Measuring the Mixing Time of Social Graphs

Abedelaziz Mohaisen
University of Minnesota
Minneapolis, MN 55455, USA
mohaisen@cs.umn.edu

Aaram Yun
University of Minnesota
Minneapolis, MN 55455, USA
aaram@cs.umn.edu

Yongdae Kim
University of Minnesota
Minneapolis, MN 55455, USA
kyd@cs.umn.edu

ABSTRACT

Social networks provide interesting algorithmic properties that can be used to bootstrap the security of distributed systems. For example, it is widely believed that social networks are fast mixing, and many recently proposed designs of such systems make crucial use of this property. However, whether real-world social networks are really fast mixing is not verified before, and this could potentially affect the performance of such systems based on the fast mixing property. To address this problem, we measure the mixing time of several social graphs, the time that it takes a random walk on the graph to approach the stationary distribution of that graph, using two techniques. First, we use the second largest eigenvalue modulus which bounds the mixing time. Second, we sample initial distributions and compute the random walk length required to achieve probability distributions close to the stationary distribution. Our findings show that the mixing time of social graphs is much larger than anticipated, and being used in literature, and this implies that either the current security systems based on fast mixing have weaker utility guarantees or have to be less efficient, with less security guarantees, in order to compensate for the slower mixing.

Categories and Subject Descriptors

C.2.0 [**Computer Communication Networks**]: General – *Security and Protection*; C.4 [**Performance of Systems**]: Design studies

General Terms

Security, Design, Experimentation

Keywords

Social networks, Sybil defenses, Mixing time, Measurement

1. INTRODUCTION

Popularity of social networks have stimulated many ideas for using these networks to build revolutionary systems in many areas, including security and communication [24, 31, 3, 30, 12, 32, 11, 23, 25, 22, 29]. The systems built on top of social networks exploit algorithmic properties of the social graph, as well as the social trust. For instance, some security designs exploit the "fast mixing" property, an indicator of how quickly a random walk on a graph approaches the stationary distribution, to build Sybil [4] defenses [31, 3, 30, 12, 32, 11, 23]. Some other designs use node betweenness, an indicator of how a node is well-situated on the path between other nodes in the graph, for building Sybil defense as well [19]. There are also designs which use betweenness and similarity for building routing algorithms in disconnected networks [2], among many other designs based on different assumptions.

The applicability and effectiveness of these designs are critically dependent on the quality or degree of these properties in underlying social graphs. But, while they base their constructions and designs on these properties—assuming a high quality of the properties in the social graphs, they do not give conclusive evidences of the quality of these properties. It is claimed that these properties hold, based on mathematical models and indirect experiments, but it is hard to find a single work that evaluates the qualities of such properties *directly* in actual social networks. But doing that, it will be then possible to determine the exact quality of the property required for these designs to work.

For example, the mixing time of the social graph, which measures how quick a random walk on the graph reaches the stationary distribution, is claimed to be *fast*. Such claim implies that social graphs are well-enmeshed and any arbitrary destination in the social graph is reachable, with a probability driven according to the stationary distribution—a distribution that is proportional to nodes' degrees, from each possible source with a relatively small number of intermediaries. Furthermore, this property has been used widely without careful measurement of the mixing time [31, 3, 30, 12]. For example, Yu et al. [31] proposed SybilGuard, a Sybil defense protocol that exploits the fast mixing property of social graphs. Even though they performed experiments on social networks, their experiment was not about actual measurement of the mixing time of these graphs (see Section 2). Danezis and Mittal [3] proposed SybilInfer to detect Sybil nodes in social graph basing their design on the fast mixing property of social graph and cited [18] as an evidence to prove that social networks are fast mixing. We notice, however, that findings in [18] do not support the mixing time with the guarantees needed by SybilInfer. Lesniewski-Laas et al. [12] introduced Whānau, a Sybil-proof routing protocol that uses the fast mixing property and, while citing the existence of the property to a large body of previous work, they have attempted to estimate the mixing time in a few social graphs. However, their evidence is only circumstantial and it does not directly follow that these social graphs are really fast mixing (see Section 2).

In this paper, we evaluate the mixing time of social graphs. We systematically measure the mixing time of social graphs in a variety of real-world small to large-scale social networks (see Table 1),

with designs based on these properties in mind. We use two methods for measuring the mixing time. First, we compute the Second Largest Eigenvalue Modulus (SLEM) of the social graphs which bounds the mixing time. Also, we sample initial points and compute the random walk with varying walk lengths. We find that in many actual large-scale social networks, the mixing time is much larger than suggested by those papers, which apply the fast mixing of social networks to design security systems. By experimenting with one of these systems—SybilLimit [30], we unveil that the quality of the mixing time required for such design is not as being claimed, yet in some real-world social graphs higher than anticipated, which calls for further investigation of the theory beyond these designs.

The rest of the paper is organized as follows. In section 2, we review some of the related work in literature. In section 3, we review the preliminaries including the network model, the random walk, and the mixing time. In section 4, we introduce the main results followed by discussion in section 5. Finally, in section 6 we draw concluding remarks and future work.

2. RELATED WORK

There are many systems built on top of social graphs and their properties. Daly et al. [2] proposed a social network-based scheme for routing in disconnected delay-tolerant mobile ad-hoc networks which uses both of the betweenness and similarity properties. Quercia et al. [19] used the betweenness property to defend against the Sybil attack in mobile networks. Yu et al. [30, 31] used the fast mixing property of a graph to build a defense mechanism against the Sybil attack. Danezis and Mittal [3] used the fast mixing property to build an inference (detection) mechanism for Sybil nodes in peer-to-peer Systems. Lesniewski-Laas et al. [12] introduced a routing protocol that uses the fast mixing property of the social graph. Kaustz et al. introduced ReferralWeb [7], a referral system that combines social networks and collaborative filtering and assumes a well-connected social network graph, a property that is very tied to the mixing time of the graph [6].

Schemes like SybilGuard [31] and SybilLimit [30] of Yu et al., and Whānau of Lesniewski-Laas et al. [12] are based on the fast mixing property of social networks, and they did perform experiments on some real social networks. But their experiments did not directly measure or estimate the mixing time of these social networks. Let us summarize contents of their experiments as follows.

Yu et al. [30] performed some experiments based on real-world social graphs. They ran their scheme with fixed, small walk length (e.g., 10 or 15), and checked whether their scheme works as well as expected (thereby indirectly trying to confirm that the graph is fast mixing). But, there are some deficiencies in their methodology. First, they manipulated the social graphs by trimming lower degree nodes in order to improve the mixing time. Second, their method used several parameters chosen heuristically without showing how these parameters are related to the mixing time. Last, they evaluated only three social graphs which would be too small for making a general conclusion for all social graphs. We would also like to point out that they measured only the false acceptance rate (i.e., the rate of accepted sybil nodes per honest nodes) and not other characteristics, like the rejection rate of honest nodes, which would be expected to increase with insufficient walk lengths. Experiments done in the SybilGuard [31] paper are similar.

Lesniewski-Laas et al. [12] also performed experiments on four large-scale social graphs. They produced CDF of tail edges of random walks with varying walk lengths, and expected that as the walk length approaches $O(\log n)$, probabilities that a random walk ends at a certain edge tend to approach $1/m$, the uniform probability

over edges. But the convergence is very loose; they claim that as the walk length approaches 80[1], each CDF approaches the ideal uniform distribution, but among the social networks in their measurement, at least the LiveJournal result shows the distribution is very far from uniform at the walk length 80. The other three results also allow a lot of deviations from the uniform distribution which make it unlikely that the total variation distance between the distribution and the uniform distribution is close to 0. In short, they provided raw measurements but did not relate the distribution of the sampled tails to the stationary distribution itself, in terms of the variation distance.[2]

Recently, and concurrent to this work, Viswanath et al. conducted an experimental analysis of sybil defenses based on social networks in [27]. Their study aimed at comparing different defenses (namely, SybilGuard [32], SybilLimit [30], SybilInfer [3], and SumUp [23]) independent of the data sets being used, by decomposing these defenses to their cores. They show that the different Sybil defenses work by ranking different nodes based on how well-connected are these nodes to a trusted node (the verifier). Also, they show that the different Sybil defenses are sensitive to community structure in social networks and community detection algorithms can be used to replace the random walk based Sybil defenses. In conclusion, results on the poor performance of Sybil defenses when applied to community structure possessing social graphs agree with our findings, where we show that such networks are slow mixing.

3. PRELIMINARIES

In this section, we formalize the network model. We define the mixing time of a random walk on a graph, and we also define the fast mixing property of a graph.

3.1 Network model

The social network can be viewed as an undirected graph $G = (V, E)$ where V is the set of nodes (social actors) in the graph and E is the set of edges (relationships or interdependencies) between the nodes. The size of the graph $n = |V|$ and the number of edges in G is $m = |E|$. We define the degree of a node $v_i \in V$ as the number of nodes in V adjacent to v_i and denote it by $\deg(v_i)$. For G, we define the stochastic transition probability matrix $P = [p_{ij}]$ of size $n \times n$ where the $(i, j)^{\text{th}}$ entry in P is the probability of moving from node v_i to node v_j defined as

$$p_{ij} = \begin{cases} \frac{1}{\deg(v_i)} & \text{if } v_i \text{ is adjacent to } v_j, \\ 0 & \text{otherwise.} \end{cases} \tag{1}$$

3.2 Mixing time

In this subsection we recall definitions of some notions about random walks on graphs. For more detailed exposition, see [21].

The "event" of moving from a node to another in the graph is captured by the Markov chain which represents a random walk over the graph G. A random walk R of length k over G is a sequence of vertices in G begining from an initial node v_i and ending at v_t, the terminal node, following the transition probability defined in (1). The Markov chain is said to be *ergodic* if it is irreducible and aperiodic. In that case, it has a unique stationary distribution π and the distribution after random walk of length k converges to

---

[1]While 80 is much larger than $\log n$ when n is close to one million, one possibility is that this may be due to the hidden constant.

[2]While [12] uses a different measure, called the separation distance, and does not require ϵ to be too small, the necessary quality of the mixing time is not measured in [12].

π as $k \to \infty$. The stationary distribution of the Markov chain is a probability distribution that is invariant to the transition matrix P (i.e., $\pi P = \pi$). The mixing time of the Markov chain, T is defined as the minimal length of the random walk in order to reach the stationary distribution. More precisely, Definition 1 states the mixing time of a Markov chain on G parameterized by a variation distance parameter ϵ.

DEFINITION 1 (MIXING TIME). *The mixing time (parameterized by ϵ) of a Markov chain is defined as*

$$T(\epsilon) = \max_i \min\{t : |\pi - \pi^{(i)} P^t|_1 < \epsilon\}, \qquad (2)$$

where π is the stationary distribution, $\pi^{(i)}$ is the initial distribution concentrated at vertex v_i, P^t is the transition matrix after t steps, and $|\cdot|_1$ is the total variation distance. We say that a Markov chain is rapidly mixing if $T(\epsilon) = poly(\log n, \log\frac{1}{\epsilon})$ where $n = |V|$.

In literature, the rapid mixing of the Markov chain is cited as "fast mixing" for the graph [3, 12, 30, 31]. In this work, we follow the tradition of referring to this bound as "fast mixing". Also, again following these previous work, we strengthen the definition by considering only the case $\epsilon = \Theta(\frac{1}{n})$, and requiring $T(\epsilon) = O(\log n)$.

THEOREM 1 (STATIONARY DISTRIBUTION). *For undirected unweighted graph G, the stationary distribution of the Markov chain over G is the probability vector $\pi = [\pi_{v_i}]$ where $\pi_{v_i} = \frac{\deg v_i}{2m}$. This is,*

$$\pi = \left[\frac{\deg(v_1)}{2m} \, \frac{\deg(v_2)}{2m} \, \frac{\deg(v_3)}{2m} \cdots \frac{\deg(v_n)}{2m}\right] \qquad (3)$$

Notice that π is uniform for a regular graph with degree d since $m = \frac{n \times d}{2}$ and $\pi = [\frac{d}{2 \times \frac{n \times d}{2}}] = [\frac{1}{n}]$.

THEOREM 2 (SECOND LARGEST EIGENVALUE [21]). *Let P be the transition matrix of G with ergodic random walk, and λ_i for $1 \le i \le n$ be the eigenvalues of P. Then all of λ_i are real numbers. If we label them in decreasing order, $1 = \lambda_1 > \lambda_2 \ge \cdots \ge \lambda_{n-1} \ge \lambda_n > -1$ holds. We define the second largest eigenvalue μ as $\mu = \max(|\lambda_2|, |\lambda_{n-1}|)$. Then, the mixing time $T(\epsilon)$ is bounded by:*

$$\frac{\mu}{2(1-\mu)} \log(\frac{1}{2\epsilon}) \le T(\epsilon) \le \frac{\log(n) + \log(\frac{1}{\epsilon})}{1-\mu} \qquad (4)$$

Mixing time versus connectivity: The mixing time is tightly related to the connectivity of the graph. This is, strongly-connected graphs are fast mixing (i.e., have small mixing time) while the weakly connected graphs are slow mixing and have large mixing time [21]. Also, the second largest eigenvalue used for measuring the mixing time bounds the graph conductance, a measure for the community structure [27]. In short, the conductance is $\Phi \ge 1 - \mu$.

3.3 Measuring the mixing time of social graphs

Measuring the mixing time, especially of large graphs, is a cumbersome task and that might be the reason why fewer efforts are made to measure this essential property in large social graphs. In order to measure the mixing time of a social graph, we begin by the definition itself in (2). We follow the definition, by starting from an initial distribution concentrated on a node v_i, and compute the distribution after the random walk of length t with t large enough so that the variation distance between the distribution after random walk and the stationary distribution is within ϵ. We repeat this for different initial points. This approach is feasible for not too small ϵ, because we may then expect long walk length for this computation.

Since the mixing time is defined as the maximum necessary walk length to achieve ϵ distance for different initial states, one such

random walk would be enough to establish a lower bound of the mixing time. Since we are interested in how large the mixing time should be, in principle only one random walk could be enough, if the walk length is sufficiently large. But in order to understand the general tendency and distribution of walk lengths, we repeat this many times (i.e., 1000) by picking an initial node randomly and perform the above computation. The end result obtained using this technique gives intuition about the tendency of the mixing time.

As a complementary method for bounding the mixing time (for even smaller ϵ and also for comparing with the previous results), we also use the method described in Theorem 2. First, we compute the second largest eigenvalue modulus (SLEM) of the transition matrix P. Given that the matrix P is sparse, we found that the computation of SLEM is feasible for graphs with a million nodes (as is the case for largest graphs we used). Once we compute SLEM, we use the lower bound in (4) to bound the mixing time of the graph.

3.4 Social graphs—the data sets

The social graphs used in our experiments are in Table 1. These graphs are selected to feature two models of knowledge between nodes in the social networks. These networks are categorized as follows. (1) social networks that exhibit knowledge between nodes and are good for the trust assumptions of the Sybil defenses; e.g., physics co-authorships and DBLP. These are slow mixing, as we will see later. (2) Graphs of networks that may not require face-to-face knowledge but require interaction; e.g., Youtube and Livejournal. Closely related to those is the set of graphs that may not require prior knowledge between nodes or where the social links between nodes are less meaningful to the context of the Sybil defenses; e.g., Facebook and wiki-vote, which are shown to be fast mixing.

Table 1: Datasets, their properties and their second largest eigenvalues of the transition matrix

Dataset	Nodes	Edges	μ
Wiki-vote [8]	7,066	100,736	0.899418
Slashdot 2 [10]	77,360	546,487	0.987531
Slashdot 1 [10]	82,168	582,533	0.987531
Facebook [26]	63,392	816,886	0.998133
Physics 1 [9]	4,158	13,428	0.998133
Physics 2 [9]	11,204	117,649	0.998221
Physics 3 [9]	8,638	24,827	0.996879
Enron [9]	33,696	180,811	0.996473
Epinion [20]	4,158	13,428	0.998133
DBLP [13]	614,981	1,155,148	0.997494
Facebook A [28]	1,000,000	20,353,734	0.982477
Facebook B [28]	1,000,000	15,807,563	0.992020
Livejournal A [14]	1,000,000	26,151,771	0.999387
Livejournal B [14]	1,000,000	27,562,349	0.999695
Youtube [14]	1,134,890	2,987,624	0.997972

4. RESULTS

Equipped with the mathematical tools explained in section 3, we measure the mixing time of the different social graphs shown in Table 1. In order to apply the tools in section 3 for measuring the mixing time, we first convert directed graphs to undirected, which is similar to what is performed in other work [27, 3, 12, 11, 23, 30, 31, 32]. We further compute the largest connected component in each graph and use it as a representative social structure for measuring the mixing time, as the mixing time is undefined for disconnected graphs. For small to medium-sized graphs, we compute SLEM directly from the transition matrix of the graph. On the other hand, for feasibility reasons, we sample the representative

subgraphs from each of the four large data sets (Facebook A, B and Livejournal A, B) using the breadth first search (BFS) algorithm beginning from a random node in the graph as an initial point.[3] We perform this sampling process to obtain graphs of 10K, 100K and 1000K nodes out of 3 to 5 million nodes in each original social graph. Bearing the different social graphs sizes in mind, as shown in Table 1, we proceed to describe the results of our experiments.

Figure 1 and Figure 2 plot the lower bound of the mixing time for the different graphs in Table 1. We choose to use the lower-bound, but not the upper bound, because it is more relevant to the context of our study. In particular, as we observe that the lower-bound of the mixing time to satisfy a given ϵ is large, it is obvious that the mixing time for social graphs is slower than anticipated. As shown in Figure 1, we also observe that the mixing time is very slow, in particular for social graphs that require physical acquaintance of the social actors, as can be seen in the general tendency of these graphs. For example, physics co-authorship, Enron, and Epinion, though the social network is small, a mixing time of 200 to 400 is required to achieve $\epsilon = 0.1$. Similarly for larger social graphs, as shown in Figure 2, the mixing time to achieve $\epsilon = 0.1$ is varying and depends on the nature of the data set. For example, while it is about 1500 to 2500 in case of Livejournal, it ranges from 100 to about 400 in case of DBLP, Youtube, and Facebook.
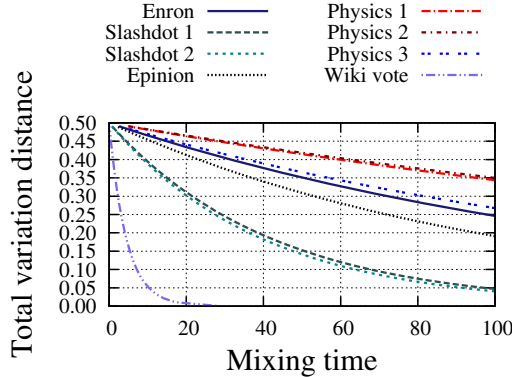

Figure 1: Lower bound of the mixing time for the different data sets used in our experiments — the case of small data sets.
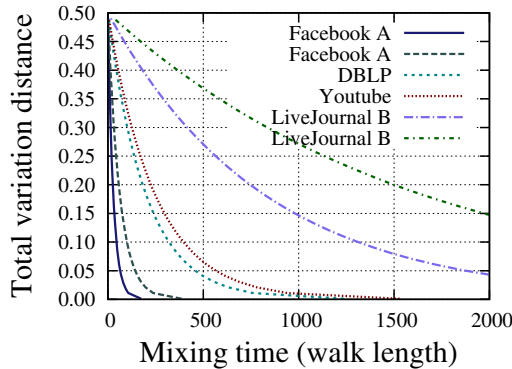

Figure 2: Lower bound of the mixing time for the different data sets used in our experiments — the case of large data sets

To see how tight are these measurements we perform the following experiment. We first compute the lower bound of the mixing time for the physics co-authorship data sets, which are also reasonably small and feasible to do exhaustive computations. Then we measure the mixing time using the model in (2) from every possible source in the graph (the CDFs of the raw measurements are

---

[3]Note that BFS algorithm may bias the sampled graph to have faster mixing. Since our goal is to show that the mixing time is slower than expected, this only strengthens our position.

shown in Figure 3 for small t and and in Figure 4 for large t). We aggregate these measurements into Figure 5, by sorting ϵ at each t and averaging values in various intervals as percentiles. We observe that while the mixing time of most sources in social graphs is better than that of the mixing time given by SLEM, the measurements using SLEM are correct since the mixing time is by definition maximum of walk lengths for given ϵ as shown in (2). However, even considering this effect, still for most sources the mixing time is slower than used by other papers (10 and 15 in SybilLimit).
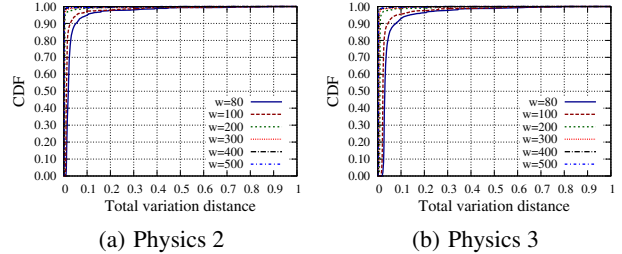

(a) Physics 2   (b) Physics 3

Figure 4: The CDF of mixing time (long walks) for the three physics datasets in Table 1. The variation distance is computed for every possible node in the graph, brute-forcefully.
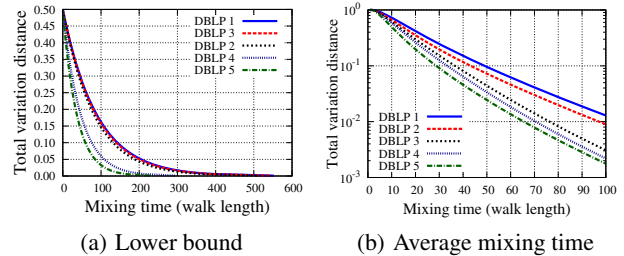

(a) Lower bound   (b) Average mixing time

Figure 6: Lower-bound vs. the top the average mixing time for a sample of 1000 nodes in each data set, where DBLP x means the minimum degree in that data set is x.

To understand the relationship between the network size and the mixing time (of the same social graph) we use the different previously sampled subgraphs, using BFS, from Facebook and Livejournal data sets (10K, 100K, and 1000K). We further measure the mixing time using SLEM and the model in (2) for 1000 initial distributions. We further aggregate the top 10, median 20, and lowest 10 percentile of ϵ corresponding to the given random walk, and plot them along with the mixing time derived using SLEM where the results are shown in Figure 7. We observe that for a million nodes graph, while the mixing time in the top 10% in the sample we computed is 100 for an averaged $\epsilon = 10^{-5}$—an excellent value to the "theoretical" guarantees of the Sybil defenses, the SLEM-based mixing time results in only $\epsilon = 10^{-2}$ as shown in Figure 7(i). We attribute this difference to similar scenario as in the physics co-authorship graphs. Similar observations can be seen in each of the different large social graphs. It is worth mentioning that Livejournal (Figure 7(k) and Figure 7(l)) present poor mixing in relation with Facebook data sets, which are shown to be fast mixing.

Finally, to understand the methodology used for experimenting in Sybilguard and SybilLimit, we perform the same trimming technique by iteratively removing lower degree nodes (for 1 up to 5) from the DBLP data set and computed the mixing time of the resulting graphs at each time (results shown in Figure 6). We observe that the pruning of lower degree greatly improves the mixing time of the social graph: for fixed mixing time of 100, by successive trimming the variation distance is reduced from about 0.2 to 0.03 (Figure 6(a)), and from about 0.015 to 0.002 (Figure 6(b)). But this is with huge reduction of the graph size: DBLP 1 is of size 614,981 but after trimming up to 4 degree nodes, DBLP 5 is of size 145,497.

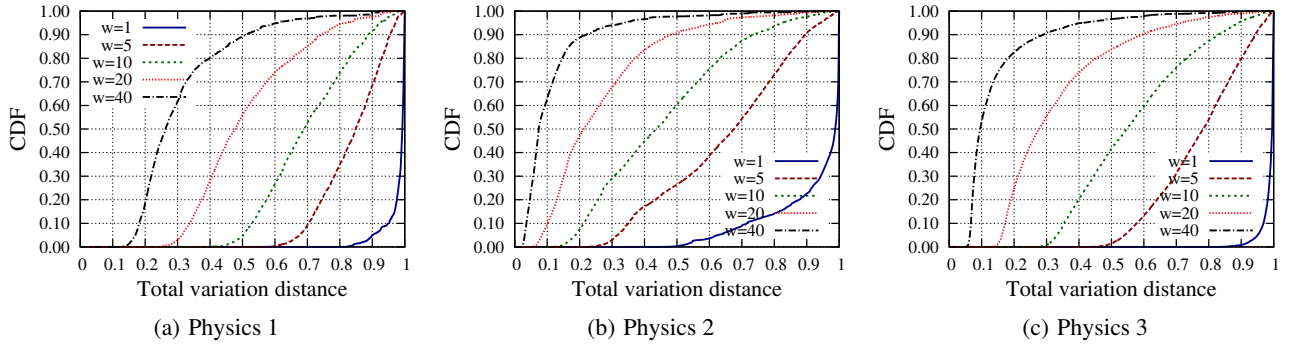(a) Physics 1   (b) Physics 2   (c) Physics 3

Figure 3: The commutative distribution function (CDF) of mixing time for the three physics datasets in Table 1. The variation distance is computed for every possible node in the graph, brute-forcefully.

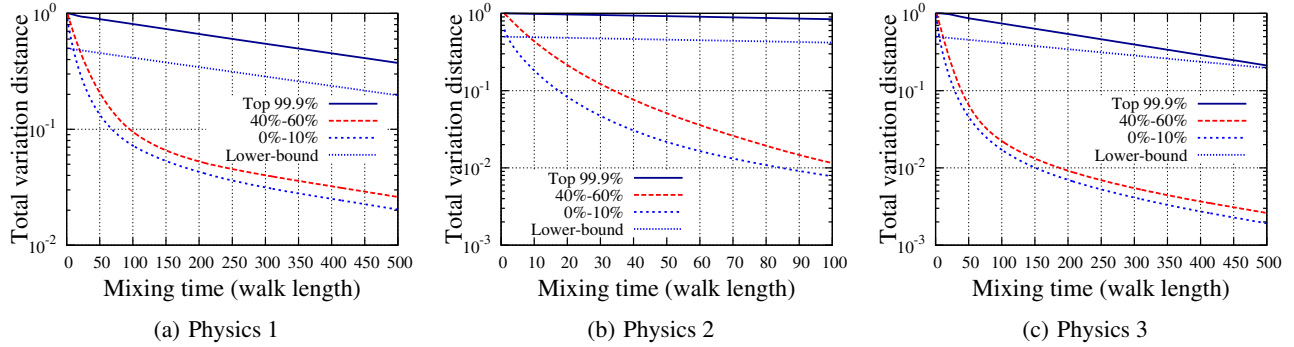
(a) Physics 1   (b) Physics 2   (c) Physics 3

Figure 5: Lower-bound of the mixing time compared to the mixing time when measured using the sampling method for the entire graphs brute-forcefully — different measurements meet the guarantees.

This means that about 75% of nodes are denied joining the service outright in order to boost the mixing time.

5. DISCUSSION

While the main finding in this study is that the mixing time of social graphs is higher than has been used in literature, we also conclude that different nodes approach the stationary distribution at different rates. This is, while the majority of walks initiated from different nodes reach closer to the stationary distribution at "higher" rate than that of the mixing time, which is defined as the maximum rate from any source, we still find—except in a few cases of online social networks—that the mixing time of the majority of nodes is larger than anticipated and used in the previous studies [30, 31, 12]. This has several implications and call for several actions.

First, since most of the theoretical guarantees of social graphs consider the model in (2), and since the majority of nodes in the social graphs measured in this paper have better mixing time than the bound in that model, this calls for rigorous study by basing such designs and analyses on the average case of the mixing, which is relatively small, instead of the worst case of the mixing time.

Second, the obvious implication of our findings is that one has to either give up some of the utility (service) guarantees—which are implied by that almost all honest nodes admit other honest nodes— by using relatively shorter walks, or give up part of the performance and security by enabling longer random walks in order to reach these isolated parts of the social graphs. Though this looks straightforward, going either way is not as simple as it seems. On the one hand, if one uses longer random walks in order to reach such isolated parts of the network it would be equally likely to escape to the Sbyil region which has a cut similar in its nature to that of the slower mixing part of the original social graph. On the other hand, using random walks shorter than the mixing time of the majority of nodes would would also be at the expense of the utility; not only

for the isolated part but also the faster mixing part as well. The end detection guarantees of the design would work as long as g, the number of attack edges is less than $\frac{n}{w}$.

Third, papers introducing SybilGuard, SybilLimit, and Whānau all did experiments on their schemes. Despite the short mixing time that these experiments use, their results seem to support that their schemes work as expected. The explanation of this is two part. First, the trimming of lower-degree nodes would shorten the mixing time. Second, although they claim that the social networks are fast mixing and as a part of the definition—which is also used in parts of the proofs for the theoretical guarantees—they insist $\epsilon = \Theta(1/n)$, this is a very strong burden to achieve and perhaps somewhat larger ϵ might also be good enough for these schemes to work. Also, we suspect that the difference between the average mixing time and the worst-case mixing time may have some effects on the discrepancy between the analysis and the experiment. In practice, the majority of nodes with "fast" mixing would be served and those few other nodes with very slow mixing would be denied service, which then will not be a problem for the probabilistic average-case guarantees.

Finally, one of the assumptions in Sybil defenses based on social networks is that the used trust model requires physical acquaintance, which is the case in social networks such DBLP and Physics co-authorship networks, for which we show slower mixing time than other "online social networks" which are known to possess less strict trust models [5, 1], which by nature tolerate Sybil nodes. This calls for considering the trust model resulting from the underlying social network as a parameter, along with the mixing time, in order to evaluate the effectiveness of the social network-based defenses according to their real value. Our work in [16, 15] is a preliminarily result in this direction.

Performance Implications—the case of SybilLimit: in order to quantitively measure the impact of the findings of slower mixing time on the performance of Sybil defenses, we implement Sybil-
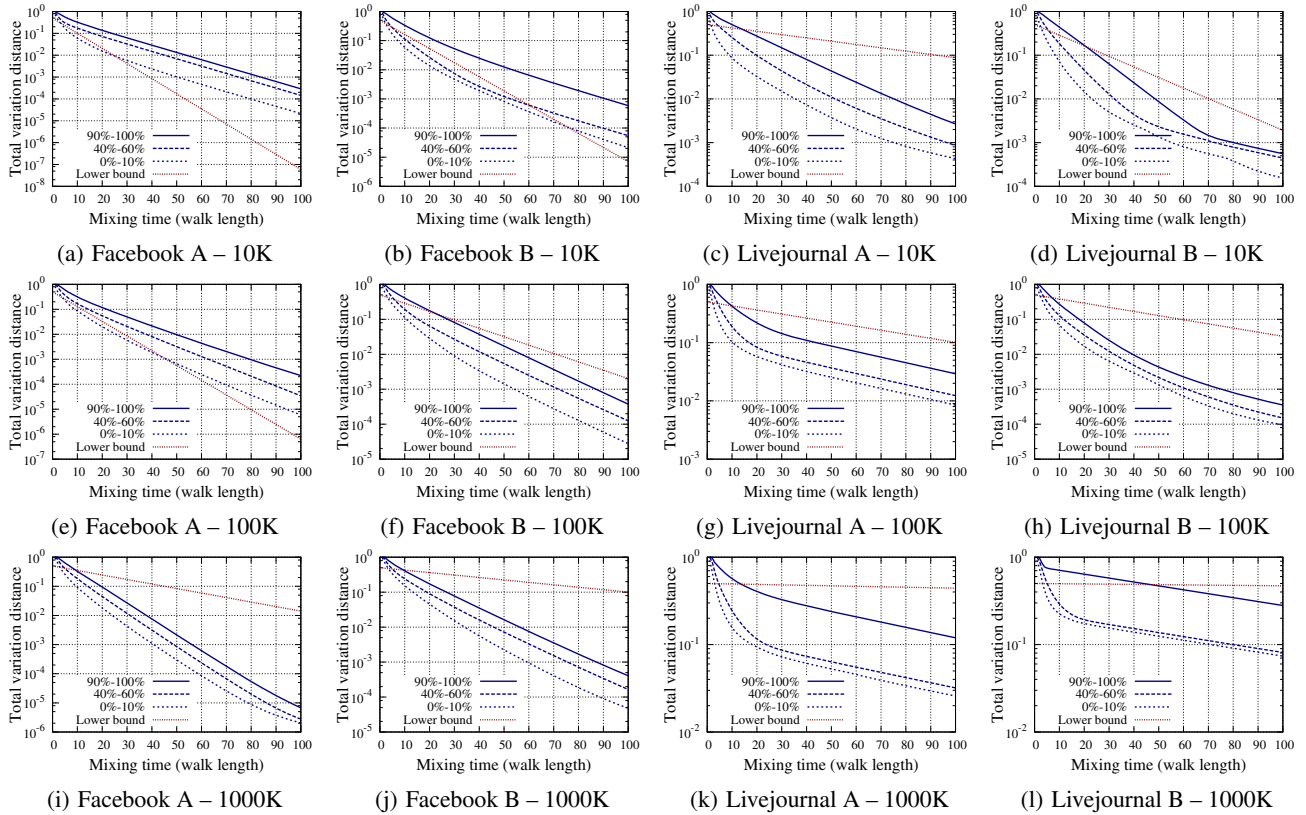
Figure 7: Sampling vs. lower-bound measurements of the mixing time for 10K, 100K and 1000K of four large-scale datasets.

Limit and apply it to some of the different social networks with the following settings. Since we already know the social graphs size—both m and n, we select the proper r that guarantees high probability of intersection. We set r to $r_0\sqrt{m}$, where m is the number of undirected edges in the graph and r_0 is computed from the birthday paradox formula to guarantee a given intersection probability. In this experiment, we consider the case without an attacker, since SybilLimit bounds the number of the Sybil identities introduced based on the number of the attacker edges. Without considering Sybil attacker, we increase t until the number of accepted nodes by a trusted node (the verifier) reaches a almost all honest nodes in the social network. Then, with this t, we find the (average) total variation distance required in each graph, which is the necessary for the operation of these designs. It is then easy to compute the number of accepted Sybil identities which is $t \times g$, where g is the number of attack edges. SybilLimit works as long as $t < \frac{n}{w}$. The result of this experiment is in Figure 8. For more details, see [17]

6. CONCLUSION

In this paper, we measured mixing times of several on-line and and information social networks which may be used for building security defenses and communication systems. Our main finding shows that these social networks generally have much slower mixing time than the previous works anticipated. Meanwhile, we also observed that the average mixing time is better than the worst-case mixing time which is the standard definition of the mixing time of a random walk on a graph, although the average mixing time is again much higher than the ones being used. In the near future, we will investigate building theoretical models that consider the average case of the mixing time. We will also investigate cost models that
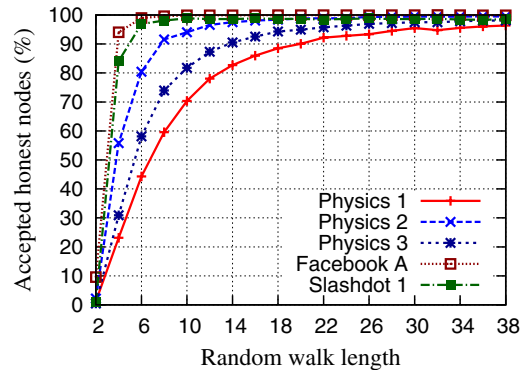

Figure 8: Admission rate of SybilLimit when using different t. Facebook (A) and Slashdot (1) have 10,000 nodes each.

consider the different mixing times of social graphs and their relation of the trust model exhibited in such networks to evaluate the overall effectiveness of design based on social networks. The later part of the future work is motivated by the observation that some social graphs are faster mixing than others while their underlying trust is different.

Acknowledgement

7. REFERENCES

[1] L. Bilge, T. Strufe, D. Balzarotti, and E. Kirda. All your contacts are belong to us: automated identity theft attacks on social networks. In *WWW '09: Proceedings of the 18th international conference on World wide web*, pages 551–560, New York, NY, USA, 2009. ACM.

[2] E. M. Daly and M. Haahr. Social network analysis for routing in disconnected delay-tolerant manets. In *MobiHoc '07: Proceedings of the 8th ACM international symposium on Mobile ad hoc networking and computing*, pages 32–40, New York, NY, USA, 2007. ACM.

[3] G. Danezis and P. Mittal. SybilInfer: Detecting sybil nodes using social networks. In *The 16th Annual Network & Distributed System Security Conference*, 2009.

[4] J. R. Douceur. The sybil attack. In *IPTPS '01: Revised Papers from the First International Workshop on Peer-to-Peer Systems*, pages 251–260, London, UK, 2002. Springer-Verlag.

[5] C. Dwyer, S. Hiltz, and K. Passerini. Trust and privacy concern within social networking sites: A comparison of Facebook and MySpace. In *Proceedings of AMCIS*, 2007.

[6] M. Jerrum and A. Sinclair. Conductance and the rapid mixing property for markov chains: the approximation of the permanent resolved (preliminary version). In *STOC*, pages 235–244. ACM, 1988.

[7] H. A. Kautz, B. Selman, and M. A. Shah. Referral web: Combining social networks and collaborative filtering. *Commun. ACM*, 40(3):63–65, 1997.

[8] J. Leskovec, D. P. Huttenlocher, and J. M. Kleinberg. Predicting positive and negative links in online social networks. In M. Rappa, P. Jones, J. Freire, and S. Chakrabarti, editors, *WWW*, pages 641–650. ACM, 2010.

[9] J. Leskovec, J. Kleinberg, and C. Faloutsos. Graphs over time: densification laws, shrinking diameters and possible explanations. In *KDD '05: Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining*, pages 177–187, New York, NY, USA, 2005. ACM.

[10] J. Leskovec, K. J. Lang, A. Dasgupta, and M. W. Mahoney. Community structure in large networks: Natural cluster sizes and the absence of large well-defined clusters. *CoRR*, abs/0810.1355, 2008.

[11] C. Lesniewski-Laas. A Sybil-proof one-hop DHT. In *Proceedings of the 1st workshop on Social network systems*, pages 19–24. ACM, 2008.

[12] C. Lesniewski-Lass and M. F. Kaashoek. Whānau: A sybil-proof distributed hash table. In *7th USENIX Symposium on Network Design and Implementation*, pages 3–17, 2010.

[13] M. Ley. The DBLP computer science bibliography: Evolution, research issues, perspectives. In *String Processing and Information Retrieval*, pages 481–486. Springer, 2009.

[14] A. Mislove, M. Marcon, P. K. Gummadi, P. Druschel, and B. Bhattacharjee. Measurement and analysis of online social networks. In *Internet Measurement Comference*, pages 29–42, 2007.

[15] A. Mohaisen, N. Hopper, and Y. Kim. Designs to account for trust in social network-based sybil defenses. In *17th ACM Conference on Computer and Communications Security*, Chicago, IL, USA, 2010. ACM.

[16] A. Mohaisen, N. Hopper, and Y. Kim. Keep your friends close: Incorporating trust in social network-based sybil defenses. Technical report, University of Minnesota, 2010.

[17] A. Mohaisen, A. Yun, and Y. Kim. Measuring the mixing time of social graphs. Technical report, University of Minnesota, 2010.

[18] S. Nagaraja. Anonymity in the wild: Mixes on unstructured networks. In N. Borisov and P. Golle, editors, *Privacy Enhancing Technologies*, volume 4776 of *Lecture Notes in Computer Science*, pages 254–271. Springer, 2007.

[19] D. Quercia and S. Hailes. Sybil attacks against mobile users: friends and foes to the rescue. In *INFOCOM'10: Proceedings of the 29th conference on Information communications*, pages 336–340, Piscataway, NJ, USA, 2010. IEEE Press.

[20] M. Richardson, R. Agrawal, and P. Domingos. Trust management for the semantic web. In D. Fensel, K. P. Sycara, and J. Mylopoulos, editors, *International Semantic Web Conference*, volume 2870 of *Lecture Notes in Computer Science*, pages 351–368. Springer, 2003.

[21] A. Sinclair. Improved bounds for mixing rates of marcov chains and multicommodity flow. *Combinatorics, Probability & Computing*, 1:351–370, 1992.

[22] N. Tran, J. Li, L. Subramanian, and S. S. M. Chow. Brief announcement: improving social-network-based sybil-resilient node admission control. In A. W. Richa and R. Guerraoui, editors, *PODC*, pages 241–242. ACM, 2010.

[23] N. Tran, B. Min, J. Li, and L. Subramanian. Sybil-resilient online content voting. In *USENIX NSDI*, 2009.

[24] E. Vasserman. *Towards freedom of speech on the Internet: Censorship-resistant communication and storage*. PhD thesis, UNIVERSITY OF MINNESOTA, 2010.

[25] E. Vasserman, R. Jansen, J. Tyra, N. Hopper, and Y. Kim. Membership-concealing overlay networks. In *Proceedings of the 16th ACM conference on Computer and communications security*, pages 390–399. ACM, 2009.

[26] 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)*, August 2009.

[27] B. Viswanath, A. Post, K. P. Gummadi, and A. Mislove. An analysis of social network-based sybil defenses. In *SIGCOMM*, 2010.

[28] C. Wilson, B. Boe, A. Sala, K. P. Puttaswamy, and B. Y. Zhao. User interactions in social networks and their implications. In *EuroSys '09: Proceedings of the 4th ACM European conference on Computer systems*, pages 205–218, New York, NY, USA, 2009. ACM.

[29] S. Xu, X. Li, and P. Parker. Exploiting social networks for threshold signing: attack-resilience vs. availability. In *ASIACCS '08: Proceedings of the 2008 ACM symposium on Information, computer and communications security*, pages 325–336, New York, NY, USA, 2008. ACM.

[30] H. Yu, P. B. Gibbons, M. Kaminsky, and F. Xiao. SybilLimit: A near-optimal social network defense against sybil attacks. In *IEEE Symposium on Security and Privacy*, pages 3–17, 2008.

[31] H. Yu, M. Kaminsky, P. B. Gibbons, and A. Flaxman. SybilGuard: defending against sybil attacks via social networks. In *SIGCOMM*, pages 267–278, 2006.

[32] H. Yu, M. Kaminsky, P. B. Gibbons, and A. D. Flaxman. SybilGuard: defending against sybil attacks via social networks. *IEEE/ACM Trans. Netw.*, 16(3):576–589, 2008.