement conference. ACM, 2009, pp. 63–75.

[6] A. Mislove, A. Post, P. Druschel, and K. Gummadi, “Ostra: Leveraging trust to thwart unwanted communication,” in Proceedings of the 5th USENIX Symposium on Networked Systems Design and Implementation. USENIX Association, 2008, pp. 15–30.

[7] M. Sirivianos, K. Kim, and X. Yang, “Introducing Social Trust to Collaborative Spam Mitigation,” in INFOCOM, 2011.

[8] ——, “FaceTrust: Assessing the Credibility of Online Personas via Social Networks,” in Proceedings of the 4th USENIX conference on Hot topics in security. USENIX Association, 2009, p. 2.

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

[10] J. Leskovec, L. Backstrom, R. Kumar, and A. Tomkins, “Microscopic evolution of social networks,” in Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2008, pp. 462–470.

[11] K. Lewis, J. Kaufman, M. Gonzalez, A. Wimmer, and N. Christakis, “Tastes, ties, and time: A new social network dataset using Facebook.com,” Social Networks, 2008.

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

[13] C. Wilson, B. Boe, A. Sala, K. P. Putsawamy, and B. Y. Zhao, “User interactions in social networks and their implications,” in Proc. of EuroSys, 2009.

[14] M. Kurant, A. Markopoulou, and P. Thiran, “On the bias of breadth first search (bfs) and of other graph sampling techniques,” in Proc. of ITC ’22, 2010.

[15] ——, “Towards unbiased bfs sampling,” in submission in JSAC on Measurement of Internet Topologies, 2010.

[16] M. Salganik and D. Heckathorn, “Sampling and estimation in hidden populations using respondent-driven sampling,” Sociological Methods & Research, vol. 34, p. 192349, 2004.

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

[18] D. Stutzbach, R. Rejaie, N. Duffield, S. Sen, and W. Willinger, “On unbiased sampling for unstructured peer-to-peer networks,” in Proc. of IMC, 2006.

[19] A. Rasti, M. Torkjazi, R. Rejaie, N. Duffield, W. Willinger, and D. Stutzbach, “Evaluating sampling techniques for large dynamic graphs,” University of Oregon, Tech. Rep., Sept 2008.

[20] B. Krishnamurthy, P. Gill, and M. Arltt, “A few chirps about twitter,” in Proceedings of the 1st ACM SIGCOMM Workshop on Social Networks (WOSN’08), 2008.

[21] “Unbiased samples of facebook users: Publicly available datasets,” http://odyseas.calit2.uci.edu/research/son.html, 2009.

[22] S. Wasserman and K. Faust, Social Network Analysis: Methods and Applications. Cambridge University Press, 1994.

[23] A. Mislove, H. S. Koppula, K. P. Gummadi, P. Druschel, and B. Bhattacharjee, “Growth of the flickr social network,” in Proceedings of the 1st ACM SIGCOMM Workshop on Social Networks (WOSN’08), August 2008.

[24] B. Viswanath, A. Mislove, M. Cha, and K. P. Gummadi, “On the Evolution of User Interaction in Facebook,” in Proceedings of the 2nd ACM SIGCOMM Workshop on Social Networks (WOSN’09), Barcelona, Spain, August 2009.

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

[26] L. Becchetti, C. Castillo, D. Donato, and A. Fazzzone, “A comparison of sampling techniques for web graph characterization,” in LinkKDD, 2006.

[27] D. Achlioptas, A. Chawla, D. Kempe, and C. Moore, “On the bias of traceroute sampling: or, power-law degree distributions in regular graphs,” in STOC, 2005.

[28] S. Ye, J. Lang, and F. Wu, “Crawling online social graphs,” in Web Conference (APWEB), 2010 12th International Asia-Pacific. IEEE, 2010, pp. 236–242.

[29] L. Lovasz, “Random walks on graphs. a survey,” in Combinatorics, 1993.

[30] M. R. Henzinger, A. Heydon, M. Mitzenmacher, and M. Najork, “On near-uniform url sampling,” in Proc. of WWW, 2000.

[31] E. Baykan, M. R. Henzinger, S. F. Keller, S. D. Castelberg, and M. Kimzer, “A comparison of techniques for sampling web pages,” in STACS, 2009.

[32] C. Kiantisidis, M. Mihail, and A. Saberi, “Random walks in peer-to-peer networks,” in Proc. of INFOCOM, 2004.

[33] J. Leskovec and C. Faloutsos, “Sampling from large graphs,” in Proc. of ACM SIGKDD, 2006.

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

[35] N. Metropolis, M. Rosenblut, A. Rosenbluth, A. Teller, and E. Teller, “Equation of state calculation by fast computing machines,” J. Chem. Physics, vol. 21, pp. 1087–1092, 1953.

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

[37] M. Latty and C. Magnien, “Complex network measurements: Estimating the relevance of observed properties,” in INFOCOM 2008. The 27th Conference on Computer Communications. IEEE, IEEE, 2008, pp. 1660–1668.

[38] R. Kumar, J. Novak, and A. Tomkins, “Structure and evolution of online social networks,” in Proc. of ACM SIGKDD, 2006.

[39] L. Backstrom, D. Huttenlocher, J. Kleinberg, and X. Lan, “Group formation in large social networks: Membership, growth, and evolution,” in Proc. of ACM SIGKDD, 2006.

[40] P. Sarkar and A. Moore, “Dynamic social network analysis using latent space models,” ACM SIGKDD Explorations Newsletter, vol. 7, no. 2, pp. 31–40, 2005.

[41] W. Willinger, R. Rejaie, M. Torkjazi, M. Valafar, and M. Maggioni, “Research on online social networks: Time to face the real challenges,” in HotMetrics, 2009.

[42] M. Stumpf, C. Wiuf, and R. May, “Subnets of scale-free networks are not scale-free: sampling properties of networks,” Proceedings of the
VIII. APPENDIX A: UNIFORM SAMPLE OF USER IDS WITH REJECTION SAMPLING

In Section III-C we obtained a uniform sample of Facebook users, by directly querying the user ID space. We then used it as ground truth for comparing against the samples collected by the crawling techniques. More specifically, we used the following procedure:

**UNI Sampling Method:** First, pick a user ID \( X \) uniformly at random in the known range of IDs \([0, \text{MAXUserID}]\). Query the OSN for user data for the chosen user ID \( X \). If the OSN does not return an error message (\( i.e., X \) is valid/allocated), include \( X \) in the sample; otherwise discard it. Repeat the process.

Note that the user ID space is not required to be sequentially or evenly allocated. In fact, as we explicitly show in Section Y-B3 and Fig [2] of the Facebook user ID space is not allocated in a sequential way. Nevertheless, by using rejection sampling, we are required to obtain a uniform sample of the allocated user IDs (not of all user IDs), as the following proposition indicates.

**Proposition:** UNI yields a uniform sample of the allocated user IDs in an Online Social Network.

**Proof.** Let us consider that there are \( N \) allocated and \( M \) non-allocated user IDs in the entire user ID space \([0, \text{MAXUserID}]\). These \( N \) and \( M \) user IDs do not need to be consecutive. UNI picks any element, w.p. \( \frac{1}{N+M} \), accepts it if it is valid (w.p. \( \text{Pr}\{\text{accepted}\} = \frac{N+M}{N+M} \)); otherwise it rejects it (w.p. \( \frac{M}{N+M} \)). It is easy to see the distribution of the accepted user IDs is uniform:

\[
\text{Pr}\{X|\text{accepted}\} = \frac{1}{N}.
\]

Indeed, \( \text{Pr}\{X|\text{accepted}\} = \frac{\text{Pr}\{X\text{ and accepted}\}}{\text{Pr}\{\text{accepted}\}} \). If the user ID is valid then \( \text{Pr}\{X|\text{accepted}\} = \frac{\frac{1}{N+M}}{\frac{1}{N+M}} = \frac{1}{N} \); otherwise it is 0.

The above is just a special case of textbook rejection sampling (\( e.g., \) see [57], Chapter 3), when the desired sampling distribution is uniform. It is only repeated here for completeness. A common misunderstanding is to interpret UNI as a uniform sample of the entire user ID space, while it is a uniform sample only of the allocated/valid user IDs, independently from where in they user ID space these IDs may be.

**Limitations.** There are some requirements for being able to implement UNI, which are met in our measurements of Facebook.

- First, we need to know or estimate the range of user IDs assigned to users, or equivalently the maximum known user ID, \text{MAXUserID}. Since we use rejection sampling, overestimating \text{MAXUserID} simply results to more rejections. Knowledge of the actual distribution of user IDs in the user ID space is not needed, as explained above.
- Second, we need to be able to query the OSN to return data for the selected user ID, if it is valid, or return an error message if it does not exist. This was supported by Facebook at the time of our experiment.

[43] B. Krishnamurthy, “A measure of online social networks,” in Communication Systems and Networks and Workshops, 2009. COMSNETS 2009. First International. IEEE, 2009, pp. 1–10.

[44] A. L. Traud, E. D. Kelsic, P. J. Mucha, and M. A. Porter, “Community structure in online collegiate social networks,” 2008, arXiv:0809.0960.

[45] B. Krishnamurthy and C. E. Wills, “Characterizing privacy in online social networks,” in Proceedings of the 1st ACM SIGCOMM Workshop on Social Networks (WOSN’08), 2008.

[46] M. Valafar, R. Rejaie, and W. Willinger, “Beyond friendship graphs: a study of user interactions in Flickr,” in Proceedings of the 2nd ACM workshop on Online social networks. ACM, 2009, pp. 25–30.

[47] J. Bonneau, J. Anderson, R. Anderson, and F. Stajano, “Eight friends are Facebook statistics,” http://facebook.com/press/info.php?statistics, December 2010.

[48] D. Liben-Nowell, J. Novak, R. Kumar, P. Raghavan, and A. Tomkins, “Geographic routing in social networks,” in Proceedings of the Second ACM EuroSys Workshop on Social Network Systems. ACM, 2009, pp. 13–18.

[49] M. Gjoka, M. Sirivianos, A. Markopoulo, and X. Yang, “Poking Facebook: characterization of osn applications,” in Proceedings of the 1st ACM SIGCOMM Workshop on Social Networks (WOSN’08), 2008.

[50] D. Liben-Nowell, J. Novak, R. Kumar, P. Raghavan, and A. Tomkins, “Geographic routing in social networks,” in Proc. of the National Academy of Sciences, 2005.

[51] M. Cha, H. Kwak, P. Rodriguez, Y.-Y. Ahn, and S. Moon, “I Tube, You Tube, Everybody Tubes: Analyzing the World’s Largest User Generated Content Video System,” in Proc. of IMC, 2007.

[52] P. Gill, M. Arlitt, Z. Li, and A. Mahanti, “Youtube traffic characterization: a view from the edge,” in Proceedings of the 7th ACM SIGCOMM conference on Internet measurement. ACM, 2007, pp. 15–28.

[53] M. Gjoka, M. Kurant, C. Butts, and A. Markopoulo, “Walking in Facebook: A case study of unbiased sampling of osns,” in Proc. of INFOCOM, 2010.

[54] M. Gjoka, C. T. Butts, M. Kurant, and A. Markopoulo, “Multigraph sampling of online social networks,” CoRR, vol. abs/1008.2565, 2010.

[55] B. Ribeiro, P. Wang, and D. Towsley, “On Estimating Degree Distributions of Directed Graphs through Sampling,” Sept. 2010.

[56] M.Hansen and W.Hurwitz, “On the theory of sampling from finite populations,” Annals of Mathematical Statistics, vol. 14, 1943.

[57] E. Volz and D. D. Heckathorn, “Probability based estimation theory for respondent-driven sampling,” Journal of Official Statistics, 2008.

[58] A. Leon-Garcia, Probability, Statistics, and Random Processes For Electrical Engineering. Prentice Hall, 2008.

[59] “Facebook announcement on 64 bit user ids,” http://developers.facebook.com/blog/post/2226, May 2009.

[60] J. Geweke, “Evaluating the accuracy of sampling-based approaches to calculating posterior moments,” in Bayesian Statistics 4, 1992.

[61] A. Gelman and D. Rubin, “Inference from iterative simulation using multiple sequences,” Statistical science, vol. 7, no. 4, pp. 457–472, 1992.

[62] “The facebook blog: Growing beyond regional networks,” http://blog.facebook.com/blog.php?post=91242982130 June 2009.

[63] “Facebook’s id numbering system,” http://www.quora.com/What-is-the-history-of-Facebooks-ID-numbering-system.

[64] D. B. Rubin, “Using the SIR algorithm to simulate posterior distributions,” in Bayesian Statistics 3, 1988.

[65] Ø. Skare, E. Balviken, and L. Holden, “Improved Sampling-Importance Resampling and Reduced Bias Importance Sampling,” Scandinavian Journal of Statistics, vol. 30, no. 4, pp. 719–737, 2003.

[66] M. Newman, “Assortative mixing in networks,” Physical Review Letters, vol. 89, no. 20, 2002.

[67] “Facebook statistics,” http://facebook.com/press/info.php?statistics, December 2010.

[68] “Inside facebook,” http://www.insidefacebook.com/2009/07/02/facebook-now-growing-by-over-700000-users-a-day-updated-engagement-stats/.

[69] J. Leskovec, J. Kleinberg, and C. Faloutsos, “Graph evolution: Densification and shrinking diameters,” ACM Transactions on Knowledge Discovery from Data (TKDD), vol. 1, no. 1, p. 2, Mar. 2007.

[70] R. Albert, H. Jeong, and A.-L. Barabasi, “Internet: Diameter of the world wide web,” in Nature, 1999.

[71] J. Leskovec, K. Lang, A. Dasgupta, and M. Mahoney, “Community structure in large networks: Natural cluster sizes and the absence of large well-defined clusters,” Internet Mathematics, vol. 6, no. 1, pp. 29–123, 2009.

[72] “Snap graph library,” http://snap.stanford.edu/snap/index.html
Furthermore, UNI is efficient only if the probability of acceptance is high. This was the case at the time of our experiments, when the userID was 32bits, but is no longer the case now that Facebook switched to 64bit IDs. We emphasize that we use UNI only as a baseline for comparing the crawling methods against ground truth.

APPENDIX B: TEMPORAL DYNAMICS

The Facebook friendship graph can be considered practically static in the timescales our crawls, consistently with our assumption A4. Facebook is growing in general (as reported by websites such as [66], [67]) but in much longer timescales than the duration of our crawls (which were in the order of a couple days, thanks to the efficient crawlers we developed). To confirm that this is indeed the case in Facebook, we took the following steps.

First, we compared our metrics of interest between the UNI sample of Table III and a similarly sized UNI sample obtained 45 days later. Fig 19 shows the distribution of node degree and privacy settings. One can easily observe that the two distributions we obtained were virtually identical. The same was true in all other comparisons we did on the same metrics. In particular, the node degree distributions we obtained were virtually identical. The same was true in all other comparisons we did on the same metrics. In particular, the node degree distributions we obtained were virtually identical. The same was true in all other comparisons we did on the same metrics.

Second, during the period that we did our crawls (see table III), Facebook was growing at a rate of 450K/day as reported by websites such as [66], [67]. With a population of ~200M users during that period, this translates to a growth of 0.22% of users/day. Each of our crawls lasted around 4-7 days (during which, the total Facebook growth was 0.9%-1.5%); in fact, our convergence analysis shows that the process converged even faster, i.e., in only one day. Therefore, the relative node growth of Facebook was negligible during our crawls.

Third, we evaluated the edge growth of the Facebook graph by conducting the following experiment. In November 2010 we ran 20 independent RW crawls and collected 20 × 10K = 200K user samples within 12 hours. This will be our user sample |S|^0 at Day 0. For the next 5 days, we re-visit the exact same nodes in the same order and check their edges to see if there is any change. These will be our user samples |S| |S|^5, at Days 1,..5 and their collection is approximately equally spaced in time. Let us define \( \Delta k^v_{v, day} = [k^v_{v, day} - k^v_{v, 0}] \) as the absolute degree change where day \( ∈ [1,5] \) and \( \Delta k^v_{v, day} / k^v_{v, day} \) as the relative degree change. Fig 20(a) shows the absolute degree and relative degree change as a function of node degree for Days 1,5. We observe that at Day 5 the relative degree change varies between 0.5%-2% depending on the node degree. To summarize the total relative change at Day 5, we estimate the value

\[
\frac{1}{|S|} \sum_{v \in |S|} \frac{\Delta k^v_{v, 5}}{k^v_{v, 0}}
\]

which is 1.13%. Fig 20(b) shows the CDF of relative and absolute degree change for Days 1,..5. It can be seen that 50% and 80% of the users at Day 5 and Day 1 respectively have an absolute degree change of one or zero. Additionally, 70% and 95% of the users at Day 5 and Day 1 respectively have a relative degree change of less than 1%.

Last, and perhaps most importantly, another way to evaluate the effect of edge growth in the estimation of the degree distribution (or any other property) is to compare the empirical distribution of node degree variance, \( \sum_{v \in |S|} (\Delta k^v_{v, 5})^2 \), with the node degree variance due to temporal dynamics at day 5, to the empirical distribution of node degree variance at day 5, \( \sum_{v \in |S|} (\bar{k} - k^v_{v, 5})^2 \), which is the variance of the node degree distribution at day 5. The former value is estimated at 556 while the latter at 617,829, which makes the node degree variance due to the temporal dynamics at least three orders of magnitude smaller than the distribution variance, i.e., essentially noise.

These results are expected, since we deal with the social graph, which is much more static than other types of graphs, i.e., the interaction graph on OSNs or P2P graphs [18], [41] that are known to have high churn. In the latter case,
considering the dynamics becomes important. Thus, while issues of dynamics are important to consider when sampling changing graphs, they appear not to be problematic for this particular study.

APPENDIX C: MHRW VS. RWRW

Although both MHRW and RWRW achieve asymptotically unbiased samples, experimental results on Facebook (Section V-C) showed that RWRW outperforms MHRW in terms of sampling efficiency. In section V we compared the two experimentally on Facebook. Fig. 9 shows that RWRW is 3 times better than MHRW. This is consistent with what was reported by Rasti et al. [17] in the context of unstructured peer-to-peer networks.

In this appendix, we investigate this further via analysis and simulation and we try to understand why RWRW outperforms MHRW and whether this observation can be generalized. We take the following steps: (i) we run simulations on a broad range of Internet topologies, (ii) we attempt to gain more intuition via analysis, and (iii) we show a counter example where MHRW is better than RWRW in a specific pathological case. Our conclusion is that RWRW is more efficient than MHRW in most topologies that are likely to arise in practice.

A. Simulation results

We compare RWRW with MHRW on a broad range of large, real-life, but fully known Internet topologies described in Table V. As our main source of data we use the SNAP Graph Library [21]. We show the results in Fig. 21. The number on the bottom-left corner of every plot indicate how much longer MHRW should be than RWRW so as to achieve the same error. One can see that, in all cases, MHRW requires 1.5-7 times more (unique) samples to achieve the same estimation quality as RWRW. Recall that in our Facebook measurement, this advantage of RWRW over MHRW is about 3.0 (see Fig. 9).

B. Intuition

Why does RWRW perform so much better than MHRW on real-life Internet topologies? There are three aspects that play a major role here, as discussed below.

1) MHRW does not really sample more (unique) low-degree nodes: By design, MHRW prefers visiting low-degree nodes over high-degree ones, which compensates for the node degree bias observed under RW. However, counter-intuitively, MHRW does not visit significantly more unique low-degree nodes than RW does, but re-samples the same node. Indeed, under MHRW, the probability of leaving the node \( u \) at any iteration is

\[
p_u = \sum_{w \in \text{Neighbors of } u} \frac{1}{k_w} \cdot \min(1, \frac{k_u}{k_w})
\]  

(1)

Therefore, the number of rounds MHRW stays in node \( u \) is a geometric random variable \( \text{Geom}(p_u) \) with parameter \( p_u \) and mean \( 1/p_u \). In some graphs, \( 1/p_u \) may be in the order of thousands.

2) Re-sampling geometric process introduces variance: RWRW compensates for the bias of RW by dividing each measured value of node \( u \) by its degree \( k_u \), which is a fixed number. In contrast, MHRW achieves it by re-sampling node \( u \), as shown above, which is a random process on its own that introduces additional variance. We demonstrate this using the example below. Consider a star topology with node \( v^* \) in the middle and nodes \( V \) around. Let \( V' \subset V \) be the nodes of some type of interest. Our goal is to estimate the fraction \( \theta = \frac{|V'|}{|V|} \). To this end, we use RW and MHRW to collect a sample \( S \subset V \) of nodes (for simplicity, we intentionally ignore node \( v^* \)) of size \( N = |S| \). Clearly, we can estimate \( \theta \)}
uniformly at random, with replacements. Therefore, towards the sample size already known node is free in terms of required bandwidth.

the MHRW estimator. In practice, however, re-sampling an same nodes, which drastically limits the performance of in mixing:

deviation error bars (not shown) are comparable with the symbol size. The standard average is equal to 13), as a function of total sample length. The standard Square Error (NMSE) of the estimation of the average node number (true value of high degree are more likely to act as hubs, i.e., to span different parts of the graph than the low-degree nodes whose connections are usually more local. Therefore, by avoiding hubs, MHRW misses mixing opportunities. In contrast, RW exploits them.

C. Counterexample

Interestingly, although RW outperformed MHRW on all real-life topologies we tested, it is not true that RW is always more efficient. We show one, artificially constructed and rather unrealistic, counterexample in Fig. 22(a). Every node carries a value equal to its number and our goal is to estimate the average value $\theta$ (in this example $\theta = 13$). If the sampling process stays in the clique $v_1 \ldots v_{10}$, then the estimation will be strongly underestimated (probably resulting in $\hat{\theta} \simeq 5.5$. Similarly, staying in the clique $v_{16} \ldots v_{25}$ yields $\hat{\theta} \simeq 20.5$. In order to achieve a good estimation $\hat{\theta} \simeq \theta$, our process should be able to switch between the two cliques.

Inside a clique, say inside $v_1 \ldots v_{10}$, RW and MHRW behave almost identically. The differences appear once we enter the line section at $v_{11}$. MHRW will typically stay on this line for many iterations (which is not good per se), and eventually end up in one of the cliques with roughly the same probabilities (which is good). In contrast, RW at $v_{11}$ has a much higher chance to return to clique $v_1 \ldots v_{10}$, which significantly slows down mixing and thus harms estimation performance. Similar reasoning applies when leaving clique $v_{16} \ldots v_{25}$ at node $v_{15}$. Therefore, MHRW should systematically outperform RW in this example. This intuition is confirmed by the simulation results in Fig. 22(b).

D. Summary

In some cases, like the one presented in Fig. 22, MHRW may outperform RWRW. These topologies are, however, degenerate and unlikely to occur in practice. In more practical examples, at least in all real-life topologies we tried, RWRW requires 1.5-7 times fewer unique samples to achieve the same estimation quality as MHRW, which directly translates into several-fold bandwidth gains. For this reason, we strongly recommend RWRW for sampling nodes in OSN and other Internet topologies.

11Note that this example is exactly the opposite of what we described as a typical Internet topology in Appendix VII-B3: the hubs provide local connectivity only, whereas low degree nodes bridge distant parts of the graph.

12Except that a few transitions to node 10 may be ignored under MHRW. This minor difference can be eliminated in a more complex, but less readable, example.