Dynamix: Anonymity on Dynamic Social Structures

Abedelaziz Mohaisen
Verisign Labs, VA, USA

Yongdae Kim
KAIST, Daejeon, South Korea

ABSTRACT

In this paper we advance communication using social networks in two directions by considering dynamics of social graphs. First, we formally define the problem of routing on dynamic graphs and show an interesting and intuitive connection between graph dynamics and random walks on weighted graphs; graphs in which weights summarize history of edge dynamics and allow for future dynamics to be used as weight adjustment. Second, we present several measurements of our proposed model on dynamic graphs extracted from real-world social networks and compare them to static structures driven from the same graphs. We show several interesting trade-offs and highlight the potential of our model to capture dynamics, enrich graph structure, and improves the quantitative sender anonymity when compared to the case of static graphs.

Categories and Subject Descriptors

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

Keywords

Social networks, Anonymity, Dynamics, P2P communication.

1. INTRODUCTION

Social networks provide rich algorithmic and structural properties that can be used for building certain classes of application benefiting from these properties. When considered along with the main characteristic of social networks, trust, the potential of these networks becomes very promising in solving real-world problems. For example, social networks have been proposed as a building block to defend against the Sybil attack [19], to enable routing in delay tolerant networks[3, 6], and to provide peer-to-peer and private communication [14, 9, 4]. However, social network-based systems make certain assumptions towards achieving their goals; trust among nodes is assumed binary [10], associations are bidirectional [12], and static [14]. For example, insight is brought on the potential of these designs by experimenting with static social graphs, and by ignoring the dynamic nature of social graphs. Ignoring this nature might be due to unavailability of tools to capture the dynamic nature of social graphs, or the unavailability of measures to quantify the performance of the proposed designs on such dynamic social graphs. However, the limited nature of the static social graphs prohibits us from making a concrete insight of these designs in reality when considered for deployment settings in which social graphs exhibit a dynamic behavior [7, 15]. Such behavior greatly alters graphs structure, which is an essential determining factor of the performance of these designs on social networks.

In this paper we proceed further to understand dynamic social graphs for another family of applications; anonymous communication systems [14, 9, 4]. On the one hand, we extend and utilize earlier findings in [14] and [4] of using social graphs as mixers for anonymity. On the other hand, we improve on these results by formalizing the use of dynamic social structures for anonymity, and establishing a relationship between dynamic and weighted graphs. We show how our new design improves anonymous communication and stands against possible attacks by empowering a richer social structure. We validate our model using empirical studies on two dynamic social structures driven from real-world networks.

The rest of this paper is organized as follows. Preliminaries are outlined in §2, theoretical formalism is introduced in§3, and results and discussion in §4. We review the related work in §5, and draw concluding remarks in §6.

2. PRELIMINARIES

In this section we review preliminaries of the prior literature on the problem, which are required for understanding the rest of this paper. This known literature assumes a static graph. Unless otherwise is mentioned, this formalism follows from [14], which is to the best of our knowledge the first work that directly touches upon the problem (other literature work use the same model [9, 4]).

2.1 System Settings and Application Scenario

The idea of building mixers over social links is very simple. Im such model, users recruit their social social acquaintance to provide anonymity to their traffic. In a nutshell, each node (user) forwards her own traffic to her friends, and friends forward that traffic to their friends, and so on, for a certain number of hops, say ℓ. The number of hops ℓ used for forwarding the traffic is a system-wide parameter, which is determined by the security level desired in the system. For simplicity, and without losing generality, let n be the number of users in the system. Accordingly, the anonymity is defined for two parties; the sender and the receiver of traffic. For the receiver, the *anonymity set* is n, and the entropy of the probability distribution for a certain node being the sender is $H_s = \log_2(n)$. On the other hand, the anonymity of a sender is determined by

ASIA CCS'13, May 8–10, 2013, Hangzhou, China.

the probability distribution achieved after the fixed number of hops used in the system. Let the distribution of the final node selected in a *random walk* after ℓ hops starting from node v_j be π_j^ℓ, where $\pi_j^\ell = [\pi_{ij}^\ell]^{1\times n}$, then the anonymity of the sender of the traffic (at the last hop in the walk) is H_r, which is given as:

$$H_r = -\sum_{i=1}^{n} \pi_{ij}^\ell \log_2 \pi_{ij}^\ell \quad (1)$$

Using (1), we define the *anonymity set* A^ℓ as

$$A^\ell = 2^{H_r} \quad (2)$$

Every random walk on a graph with certain properties—see §2.3 for details—has a unique bounding or stationary distribution which captures the maximum achieved entropy.

In the rest of this paper, and to simplify the notation, we omit the index j, which is understood implicitly. Furthermore, the entropy and anonymity set for the sender v_j is obtained from the $j-$th column in the matrix P, after ℓ hops (multiplications).

2.2 Threat Model and Design Goals

In this paper we use the classical model of a colluding adversary with the capability of launching Byzantine attacks against the system built on top of social networks [9]. The adversary has the capability of logging end-to-end information, perform active attacks, and passive attacks. We also assume that the adversary has a limited capability of launching a Sybil attack in reality by inserting large number of Sybil identities [19]. In this work, however, we evaluate the performance of our system under an ideal setting where there is no Sybil identities are injected into the system.

As for the design goals, our design aims to provide a scalable and efficient solution, to provide natural incentives of participation, and to limit the attackers capabilities.

2.3 Formalization: The Case of Static Graphs

Let $G = (V, E)$ be an undirected and unweighted graph where $|V| = n$, $|E| = m$, $V = \{v_1, v_2, \ldots, v_n\}$, and $e_{ij} \in E$ iff $v_i \sim v_j \in V$. We define $\mathbf{A} = [a_{ij}]^{n\times n}$ as the adjacency matrix of G where $a_{ij} = 1$ iff $e_{ij} \in E$ or 0 otherwise. Define the Markov chain on the graph G following the transition matrix $\mathbf{P}$ which is defined according to $\mathbf{P} = [p_{ij}]^{n\times n}$ where:

$$p_{ij} = \begin{cases} \frac{1}{\deg(v_i)}, & v_i \sim v_j \\ 0 & \text{otherwise} \end{cases} \quad (3)$$

A unique stationary distribution is defined for the Markov chain over the transition probabilities defined above if the Markov chain is ergodic—requiring it to be both *irreducible* and *aperiodic* [13]. Theorem 1 states such distribution.

THEOREM 1. *(Stationary distribution on static graph) For an undirected and unweighted graph G, the stationary distribution of the Markov chain defined over G according to transitions in (3) is the probability vector, given as* $\pi = [\pi_i]^{1\times n}$*, where*

$$\pi_i = \deg(v_i)/2m \quad (4)$$

PROOF. The proof is a special case of the weighted graph case discussed in section 3 and follows from Theorem 2 □

Using the model in (1) and the distribution in (4), we define the maximal (in size) anonymity set following the same model as in (2) as $A^\infty = 2^{H_r^\infty}$, where:

$$H_r^\infty = -\sum_{i=1}^{n} (\frac{\deg(v_i)}{2m}) \log_2 (\frac{\deg(v_i)}{2m}) \quad (5)$$

2.4 Lower-bound of Anonymity

In [14] Nagaraja considered the average distribution achieved after ℓ hops from any potential source in the social graph as the anonymity achieved of every potential source. While this captures the average performance in the system, it simply does not show the worst case scenario observed at the lower-bound of the achieved anonymity for sender. Here, we revise Nagaraja's definition in [14] and outline a straightforward fix for the measure of the anonymity provided in a system that uses walks on the social graph.

Without losing generality, let ℓ be a system-wide parameter, which represents the number of hops from the source to the destination (or receiver) in the graph, and each node between them is chosen uniformly at random from its predecessor. For each source v_j (for $1 \le j \le n$), we define the probability distribution after ℓ hops as $\pi^\ell(v_j) = [\pi_i^\ell(v_j)]^{1\times n}$ for $(1 \le i \le n)$. The anonymity achieved in the system is bounded below by the entropy achieved in the probability distribution obtained by walking from the worst source in the graph:

$$H_r \ge \inf_{v_j} \{-\sum_{i=1}^{n} \pi_i^\ell(v_j) \log_2 \pi_i^\ell(v_j)\} \quad (6)$$

By extending (2) to the case in (6), we get the following

$$A^\ell = 2^{H_r} \ge 2^{\inf_{v_j}\{-\sum_{i=1}^{n} \pi_i^\ell(v_j) \log_2 \pi_i^\ell(v_j)\}} \quad (7)$$

The intuition of this lower bound is very simple, practical, and follows from the definition. Technically, this lower bound follows the classical theoretical trend in security: proving lower bounds of security (or anonymity as it is the case in hand) would enable us to guarantee, in the worst time, that our system would perform better than this bound for every user. On the other hand, considering the average case for achieved entropy might be very deceiving since many senders are not likely to achieve this average bound.

3. DYNAMIC GRAPHS FOR ANONYMITY

We extend findings in the literature on static graphs as mixers for anonymous communication to the case of the dynamic graphs. Such dynamic graphs arise naturally in many contexts due to social churn imposed by node and edge dynamics (joining and leaving social networks). It is worth noting that this is the first work of its own type to consider extending such results for building anonymous communication systems on top of dynamic social graphs.

3.1 Formalization: Dynamic Graphs

The dynamic graph is a simple generalization of the static graph used in literature. In particular, $G = \{G^{(i)}\}$ for $1 \le i \le t$ is a dynamic graph over t time periods. Let $G^{(i)} = (V^{(i)}, E^{(i)})$ for $1 \le i \le t$, where $|V^{(i)}| = n^{(i)}$ and $|E^{(i)}| = m^{(i)}$, be an unweighted and undirected graph (later we extend that to the weighted graph case). Let $V^{(i)} = \{v_1^{(i)}, v_2^{(i)}, \ldots, v_{n^{(i)}}^{(i)}\}$ and $E^{(i)}$ be the set of pairs of vertices $v_j^{(i)}$-$v_k^{(i)}$ if both nodes $v_j^{(i)}$ and $v_k^{(i)}$ in $V^{(i)}$ are connected to each other. For $G^{(i)}$, we define $\mathbf{A}^{(i)}$ where $\mathbf{A}^{(i)} = [a_{jk}^{(i)}]^{n^{(i)}\times n^{(i)}}$ (the superscription is used as part of the notation, and does not mean power), where:

$$a_{jk}^{(i)} = \begin{cases} 1 & v_j^{(i)} \sim v_k^{(i)} \in G^{(i)} \\ 0 & \text{otherwise} \end{cases}. \quad (8)$$

For the same graph $G^{(i)}$, we define the transition probability matrix

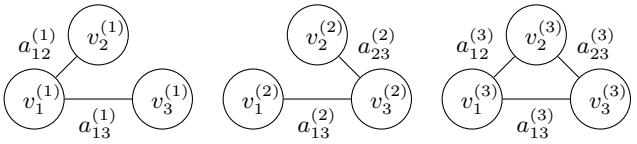

Figure 1: Simple example of dynamic graph. $a_{12}^{(i)} = w(v_1^{(i)}, v_2^{(i)})$ is as per the definition.

$\mathbf{P}^{(i)}$ such that $\mathbf{P}^{(i)} = [p_{jk}^{(i)}]^{n^{(i)} \times n^{(i)}}$, where:

$$p_{jk}^{(i)} = \begin{cases} 1/\deg\left(a_{jk}^{(i)}\right) & v_j^{(i)} \sim v_k^{(i)} \in G^{(i)} \\ 0 & \text{otherwise} \end{cases}. \tag{9}$$

Extending and generalizing (8) and (9) to the weighted case is easy if weights are given on edges in the graph. We define

$$a_{jk}^{(i)} = \begin{cases} w(v(i)_j, v(i)_k) & v_j^{(i)} \sim v_k^{(i)} \in G^{(i)} \\ 0 & \text{otherwise} \end{cases}, \tag{10}$$

where $w : E^{(i)} \to \mathbb{R}$ is a weight function that assigns real-valued weights to edges in $G^{(i)}$. Using (10), we define the degree of a node to be in terms of weights associated with edges for which that node is an end-vertex, as

$$\deg^w\left(v_j^{(i)}\right) = \sum_k w(v_j^{(i)}, v_k^{(i)}), \;\; v_j^{(i)} \sim v_k^{(i)} \in G^{(i)}. \tag{11}$$

Notice that (11) can also be written as $\deg^w\left(v_j^{(i)}\right) = \sum_k a_{jk}^{(i)}$—where $a_{jk}^{(i)}$ is defined in (10). Using (11), we can compute $\mathbf{P}^{(i)} = [p_{jk}^{(i)}]$ for weighted graphs, where

$$p_{jk}^{(i)} = \begin{cases} w(v_j^{(i)}, v_k^{(i)})/\deg^w\left(v_j^{(i)}\right) & v_k^{(i)} \sim v_k^{(i)} \in G^{(i)} \\ 0 & \text{otherwise} \end{cases}. \tag{12}$$

In a matrix form, $\mathbf{P}^{(i)}$ can be defined as $\mathbf{P}^{(i)} = (\mathbf{D}^{(i)})^{-1}\mathbf{A}^{(i)}$ where $\mathbf{D}^{(i)}$ is a diagonal matrix computed from $\mathbf{A}^{(i)}$, where the diagonal element $d_{jj}^{(i)}$ in $\mathbf{D}^{(i)}$ is the sum of ones in the j-th row in $\mathbf{A}^{(i)}$ (that is, the degree of node v_{ix} in $G^{(i)}$). At any time slot i, we define the bounding distribution of the Markov chain on the graph G_i as in literature defined $[\deg\left(v_j^{(i)}\right)/2m^{(i)}]$. It is, however, unclear how to proceed with the different snapshots of the same graphs at different times.

As shown in Figure 1, both nodes $v_1^{(1)}$ and $v_2^{(1)}$ are connected, but not with their future images—$v_1^{(2)}$ or $v_1^{(2)}$ and $v_2^{(2)}$ or $v_2^{(3)}$, respectively. This also applies to states in the future not connected to the past images. In the following, we investigate several techniques for modeling the dynamic social graph as a graph where transitions from future states to past states is possible. Techniques utilize here are generic, and can be used to any graph with multiple labels.

Prior work in the literature has tried to model dynamic graphs as *3-mode tensor* [1] or *union multigraph* [5]. However, while the first uses high dimensionality—making computations on the tensor computationally expensive, the second technique reduces dimensionality and loses some information about the graph. Indeed, the second technique computes the union between multiple graphs (edge- and node-wise) and omits any potential multiple edges between two nodes in the union. While this is meaningful to understand a union snapshot of multiple graphs, demonstrate connectivity characteristics of the union graph driven from multiple attributes, and potentially other benefits, it does not capture the "depth" of edges and does not differentiate between different edges based on their "real value". For example, while edges in the union

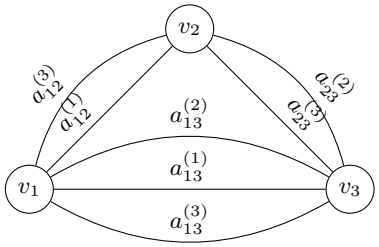

Figure 2: Simple example of converting a dynamic graph into multigraph by collapsing all images of a node to the node itself.

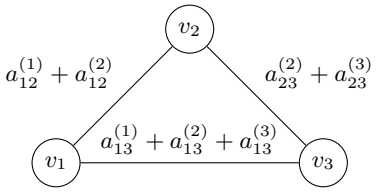

Figure 3: An example of multigraph conversion into weighted graph by summing weights of edges between pairs of nodes.

multigraph are all the same, some in reality might be the result of multiple edges whereas others could be the result of a single edge.

3.2 Dynamic Graphs as Multigraphs

Formally, for the dynamic graph $G = \{G^{(i)}\}$ described in section 3.1, we define a multigraph $\mathbb{G}$ as $\mathbb{G} = (\mathbb{V}, \mathbb{E})$, where

$$\mathbb{V} = \bigcup_{i=1\ldots t} \{V^{(i)}\}, \text{ and } \mathbb{E} = \biguplus_{i=1\ldots t} \{E^{(i)}\}. \tag{13}$$

Notice that $\cup$ is a *set union*, which does not allow repetition of vertices, whereas $\uplus$ is a *multiset union*, which allows edge repetition. When $\mathbb{E}$ is computed, the index that corresponds to the time of the edges in E_i can be removed for simplicity. A simple toy example of transforming the multiple snapshots of the dynamic graph in Figure 1 into a multigraph is in Figure 2.

Our formalization above of the graph as a multigraph (rather than union multigraph as per the way defined in [5]) follows the intuition of what a dynamic graph could yield of associations at any time. At a time i, where $1 \leq i \leq t$, constructing the proper graph for operating a potential system, like mixing-based anonymous communication system, and maintaining the same information driven from the original multiple snapshots of the graph is possible.

3.3 Dynamic Graphs as Weighted Graphs

Now, we convert the dynamic graph model represented as a multigraph, as in (13), into a weighted graph. We generalize formalizations in section 3.1. In particular, the model in (10) can be rewritten (for weighted undirected graph) as $\mathbf{A} = [a_{jk}]^{n \times n}$—here, $n = |\mathbb{V}|$—where

$$a_{jk} = \sum_{i=1\ldots t} w(v_j^{(i)}, v_k^{(i)}), \;\; v_j^{(i)} \sim v_k^{(i)} \in G^{(i)} \forall i. \tag{14}$$

Similarly, we extend the model in (11) into

$$\deg^w\left(v_j\right) = \sum_{i=1\ldots t} \deg^w\left(v_j^{(i)}\right) = \sum_{\forall k} a_k^{(j)} \tag{15}$$

$$= \sum_{\forall k} \sum_{i=1\ldots t} w(v_j^{(i)}, v_k^{(i)}), v_j^{(i)} \sim v_k^{(i)} \in G^{(i)}. \tag{16}$$

We can further extend the transition probability formulation to cover the weighted graph by plugging both (14) and (15) into a similar

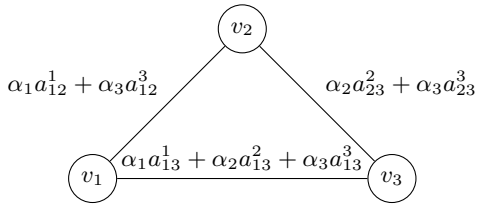

Figure 4: A weighted graph model to express dynamic graphs.

model to that of (12), to get $\mathbf{P} = [p_{jk}]^{n \times n}$, where

$$p_{jk} = a_{jk} / \deg^w(v_j) \tag{17}$$

For a random walk defined on $\mathbb{G}$ according to the transition probability defined in (17), the following theorem states the stationary distribution. This theorem (and the proof herein) are essential for latter results on characterizing and operating on dynamic graphs. Also, the proof of Theorem 1 follows similarly as in below.

THEOREM 2. *Let $\mathbb{G} = (\mathbb{V}, \mathbb{E})$ be a* connected*,* undirected*, and* weighted *graph defined as in (13). For a random walk following transition probabilities as in (17), the stationary distribution is defined as $\pi = [\pi_i]^{1 \times n}$ (for $n = |\mathbb{V}|$), where:*

$$\pi_i = \deg^w(v_i) / \sum_{k=1 \ldots n} \deg^w(v_k) \tag{18}$$

PROOF. See the appendix. The proof is also in [2]. □

3.4 Generalized weighted graphs

In many natural social contexts, recent associations are more valued than older ones, or vice-versa. Accordingly, a general framework for quantifying the potential of any system on top of social networks should consider implicit social network characteristics, such as link age, in addition to the explicit differences among links captured by the topological structure. We generalize the model in section 3.3 to accommodate for implicit values of associations over time. Without losing generality, let α_i (for $1 \leq i \leq t$) be a set of parameters that take numerical values. An extension of the social graph model in (14) is as follows:

$$a_{jk} = \sum_{i=1 \ldots t} \alpha_i w(v_j^{(i)}, v_k^{(i)}),\ v_j^{(i)} \sim v_k^{(i)} \in G^{(i)} \forall i. \tag{19}$$

The rest of the model in section 3.3, particularly in (15) onward, holds for this generalization after adjusting a_{jk} as in (19). A toy example demonstrating the adjustment of weights in Figure 3 is shown in Figure 4.

4. RESULTS

4.1 Datasets and Data Preprocessing

Our sources of data are the Facebook social network dataset [18] and the DBLP [8] co-authorship graph, which are explained below.

The DBLP Dataset. The DBLP dataset represents co-authorship graph, where nodes are authors and a link between two authors implies that the authors have co-authored a paper. The original DBLP dataset consists of $943,316$ nodes and $6,379,554$ edges between them, for publication records until May 2011 in computer science areas. The largest connected component consists of $769,642$ and $3,051,127$ undirected edges. To generate dynamic graphs from that component, we select the period of 2006 to 2010 inclusive, by selecting authors who have publications in each and every of these years. The result is a multigraph where two nodes would have an edge if they co-authored a paper in a given year, and number of such papers per year is used as weight for weighted graphs. Multiple edges could be created between two authors if they co-authored over multiple years. Multiple edges are labeled with respect to the year of publication. The final multigraph has $46,994$ nodes and $458,736$ edges. We decompose each multigraph to multiple-graphs with respect to the edge label. Finally, as some nodes who published in the given period could be isolated in a certain year, we remove these nodes so as each resulting graph is connected. Statistics of the different resulting graphs are shown in Table 1. For our study, we consider several cases of the same graph including both weighted and unweighted, with respect to the time.

Table 1: Statistics of DBLP time-varying graphs. Metrics of comparison are number of nodes (n), number of edges (m), average clustering coefficient, diameter, and radius.

	nodes	edges	clustering	diameter	radius
DBLP (1)	31704	71994	0.483	26	14
DBLP (2)	33012	79475	0.480	27	14
DBLP (3)	33923	84125	0.467	24	13
DBLP (4)	33071	82282	0.453	23	12
DBLP (5)	26150	62161	0.419	24	13

Graphs used in our experiments are as follows. (i) Unweighted graph with respect to each year (5 graphs). (ii) Unweighted single graph representing the entire dataset (1 graph). (iii) Weighted single graph representing the entire dataset. The weight on an edge connecting two nodes is the sum of all weights of edges between these nodes over time from the beginning to the end of recording the graph structure. (iv) Weighted multiple snapshot graphs (up to each year; 5); a graph G_{1i} combines all nodes in G_1 to G_i and edges between them. (v) Unweighted multiple snapshot (up to each year; 5). These graphs are obtained using the same method as in the previous step but without weights. (vi) Weighted graphs with weights assigned based on the age of the link. We use geometrical (2^{1-x}; $x = 1$ newest) and reciprocal ($1/x$) decay distributions.

In Table 1 and Table 2, the basic structural properties are as follows. (i) Graph size: the number of nodes and number of edges in the social graph (denoted as n and m). (ii) Clustering coefficient: is the average (thus in [0, 1]) of local clustering coefficient for all nodes. The local clustering coefficient for a node is the fraction of possible triangles that go through that node. (iii) Diameter: the longest of eccentricities among all nodes in the graph. The eccentricity of a node is defined as the longest shortest path from that node to other nodes in the graph. (iv) Radius: shortest eccentricity.

The Facebook Dataset. The Facebook dataset [18] is for wall posts in New Orleans regional network from 2004 to 2009. A link between two nodes indicates that the first node has interacted with the second node. Further details on statistics of the entire dataset is in [18]. To obtain a dynamic graph from this dataset, we limit ourselves to the last 30 months of interactions, with each graph obtained over 6 months of interaction. The resulting five graphs are shown in Table 2. The same variations used above are also used for Facebook.; we omit details for the lack of space.

4.2 Results

In the following we outline the results of utilizing the different social graphs obtained in section 4.1, using the techniques described earlier in this work. Our main measurement metric is the achieved anonymity in terms of the total entropy in the distribution of the last hop in a random walk, as the length of the random walk increases; computed as in (1). We keep in mind that potential utilization of social graphs for anonymity systems would be sub-

Table 2: Statistics of Facebook time-varying graphs. Metrics of comparison are number of nodes (n), number of edges (m), average clustering coefficient, diameter, and radius.

	nodes	edges	clustering	dimeter	radius
Facebook (1)	9154	23245	0.102	19	10
Facebook (2)	13288	37908	0.101	18	10
Facebook (3)	16540	42427	0.092	19	10
Facebook (4)	23879	59190	0.085	21	11
Facebook (5)	35665	86525	0.084	18	10

ject to the performance of this systems, which necessitate to a short random walk length. We consider walk lengths varying from 1 to 20 steps, where walk lengths of $1 - 12$ are demonstrated in most experiments. As the entropy varies depending on the source of the random walk, we are interested in the maximum, minimum (advocated in section 2.4 and expressed in (6)), and mean entropies for a given dataset as the walk length increases.

Original Unweighted Graphs. We first consider operating the anonymity system on top of the graphs shown in Table 1 and Table 2. First, we observe that lower-bound on the achieved entropy or the entire system is much smaller than that of the average and maximum entropy, for any walk length. This tells that the measure of the lower-bound on the entropy, while theoretically appealing for the guarantees advocated earlier, do not provide a representative measure for the whole set of nodes in the system or graph. Second, we observe that both the mean and the maximum of the entropy in each of the graphs stabilizes, and reaches its potential maximum entropy within a relatively smaller number of steps, corresponding to shorter random walk length. This indeed interesting, and agrees with prior work in [14], despite that the results in the prior work have been on relatively a faster mixing social graph [13].

Dynamics as Weights. We consider modeling dynamics of social graphs as weights on edges. We use the method in 3.3 for generating these graph with weights. The method of obtaining the graphs is explained in 4.1. Similar observations on the results for the tendency of weights is made as on the previous measurements. Furthermore, the *general* tendency of improvement of the entropy value as the time goes is made clear in both measurements.

Unweighted Dynamic Graphs. We consider removing weights from the different dynamic graphs to observe how this affects the entropy as walks length increases. For the same experiment above, with the only difference being deletion of weights. The most important point made clear in this experiment is that unweighted graph generated from these weighted dynamic graphs provide higher entropy for the same walk length (over the same number of nodes).

Different Weighting Scenarios. We consider the potential ways of assigning the different weights on the social graph, based on the age of the link, and how this impacts the achieved anonymity on these graphs. We consider the graphs constructed from the multiple-snapshots, according to the way described in section 3.3 and section 3.4. We use the result in section 3.3 to generate a "linear" weighting factor (the coefficient is 1), and thus the weight of an edge is the number of interactions between the two nodes over all years. We consider the same graph generated in this step without weights as well. Finally, we use the model in section 3.4 to generate generalized weighted graphs, where weights are formed according to the reciprocal or geometrical decay distributions explained in 4.1. In these measurements, and somewhat counter-intuitively, we observe that the unweighted graph model results in best entropy (in all three categories: min, mean, and max)—Further details are in the discussion section. On the other hand, we also observe that in all of these graphs, the achieved anonymity is good enough (as a portion of the maximal) even with a walk length of 10, suggesting the usefulness of this design. Both remarks apply to both datasets, though bias is a lot higher in Facebook than in DBLP.

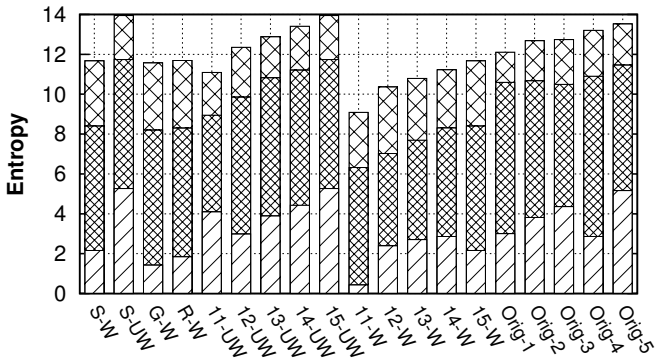

Figure 5: Average entropy for walk length of 10; all Facebook graphs (S. is single, G and R are geometrical reciprocal distributions of weights, W is weighted, and UW is unweighted, and numbers are to indicate which graph is used: 1-5 are original graphs whereas 11-15 are the dynamic graph model).

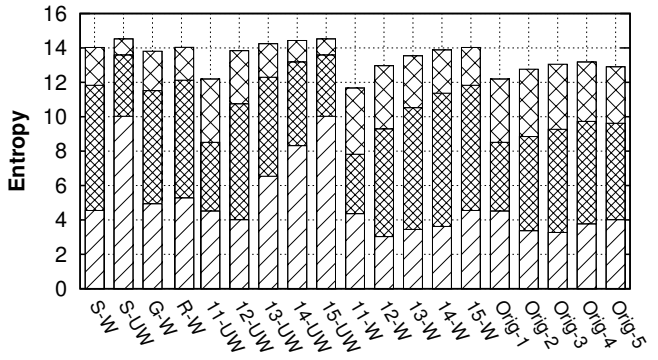

Figure 6: Average entropy for walk length of 10; all DBLP graphs (S. is single, G and R are geometrical reciprocal distributions of weights, W is weighted, and UW is unweighted, and numbers are to indicate which graph is used: 1-5 are original graphs whereas 11-15 are the dynamic graph model).

Combining All Scenarios. We consider all scenarios mentioned above, for each of the datasets we had, and for a fixed random walk length to compare them relatively and draw final conclusions on the impact of the underlying social structure on the achieved anonymity. We consider the random walk length $\ell = 10$, and experiment for both datasets to compute the entropy—both mean and max. The results are shown in Figure 5 through Figure 6.

4.3 Analysis and Discussion

In most of the measurements of the entropy on the distribution of the random walk after ℓ hops, we observe a relatively good entropy which supports the claimed efficiency advocated in this work and [14]. However, this entropy, for example is not as high in some graphs, especially those the single snapshots that consider the graph at one time period, and those resulting from assigning weights on the graphs corresponding to the richness of the edges. This pattern is shown in both datasets, which call for explanations.

One potential explanation of the relative difference between the achieved entropy in the individual graphs and that obtained from

the graphs computed using our dynamic graph model is the inherit increment in the size of the resulting final graph. For example, while the largest DBLP graph is about 36,000 nodes, the final graph of the DBLP after using our model that considers graph dynamics would result into about 44,000 nodes in the largest connected components. This, and the fact that the graph becomes richer of more edges that connect multiple components in the graph, improves the mixing characteristics of the graph, which ultimately improves the achieved entropy as ℓ increases.

This does not explain the difference between the achieved entropy in both weighted and unweighted graphs, even for the graph with the same number of nodes and edges. For example, when $\ell = 10$, the achieved entropy on DBLP-15 when weighted is 14 while it's 14.5 for the same graph when it is unweighted—while the difference in entropy is small, i.e., 0.5, the 14.5 bits of entropy provide about $23,170$ anonymity set whereas only $16,384$ are provided for the other case, which translates to more than $6,786$ of difference in anonymity set. One possible explanation of this behavior is the intuitive meaning of weighting graphs: by assigning weights on edges, we are biasing the random walk on such graphs and favoring a node over another of being reaching at any time when running the random walk. This is, some nodes are more likely reached whereas other nodes while less likely reaching by the random walk which definitely decreases the potential set of nodes being used as a last hop in the random walk. This intuitive meaning explains the difference in the entropy in both cases.

Unexpectedly, both entropy and anonymity sets are greatly decreased when using the weighted graphs that model dynamic structure. One possible explanation is that these weights are obtained by favoring some edges over others, which is more meaningful from an anonymity point of view, whereas edges in the unweighted graph simply make all relationships over time equal. In a realistic scenario, where potential insider attacker could exist to penetrate the system to get communicated messages the social overlay, the model, which considers links to be equal independent of their history or time of creation, could be problematic. Given this intuitive explanation of weights associated with edges, one would anticipate the use of weighted graphs in real-world scenarios despite this degradation in the achieved entropy and anonymity set sizes given their potential for minimizing harms due to edge infiltration.

5. RELATED WORK

Exploiting static social networks for anonymous communication has been explored in [14, 9, 4], some of which has been discussed earlier, and all did not consider the dynamic graph case. Modeling of dynamic social graph and extracting a definition for the mixing time is done in [1], whereas sampling multigraph defined as node set union graphs is done in [5]. To the best of our knowledge, no prior work considered dynamic graphs in the context of the problem in hand and as per our method. On the other hand, other assumptions in social network-based systems, like binary trust [10], edge directionality [12], and expansion properties [11] are previously consider and challenged in separate studies.

Dynamic social graphs have been studied in [7, 17, 15, 16]. Most of these studies, however, considered mining known simple properties of social graphs, but not the mixing time and patterns used for anonymity. Finally, observing dynamics of social networks as set of static graphs over time has been most recently used in [20].

6. CONCLUDING REMARKS

In this paper we considered the problem of building anonymous communication systems on (unstructured) dynamic social graphs. We have pointed out an interesting relationship between dynamic structures and weighted graphs, and formalized the anonymity achieved under dynamics as a random walk on weighted graphs. We formulated the problem in hand, and shown the bounding distribution, which captures the maximal achieved entropy of a random walk on an anonymous communication system, which uses these dynamic structures. Through experiments on real-world datasets, we have shown the potential of these dynamic structures, and despite their numerical disadvantage over unweighted versions, we have pointed out their benefits for anonymity for that they capture more meaningful structure that represents stronger ties.

This work has considered "unstructured" social graphs, which a non-constant (and likely power-law) degree distribution. On one hand, the potential of structured graphs for anonymity is well studied, and beautiful theoretical results are already provided. On the other hand, suitability of these results for real-world social structures, especially when considering dynamics, is unclear. In the future, we will look at creating structured graphs from unstructured social graphs, and explore their potential for anonymity systems.

Acknowledgement. Yongdae Kim was supported by the KCC (Korea Communications Commission), Korea, under the R&D program supervised by the KCA (Korea Communications Agency), KCA- 2013-12911-05003. Part of this work was done while A. Mohaisen was at the University of Minnesota.

7. REFERENCES

[1] U. Acer, P. Drineas, and A. Abouzeid. Random walks in time-graphs. In *WMON*. ACM, 2010.
[2] B. Bollobás. *Modern graph theory*, volume 184. Springer, 1998.
[3] E. M. Daly and M. Haahr. Social network analysis for routing in disconnected delay-tolerant manets. In *MobiHoc*, 2007.
[4] G. Danezis, C. Díaz, C. Troncoso, and B. Laurie. Drac: An architecture for anonymous low-volume communications. In *PETS*, 2010.
[5] M. Gjoka, C. T. Butts, M. Kurant, and A. Markopoulou. Multigraph sampling of online social networks. *IEEE JSAC*, 2011.
[6] P. Hui, J. Crowcroft, and E. Yoneki. Bubble rap: Social-based forwarding in delay-tolerant networks. *IEEE TMC*, 2011.
[7] R. Kumar, J. Novak, and A. Tomkins. Structure and evolution of online social networks. *Link Mining: Algorithms and Apps*, 2010.
[8] M. Ley. The DBLP computer science bibliography: Evolution, research issues, perspectives. In *SPIR*, 2009.
[9] P. Mittal, M. Wright, and N. Borisov. Pisces: Anonymous communication using social networks. In *NDSS*, 2013.
[10] A. Mohaisen, N. Hopper, and Y. Kim. Incorporating trust into social network-based sybil defenses. In *INFOCOM*, 2011.
[11] A. Mohaisen, H. Tran, N. Hopper, and Y. Kim. Understanding social network properties for trustworthy computing. In *SIMPLEX*, 2011.
[12] A. Mohaisen, H. Tran, N. Hopper, and Y. Kim. On the mixing time of directed social graphs and security implications. In *ASIACCS*, 2012.
[13] A. Mohaisen, A. Yun, and Y. Kim. Measuring the mixing time of social graphs. In *IMC*, pages 383–389. ACM, 2010.
[14] S. Nagaraja. Anonymity in the wild. In *PETS*, 2007.
[15] L. Tang, H. Liu, J. Zhang, and Z. Nazeri. Community evolution in dynamic multi-mode networks. In *KDD*, 2008.
[16] C. Tantipathananandh, T. Berger-Wolf, and D. Kempe. A framework for community ident. in dynamic social networks. In *KDD*, 2007.
[17] H. Tong, S. Papadimitriou, J. Sun, P. Yu, and C. Faloutsos. Colibri: fast mining of large static and dynamic graphs. In *KDD*, 2008.
[18] B. Viswanath, A. Mislove, M. Cha, and K. P. Gummadi. On the evolution of user interaction in facebook. In *WOSN*, 2009.
[19] H. Yu, M. Kaminsky, P. Gibbons, and A. Flaxman. SybilGuard: defending against sybil attacks via social networks. In *SIGCOMM*, 2006.
[20] X. Zhao, A. Sala, C. Wilson, X. Wang, S. Gaito, H. Zheng, and B. Y. Zhao. Multi-scale dynamics in a massive online social network. In *IMC*, 2012.