Efficiently Estimating Motif Statistics of Large Networks

Pinghui Wang, Huawei Noah's Ark Lab
John C.S. Lui, The Chinese University of Hong Kong
Bruno Ribeiro, Carnegie Mellon University
Don Towsley, University of Massachusetts Amherst
Junzhou Zhao, Xi'an Jiaotong University
Xiaohong Guan, Xi'an Jiaotong University

Exploring statistics of locally connected subgraph patterns (also known as network motifs) has helped researchers better understand the structure and function of biological and online social networks (OSNs). Nowadays the massive size of some critical networks – often stored in already overloaded relational databases – effectively limits the rate at which nodes and edges can be explored, making it a challenge to accurately discover subgraph statistics. In this work, we propose *sampling methods* to accurately estimate subgraph statistics from as few queried nodes as possible. We present sampling algorithms that efficiently and accurately estimate subgraph properties of massive networks. Our algorithms require no pre-computation or complete network topology information. At the same time, we provide theoretical guarantees of convergence. We perform experiments using widely known data sets, and show that for the same accuracy, our algorithms require an order of magnitude less queries (samples) than the current state-of-the-art algorithms.

1. INTRODUCTION

Understanding the structure and function of complex systems is of wide interest across many fields of science and technology, from sociology to physics and biology. Networks with similar topological features such as degree distribution or graph diameter can exhibit significantly different local structure. Thus, there is much interest in exploring small connected subgraph patterns in networks, which are often shaped during their growth and have been used to characterize communication and evolution patterns in OSNs [Chun et al. 2008; Kunegis et al. 2009; Zhao et al. 2011; Ugander et al. 2013]. For example, simple 3-node subgraph classes such as "the enemy of my enemy is my

friend" and "the friend of my friend is my friend" are well known evolution patterns in social networks. Kunegis et al. [Kunegis et al. 2009] considered the significance of these subgraph patterns in Slashdot Zoo[1] evaluating the stability of signed friend/foe subgraphs. Other examples include counting relative frequencies of closed triangle (i.e., three users connected with each other), which are probably more prevalent in Facebook than Twitter, since Twitter serves more as a news media than an OSN service [Kwak et al. 2010]. More complex examples of k-node subgraph frequencies, $k > 3$, include Milo et al. [Milo et al. 2002] defined network motifs (or local subgraph patterns) as small subgraph classes occurring in networks at numbers that are significantly larger than found in random networks. Network motifs have been used for pattern recognition in gene expression profiling [Shen-Orr et al. 2002], protein-protein interaction predication [Albert and Albert 2004], and coarse-grained topology generation [Itzkovitz et al. 2005].

Unfortunately, characterizing the frequencies of subgraph patterns by searching and counting subgraphs is computationally intensive since the number of possible k-node combinations in the original graph increases exponentially with k. To address this problem, Kashtan et al. [Kashtan et al. 2004] proposes to sample subgraphs using random edge sampling but this method scales poorly with the subgraph size and the results can be heavily biased. Wernicke [Wernicke 2006] proposes another approach (FANMOD) based on enumerating subgraph trees. The latter relies on random node sampling, which is either not supported by most OSN APIs or is too resource intensive to be practical (with respect to cache misses and vacant user ID space [Ribeiro et al. 2012]).

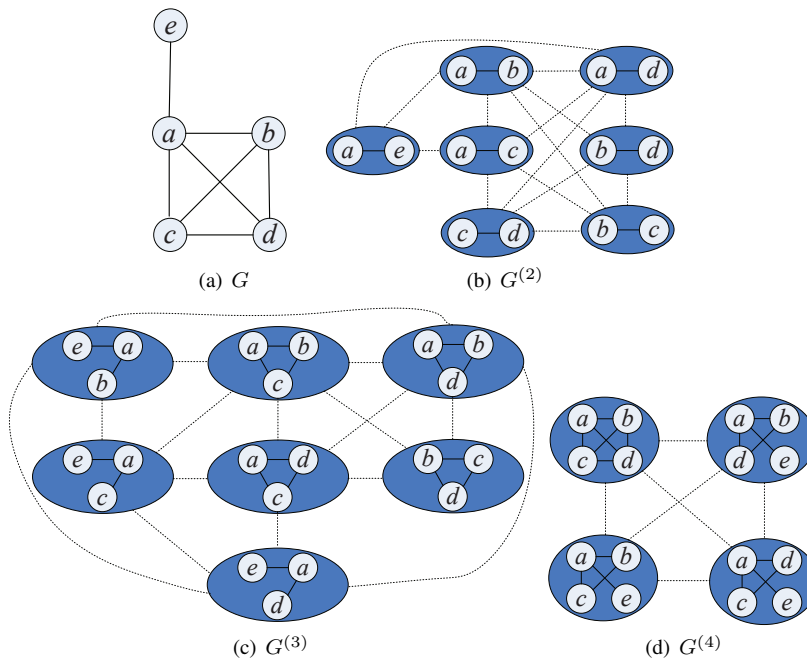Fig. 1. An example of G, and CIS relationship graphs $G^{(2)}$, $G^{(3)}$, and $G^{(4)}$.

Thus, it is paramount for such algorithms to query the graph on-the-fly without knowledge of the complete topology. Moreover, such algorithms should output accurate high accuracy estimates of subgraph concentrations with as few queries as possible. Recently, Bhuiyan et al. [Bhuiyan et al.

---

[1] www.slashdot.org

2012] proposed a Metropolis-Hastings-based algorithm (henceforth denoted GUISE) that jointly estimates concentrations of 3-node, 4-node, and 5-node connected induced subgraphs (CISes). However, the rejection sampling procedure of the Metropolis-Hastings random walk (MHRW) used in GUISE has received much criticism lately in the precise context of graph sampling [Gjoka et al. 2011; Ribeiro and Towsley 2012]. Rejecting a sample incurs the *cost of sampling* without gathering information in exchange. And, information-wise, the rejected samples may contain more information about the statistic of interest than the accepted samples. Recently, Ribeiro and Towsley [Ribeiro and Towsley 2012] shows that MHRW rejects information-rich samples when trying to estimate the degree distribution of a graph. The end result is a sampling method that exhibits large estimation errors. A new estimation method that does not suffer from the above mentioned problems is needed.

In this work we propose two algorithms to accurately estimate subgraph concentrations. The first one, denoted PSRW, significantly improves upon GUISE in two fronts: (a) PSRW can estimate statistics of CISes of any size, in contrast to GUISE that is limited to jointly estimating 3-node, 4-node, and 5-node CISes; and (b) through careful design PSRW does not reject samples, making the estimation errors of PSRW significantly lower than those of GUISE. Most importantly, PSRW is not an incremental improvement over GUISE but rather a different type of random walk that is designed without the need to reject samples, using the Horvitz-Thompson estimator [Ribeiro and Towsley 2010] to unbias the observations. The second algorithm we propose, denoted Mix Subgraph Sampling or MSS, can jointly estimate CISes of sizes $k - 1$, k, and $k + 1$ for any $k \geq 4$, not only generalizing GUISE (GUISE 3-node, 4-node, and 5-node CISes is the special case $k = 4$) but also achieving lower estimation errors. One of the main differences between GUISE and PSRW or MSS is that our random walk is designed to sample nodes that are important for the CIS estimation and use *all of the gathered samples* in the estimation phase.

Through simulations we show that both of our methods (PSRW and MSS) are significantly more accurate than GUISE for either the same number queried nodes or the same walk clock time (using a modern computer). The walk clock time is measured under the assumptions of access to a local database or a remote database (assuming 100 milliseconds of query response delay). Our methods represent the network as a *CIS relationship graph*, whose nodes are *connected and induced subgraphs* (CISes) of the original network. Fig. 1 illustrates a CIS relationship graph for subgraphs of two, three, and four node subgraphs. Our algorithms consist of running a random walk (RW) on the CIS relationship graph. Besides its accuracy, our algorithms are lightweight. They require little memory (more precisely, $O(k^2 + B)$ space where k is the subgraph size and B is the number of queried nodes) and, more importantly, *significantly fewer queries than the state-of-the-art methods to achieve the same accuracy*. Note that building the completely CIS relationship graph is prohibitively expensive, both in terms of queries and memory. Thus, our RW methods *do not require* the CIS relationship graph and there is no need to know the complete graph topology in advance, only the parts of the network already queried. We also prove that a RW on the CIS relationship graph achieves asymptotically unbiased concentration estimates of the distinct subgraphs on the original network.

This paper is organized as follows. The problem formulation is presented in Section 2. Section 3 presents methods for estimating subgraph class concentrations. The performance evaluation and testing results are presented in Section 4. Section 5 presents applications of our methods to two real OSN websites. Section 6 summarizes related work. Concluding remarks then follow.

2. PROBLEM FORMULATION

Let $G = (V, E, L)$ be a labeled undirected graph where V is the set of nodes, E be a set of pairs of V (edges), and L is a set of labels $l_{i,j}$ associated with edges $(i, j) \in E$. If G represents a directed network, then we attach a label to each edge that indicates the direction of the edge ($\rightarrow, \leftarrow, \leftrightarrow$). Edges may have other labels too, for instance, in a signed network, edges have positive or negative labels.

An induced subgraph of G, $G' = (V', E', L')$, $V' \subset V$, $E' \subset E$ and $L' \subset L$, is a subgraph whose edges and associated labels are all in G, i.e. $E' = \{(i, j) : i, j \in V', (i, j) \in E\}$, $L' = \{l_{i,j} :$

$i, j \in V', (i, j) \in E$}. We define $C^{(k)}$ as the set of all connected and induced subgraphs (CISes) with k nodes in G. Then we partition $C^{(k)}$ into T_k equivalence classes $C_1^{(k)}, \ldots, C_{T_k}^{(k)}$ where CISes within $C_i^{(k)}$ are isomorphic and any pair of CISes from $C_i^{(k)}, C_j^{(k)}, i \neq j$ are not isomorphic. To illustrate our notation, in what follows we present some simple examples. Fig. 2 (a) shows all three-node motifs of any unlabeled undirected network. When G is an unlabeled undirected network, then the number of three-node motifs is $T_3 = 2$, and $C_1^{(3)}$ and $C_2^{(3)}$ are the sets of CISes in G isomorphic to motifs 1 and 2 in Fig. 2 (a) respectively. Fig. 2 (b) shows all four-node motifs of any unlabeled undirected network, in this case $T_4 = 6$. Fig. 3 shows all three-node motifs of any directed network, in this case $T_3 = 13$. Fig. 4 shows all motifs when G is of any signed network, in this case $T_3 = 7$.
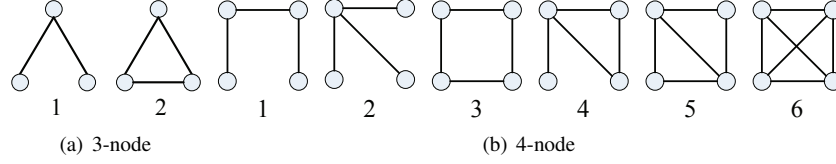
Fig. 2. All classes of three-node and four-node undirected and connected subgraphs (The numbers are the subgraph class IDs).
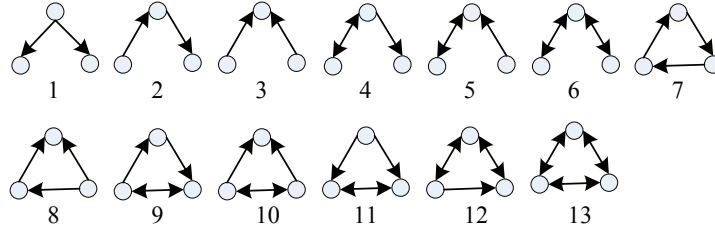
Fig. 3. All classes of three-node directed and connected subgraphs.
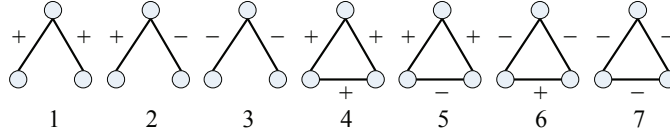
Fig. 4. All classes of three-node signed and undirected subgraphs.

The concentration of $C_i^{(k)}$ is

$$\omega_i^{(k)} = \frac{|C_i^{(k)}|}{|C^{(k)}|}, \qquad 1 \leq i \leq T_k,$$

where $|C_i^{(k)}|$ is the number of CISes in $C_i^{(k)}$. For example, we have $|C^{(3)}| = 7$ for G in Fig. 1, and $C_1^{(3)}$ includes three CISes: 1) CIS consisting of a, b, and e; 2) CIS consisting of a, c, and e; 3) CIS consisting of a, d, and e. Then we have $\omega_1^{(3)} = 3/7$. In this work we are interested in accurately estimating $\omega_i^{(k)}$ by querying a small number of nodes. Note that the network topology is unknown

to us and we are only given one initial connected subgraph of size $k > 1$ in G to bootstrap our algorithm.

3. CONNECTED AND INDUCED SUBGRAPH SAMPLING METHODS

In this section we first introduce the notion of a "*CIS relationship graph*". Then we propose two subgraph sampling methods based on random walks (RWs) on CIS relationship graphs to estimate the concentrations of subgraph classes of a specific size k. Finally, we propose a sampling method to solve the problem of measuring the concentrations of subgraph classes of sizes $k - 1$, k, and $k + 1$ simultaneously, where the special case $k = 4$ is equivalent to the problem studied in Bhuiyan et al. [Bhuiyan et al. 2012]. A list of notations used is shown in Table I.

Table I. Table of notations

$G = (V, E, L)$	graph under study
$d(v), v \in V$	degree of node v in G
$G^{(k)} = (C^{(k)}, R^{(k)})$	k-node CIS relationship graph
$V(s), s \in C^{(k)}$	set of nodes for the k-node CIS s
$E(s), s \in C^{(k)}$	set of edges for the k-node CIS s
$N(s), s \in C^{(k)}$	$N(s) \subset V$, set of nodes in $V \backslash V(s)$ which are connected to nodes in the k-node CIS s
$X(s), s \in C^{(k)}$	$X(s) \subset C^{(k)}$, neighbors of k-node CIS s in graph $G^{(k)}$
$d^{(k)}(s), s \in C^{(k)}$	degree of the k-node CIS s in $G^{(k)}$
$C(s)$	subgraph class of the CIS s
$C_i^{(k)}$	the i-th k-node subgraph class in G
T_k	number of k-node subgraph classes
$\omega_i^{(k)}$	concentration of subgraph class $C_i^{(k)}$
$I^{(k)}(x), x \in C^{(k+1)}$	number of k-node CISes contained in $(k+1)$-node CIS x
$C^{(k-1)}(s), s \in C^{(k)}$	the set of $(k-1)$-node CISes contained in the CIS s
$O^{(k)}(s'), s' \in C^{(k-1)}$	the set of k-node CISes that contain the CIS s'
B	number of sampled CISes
B^*	number of queries

3.1. CIS relationship graph

Let $C^{(k)}$ ($2 \le k < |V|$) denote the set of all k-node CISes in G. Two different k-node CISes s_1 and s_2 in $C^{(k)}$ are connected if and only if they have exactly $k-1$ nodes in common. Formally, the undirected graph $G^{(k)} = (C^{(k)}, R^{(k)})$ represents the *CIS relationships* between all k-node CISes in G, where $C^{(k)}$ and $R^{(k)}$ are the node and edge sets for graph $G^{(k)}$ respectively. When two k-node CISes s_i and s_j in $C^{(k)}$ differ in one and only one node, there exists an edge (s_i, s_j) in graph $G^{(k)}$. We say that two k-node CISes s_i and s_j are reachable if and only if there is at least one path between them in graph $G^{(k)}$, and $G^{(k)}$ is connected if and only if every pair of subgraphs in $C^{(k)}$ is reachable. Fig. 1 shows an example of an original unlabeled graph G and its associated CIS graphs $G^{(2)}$, $G^{(3)}$, and $G^{(4)}$. Then we have the following theorems.

THEOREM 3.1. *If the graph G is connected, then the k-node CIS graph $G^{(k)}$ is connected, $2 \le k < |V|$.*   □

THEOREM 3.2. *If the graph G is connected and either non-bipartite, or contains a node with degree larger than two, all k-node CIS graphs $G^{(k)}$ are non-bipartite, where $2 \le k < |V|$.*   □

The proofs of all Theorems in this section are included in the Appendix for completeness.
Remark: Theorems 3.2 states that $G^{(k)}$ is non-bipartite for most connected graph G. Connectedness is critical for removing biases from RW sampling of $G^{(k)}$. Biases introduced through sampling a bipartite graph using a lazy RW are easily removed. Biases introduced though sampling via a classical RW can only be removed if the graph is non-bipartite.

3.2. Subgraph random walk (SRW)

In this subsection, we propose a sampling method, *subgraph random walk* (SRW), and apply it over graph $G^{(k)}$, $2 \le k < |V|$ to estimate concentrations of subgraph classes. SRW can be viewed as a regular RW over graph $G^{(k)}$. We define $X(s) \subset C^{(k)}$ as the set of neighbors of the current k-node CIS s sampled in $G^{(k)}$. From an initial k-node CIS, SRW randomly selects a CIS from $X(s)$ as the next-hop CIS. SRW moves to this neighbor CIS and repeats the process. Clearly $X(s)$ cannot be directly obtained since $G^{(k)}$ is not available in advance. In what follows we first discuss the number of queries required to compute $X(s)$ at each step. We observe that $X(s)$ can be obtained using at most k queries, which makes SRW both practical and efficient. We then show in detail how to apply SRW to sample k-node CISes from the original graph G in detail. Finally we present our method for estimating concentrations of subgraph classes based on sampled CISes.

First, we present a critical observation for analyzing the performance of the SRW: *Querying nodes in a k-node CIS s is suffices to obtain $X(s)$, i.e., the neighbors of s in $G^{(k)}$.* Denote by $V(s)$ the set of nodes in s and $E(s)$ the set of edges in s. Denote by $N(s)$ the set of nodes in $V \backslash V(s)$ connected to nodes in $V(s)$. Let $E^{(N)}(s)$ denote the set of edges between nodes in $N(s)$ and nodes in $V(s)$. For example, when s is the 3-node CIS consisting of nodes b, c, d shown in Fig. 1 (c), we have $d^{(3)}(s) = 3$, $V(s) = \{b, c, d\}$, $E(s) = \{(b, c), (b, d), (c, d)\}$, $N(s) = \{a\}$, $E^{(N)}(s) = \{(a, b), (a, c), (a, d)\}$, and $X(s)$ includes three CISes: the CIS consisting of nodes 1) a, b, and c; 2) a, b, and d; as well as 3) a, c, and d. Clearly a neighbor of s in $G^{(k)}$ corresponds to a subgraph that includes $k - 1$ nodes in s and one node in $N(s)$. For each $(k - 1)$-node set $\{v_1, \ldots, v_{k-1}\} \subset V(s)$ and each node $u \in N(s)$, the induced subgraph $s' = (V(s'), E(s'))$ of these k nodes, i.e., $V(s') = \{v_1, \ldots, v_{k-1}, u\}$ and $E(s') = \{(u, v) : u, v \in V(s')$ and $(u, v) \in E\}$, is a neighbor of s in $G^{(k)}$ when s' is a connected graph. Note that we can obtain $E(s')$ without querying any node in $N(s)$, since $E(s') = \{(u, v) : u, v \in V(s')$ and $(u, v) \in E\} = \{(u, v) : u, v \in V(s')$ and $(u, v) \in E(s) \cup E^{(N)}(s)\}$. Therefore $X(s)$ can be computed based on $V(s)$, $N(s)$, $E(s)$, $E^{(N)}(s)$, which are all obtained by querying nodes in s.
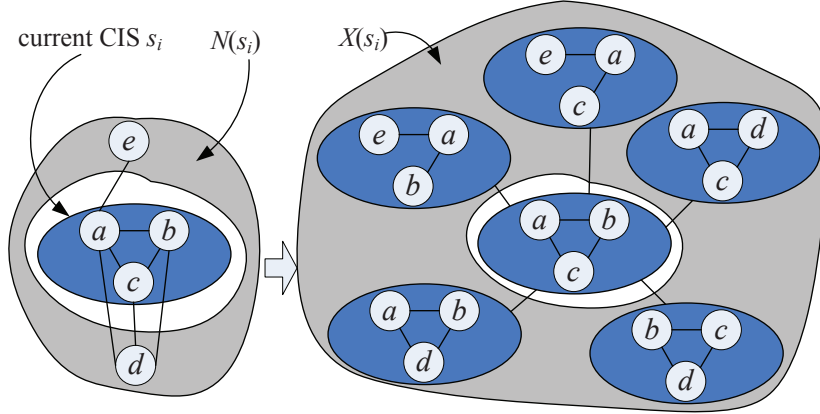
Fig. 5.   An example of applying a SRW to graph $G^{(3)}$.

The SRW algorithm proceeds in steps. Consider the i-th step, $i \ge 1$, and the current node is $s_i \in C^{(k)}$. SRW first computes $X(s_i)$, and then selects a CIS randomly from $X(s_i)$ as the next CIS to visit. For example, as shown in Fig. 5, suppose that the current sampled CIS s_i consists of nodes a, b, and c. After querying a, b, and c, we obtain $N(s_i) = \{e, d\}$ and the edges between nodes in $N(s_i)$ and nodes in s_i. Then our SRW algorithm computes $X(s_i)$ (i.e., the five CISes connecting

---

Algorithm 1: The pseudo-code of computing $X(s)$.

```
/*  s ∈ C^(k). N(s) is the set of nodes in V\V(s) connected to nodes in V(s). E^(N)(s)
    is the set of edges between nodes in N(s) and nodes in V(s).                    */
```
input : k-node CIS $s = (V(s), E(s))$, $N(s)$, $E^{(N)}(s)$
```
/*  X(s) ⊂ C^(k) is the set of neighbors of s in G^(k).                            */
```
output: $X(s)$

$X(s) = \{\}$;
foreach $\{v_1, \ldots, v_{k-1}\} \subset V(s)$ **do**
     foreach $u \in N(s)$ **do**
         ```
/*  generateGraph({v_1,...,v_{k-1}}, u, E(s), E^(N)(s)) returns a graph s' = (V(s'), E(s')),
    whose node set V(s') = {v_1,...,v_{k-1}, u}, and edge set
    E(s') = {(u,v) : u,v ∈ V(s') and (u,v) ∈ E(s) ∪ E^(N)(s)}                       */
```
         $s' = \text{generateGraph}(v_1, \ldots, v_{k-1}, u, E(s), E^{(N)}(s))$;
         ```
/*  connectedGraph(s') returns "True" when s' is a connected graph, and
    "False" otherwise.                                                             */
```
         if $\text{connectedGraph}(s')$ **then**
          | $X(s) = X(s) \cup \{s'\}$
         end
     end
end

---

to s_i shown in Fig. 5), and randomly select a new CIS from $X(s_i)$ as the next CIS s_{i+1} to sample. The pseudo-code of computing $X(s_i)$ is shown in Algorithm 1. As mentioned earlier, $X(s_i)$ is computed without querying any node in $N(s_i)$. Moreover, since $V(s_{i+1})$ differs from $V(s_i)$ in one and only one node, SRW only needs to **query one node** in the original graph G at each step. Let $d^{(k)}(s)$ be the degree of a k-node CIS s in graph $G^{(k)}$, that is the number of k-node CISes connected to s. Formally, SRW then can be modeled as a Markov chain with transition matrix $\boldsymbol{P}^{(k)} = [P_{x,y}^{(k)}]$, $x, y \in C^{(k)}$, where $P_{x,y}^{(k)}$ is defined as the probability that CIS y is selected as the next sampled k-node CIS given that the current k-node CIS is x. $P_{x,y}^{(k)}$ is computed as

$$P_{x,y}^{(k)} = \begin{cases} \frac{1}{d^{(k)}(x)}, & x \in C^{(k)}, y \in X(x), \\ 0, & \text{otherwise.} \end{cases}$$

The stationary distribution $\boldsymbol{\pi}^{(k)} = \left(\pi^{(k)}(s) : s \in C^{(k)}\right)$ of this Markov chain is

$$\pi^{(k)}(s) = \frac{d^{(k)}(s)}{\sum_{t \in C^{(k)}} d^{(k)}(t)}.$$

SRW is a regular RW over the undirected graph $G^{(k)}$, and we have the following theorem from [Lovász 1993; Ribeiro and Towsley 2010].

THEOREM 3.3. *If graph $G^{(k)}$ ($2 \leq k < |V|$) is non-bipartite and connected, the stationary distribution for the SRW to be at a k-node CIS $s \in C^{(k)}$ converges to $\boldsymbol{\pi}^{(k)} = \left(\pi^{(k)}(s) : s \in C^{(k)}\right)$. The probabilities of a SRW sampling edges in $E^{(k)}$ are equal when the SRW reaches the steady state.* □

Remark: As mentioned earlier, in most practical applications the connected non-bipartite assumption over $G^{(k)}$ only implies that the original graph G must have at least one node with degree three or larger and be connected.

Let $C(s)$ denote the subgraph class of a CIS s. $C(s)$ can be easily obtained by using the NAUTY algorithm [McKay 1981; 2009]. Define $\boldsymbol{1}(\mathcal{P})$ as the indicator function that equals one when the predicate $\mathcal{P}$ is true, and zero otherwise. Let s_j, $j > 0$, be the k-node CIS sampled by the SRW

at step j. Using the CISes visited by a SRW after $B > 1$ steps, we use the Horvitz-Thompson estimator [Ribeiro and Towsley 2010] to estimate the concentration of subgraph class $C_i^{(k)}$ as:

$$\hat{\omega}_i^{(k)} = \frac{1}{L} \sum_{j=1}^{B} \frac{\mathbf{1}(C(s_j) = C_i^{(k)})}{d^{(k)}(s_j)}, \qquad 1 \le i \le T_k, \tag{1}$$

where $L = \sum_{j=1}^{B} \frac{1}{d^{(k)}(s_j)}$.

THEOREM 3.4. *If $G^{(k)}$ ($2 \le k < |V|$) is non-bipartite and connected, then $\hat{\omega}_i^{(k)}$ ($1 \le i \le T_k$) in Eq. (1) is an asymptotically unbiased estimator of $\omega_i^{(k)}$.* $\qquad\qquad\square$

Remark: Theorem 3.4 provides the theoretical basis for producing unbiased estimates of the concentration of each CIS class in the graph under study. Proof is found in the appendix.

3.3. Pairwise subgraph random walk (PSRW)

In this subsection, we use SRW as a building block for our next subgraph statistics method. Instead of sampling over graph $G^{(k)}$, we apply SRW to $G^{(k-1)}$, and observe that two adjacent sampled $(k-1)$-node CISes (i.e., an edge in $G^{(k-1)}$) contains exactly k distinct nodes. Thus, we show how to obtain k-node CISes by sampling edges from $G^{(k-1)}$. Using this property we propose a better sampling method than SRW, which we denote *pairwise subgraph random walk* (PSRW). PSRW samples k-node CISes by applying SRW to $G^{(k-1)}$, $2 < k \le |V|$. In what follows we show that PSRW produces more accurate estimates than SRW as observed from experimental results in Section 4. It remains an open theoretical problem why PSRW significantly outperforms SRW. Our conjecture is that it is due to the fact that a RW on $G^{(k-1)}$ converges to its stationary behavior more quickly than $G^{(k)}$. Let s_j ($1 \le j \le B$) be the j-th $(k-1)$-node CIS sampled by applying a SRW to $G^{(k-1)}$. Consider the edge (s_j, s_{j+1}) in $G^{(k-1)}$. This edge is associated with a k-node CIS consisting all nodes contained in s_j and s_{j+1}. Therefore we obtain k-node CISes s_j^* ($1 \le j < B$), where s_j^* is the k-node CIS generated by (s_j, s_{j+1}). Fig. 6 shows an example of applying a PSRW to sample 3-node CISes $s_1^*, s_2^*, \ldots$ from G. We can see that $s_1^*, s_2^*, \ldots$ are generated based on 2-node CISes s_1, s_2 sampled by applying a SRW to $G^{(2)}$, where G and $G^{(2)}$ are shown in Fig. 1. For any k-node CIS $x \in C^{(k)}$, let $I^{(k-1)}(x)$ denote the number of $(k-1)$-node CISes contained by x. For example, 3-node CIS s_1^* in Fig. 1 contains two 2-node CISes: 1) the CIS consisting of nodes A and B, and 2) the CIS consisting of nodes A and E. Thus, $I^{(2)}(s_1^*) = 2$. Similarly we have $I^{(2)}(s_3^*) = 3$. It is easy to show that a k-node CIS x associates with $\frac{\left(I^{(k-1)}(x)\right)\left(I^{(k-1)}(x)-1\right)}{2}$ edges in graph $G^{(k-1)}$ and x is sampled by PSRW if and only if at least one of its associated edges in $G^{(k-1)}$ is sampled. For example, as shown in Fig. 7 where $k = 3$, s_3^* is sampled when PSRW samples at least one of its associated edges in $G^{(2)}$, i.e., the $\frac{\left(I^{(2)}(s_3^*)\right)\left(I^{(2)}(s_3^*)-1\right)}{2} = 3$ red edges in Fig. 7. From Theorem 3.3, we know that SRW samples each edge in $G^{(k-1)}$ with equal probability at steady state, therefore k-node CIS x is sampled with the following probability

$$\pi_E^{(k)}(x) = \frac{I^{(k-1)}(x)\left(I^{(k-1)}(x) - 1\right)}{\sum_{y \in C^{(k)}} I^{(k-1)}(y)\left(I^{(k-1)}(y) - 1\right)}.$$

Thus, using the Horvitz-Thompson estimator, we estimate the concentration of subgraph class $C_i^{(k)}$ as follows,

$$\tilde{\omega}_i^{(k)} = \frac{1}{H} \sum_{j=1}^{B-1} \frac{\mathbf{1}(C(s_j^*) = C_i^{(k)})}{I^{(k-1)}(s_j^*)\left(I^{(k-1)}(s_j^*) - 1\right)}, \quad 1 \le i \le T_k, \tag{2}$$

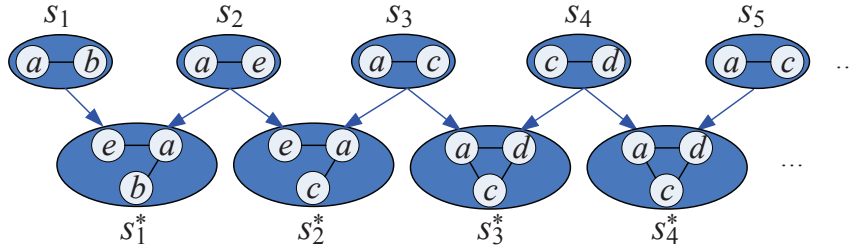where $H = \sum_{j=1}^{B-1} \left[I^{(k-1)}(s_j^*) \left(I^{(k-1)}(s_j^*) - 1 \right) \right]^{-1}$.

Fig. 6. An example of applying a PSRW to sample 3-node CISes from G. s_1, s_2, ..., are 2-node CISes sampled by applying a SRW to $G^{(2)}$. s_1^*, s_2^*, ..., are 3-node CISes generated based on s_1, s_2, ... G and $G^{(2)}$ are graphs shown in Fig. 1.
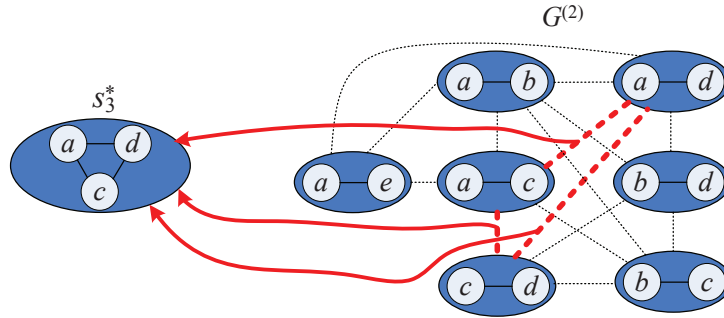

Fig. 7. 3-node CIS s_3^* can be generated by each of the three red edges in $G^{(2)}$.

THEOREM 3.5. *If $G^{(k)}$ ($2 \leq k \leq |V|$) is non-bipartite and connected, then $\tilde{\omega}_i^{(k)}$ ($1 \leq i \leq T_k$) in Eq. (2) is an asymptotically unbiased estimator of $\omega_i^{(k)}$.* □

Remark: We find that PSRW cannot be easily further extended, that is, we cannot sample k-node CISes based on applying a SRW to graph $G^{(k')}$, where $k' < k - 1$. This is because it is difficult to analyze and remove sampling errors when we generate a k-node CIS using $k - k' + 1$ CISes consequently sampled by a SRW over $G^{(k')}$. Note that $k - k' + 1$ consequently sampled k'-node CISes might contain less than k different nodes.

3.4. Mix Subgraph Sampling (MSS)

The previous two subsections focus on subgraph classes of a specific size k. Motivated by [Bhuiyan et al. 2012], we study the problem of estimating the concentrations of subgraph classes of sizes $k - 1$, k, and $k + 1$ simultaneously. Clearly we can naively solve this problem by applying three independent PRSWs to calculate the concentrations of subgraph classes of sizes $k - 1$, k, and $k + 1$. However, this naive approach is inefficient. Next, we propose a more efficient sampling method, mix subgraph sampling (MSS), which requires fewer queries to achieve the same estimation accuracy. MSS samples k-node CISes by applying a SRW to $G^{(k)}$.

Let s_j, $j > 0$, be the k-node CIS sampled by the SRW at step j. Using the CISes visited by a SRW after $B > 1$ steps, MSS estimates the concentrations of k-node subgraph classes using Eq. (1). Similar to PSRW, MSS uses s_j ($1 \leq j \leq B$) to generate (k+1)-node CISes, and then estimates the concentrations of $(k+1)$-node subgraph classes using Eq. (2). Last, we show how MSS estimates the concentrations of $(k-1)$-node subgraph classes based on s_j, $1 \leq j \leq B$. Let $C^{(k-1)}(s)$ denote the set of $(k-1)$-node CISes contained in a CIS s. For example, if s is the 4-node CIS consisting of nodes a, b, c, and e shown in Fig. 1, then $S^{(3)}(s)$ consists of three 3-node CISes: 1) the CIS consisting of nodes a, b, and c; 2) the CIS consisting of nodes a, b, and e; 3) the CIS consisting of nodes b, c, and e. Define $O^{(k)}(s')$ as the set of k-node CISes that contain a CIS s'. For example, if s' is the 3-node CIS consisting of nodes a, c, and e shown in Fig. 1, then $O^{(4)}(s')$ consists of two 4-node CISes: 1) the CIS consisting of nodes a, b, c, and e; 2) the CIS consisting of nodes a, c, d, and e. Finally MSS estimates the concentration of subgraph class $C_i^{(k-1)}$ as follows,

$$\breve{\omega}_i^{(k-1)} = \frac{1}{Q} \sum_{j=1}^{B} \frac{1}{d^{(k)}(s_j)} \sum_{s' \in C^{(k-1)}(s_j)} \frac{\mathbf{1}(C^{(k-1)}(s') = C_i^{(k-1)})}{|O^{(k)}(s')|}, \quad 1 \leq i \leq T_{k-1}, \quad (3)$$

where $Q = \sum_{j=1}^{B} \frac{1}{d^{(k)}(s_j)} \sum_{s' \in C^{(k-1)}(s_j)} \frac{1}{|O^{(k)}(s')|}$.

THEOREM 3.6. *If $G^{(k)}$ ($3 \leq k \leq |V|$) is non-bipartite and connected, then $\breve{\omega}_i^{(k-1)}$ ($1 \leq i \leq T_{k-1}$) in Eq. (3) is an asymptotically unbiased estimator of $\omega_i^{(k-1)}$.*   □

4. DATA EVALUATION

In this section, we first introduce our experimental datasets and a comparison model, which is used to evaluate the performance of our methods for characterizing CIS classes of a specific size k in comparison with state-of-the-art methods. Then we present the experimental results of PSRW for $k \in \{3, \ldots, 6\}$. Last, we compare the special case MSS $k = 3, 4, 5$ against GUISE [Bhuiyan et al. 2012]. Our experiments are conducted on a Dell Precision T1650 workstation with an Intel Core i7-3770 CPU 3.40 GHz processor and 8 GB DRAM memory.

4.1. Datasets

Our experiments are performed on a variety of publicly available datasets taken from the Stanford Network Analysis Platform (SNAP)[2]. We start by evaluating the performance of our methods in characterizing 3-node CISes over four million-node graphs: Flickr, Pokec, LiveJournal, and YouTube, contrasting our results with ground truth computed through brute force. Since it is computationally intensive to calculate the ground-truth of k-node CIS classes for $k \geq 4$ in large graphs, the experiments for k-CISes, $k \geq 4$, are performed on three relatively small graphs Epinions, Slashdot, and Gnutella, where computing the ground-truth by brute force is feasible. We also specifically evaluate the performance of our methods for characterizing signed CIS classes in the Epinions and Slashdot signed graphs. Flickr, LiverJournal, and YouTube are popular photo, blog, and video sharing websites respectively, where a user can subscribe to other user updates such as photos, blogs, and videos. Pokec is the most popular on-line social network in Slovakia, and has been in existence for more than ten years. These four networks can be represented by directed graphs, where nodes representing users and a directed edge from u to v represents that user u subscribes to user v or u tags user v as a friend. Epinions is a who-trust-whom OSN providing consumer reviews, where a directed edge from u to v represents that user u trusts user v. Slashdot is a technology-related news website for its specific user community, where a directed edge from u to v represents that user u tags user v as a friend or foe. Epinions and Slashdot networks can be represented by signed graphs, where a positive edge from u to v indicates that u trusts v or u tags user v a friend, and a negative

---

edge from u to v indicates that u distrusts v or u tags user v a foe. Gnutella is a peer-to-peer file sharing network. Nodes represent users in the Gnutella network and edges represent connections between the Gnutella users. In the following experiments, we evaluate our proposed methods on the largest connected component (LCC) of these graphs (summarized in Table II).

Table II. Overview of graph datasets used in our simulations.

Graph	LCC		
	nodes	edges	directed-edges
Flickr [Mislove et al. 2007]	1,624,992	15,476,835	22,477,014
Pokec [Takac and Zabovsky 2012]	1,632,805	22,301,964	30,622,564
LiveJournal [Mislove et al. 2007]	5,284,459	48,688,097	76,901,758
YouTube [Mislove et al. 2007]	1,134,890	2,987,624	4,942,035
Epinions [Richardson et al. 2003]	119,130	704,267	833,390
Slashdot [Leskovec et al. 2009]	77,350	416,695	516,575
Gnutella [Leskovec et al. 2009]	6,299	20,776	20,776

Note: "directed-edges" refers to the number of directed edges in a directed graph, "edges" refers to the number of edges in an undirected graph, and "LCC" refers to the largest connected component of a given graph.

4.2. Comparison model

First we introduce the error metric used to compare the different sampling methods. Mean square error (MSE) is a common measure to quantify the error of an estimate $\hat{\omega}$ with respect to its true value $\omega > 0$. It is defined as $\text{MSE}(\hat{\omega}) = \text{E}[(\hat{\omega} - \omega)^2] = \text{var}(\hat{\omega}) + (\text{E}[\hat{\omega}] - \omega)^2$. We can see that $\text{MSE}(\hat{\omega})$ decomposes into a sum of the variance and bias of the estimator $\hat{\omega}$, both quantities are important and need to be as small as possible to achieve good estimation performance. When $\hat{\omega}$ is an unbiased estimator of ω, then $\text{MSE}(\hat{\omega}) = \text{var}(\hat{\omega})$. In our experiments, we study the normalized root mean square error (NRMSE) to measure the relative error of the estimator $\hat{\omega}_i$ of the subgraph class concentration ω_i, $i = 1, 2, \ldots$. $\text{NRMSE}(\hat{\omega}_i)$ is defined as:

$$\text{NRMSE}(\hat{\omega}_i) = \frac{\sqrt{\text{MSE}(\hat{\omega}_i)}}{\omega_i}, \qquad i = 1, 2, \ldots.$$

When $\hat{\omega}_i$ is an unbiased estimator of ω_i, then $\text{NRMSE}(\hat{\omega}_i)$ is equivalent to the normalized standard error of $\hat{\omega}_i$, i.e., $\text{NRMSE}(\hat{\omega}_i) = \sqrt{\text{var}(\hat{\omega}_i)}/\omega_i$. Note that our metric uses the relative error. Thus, when ω_i is small, we consider values as large as $\text{NRMSE}(\hat{\omega}_i)$ to be acceptable. In all our experiments, we average the estimates and calculate their NRMSEs over 1,000 runs.

We compute the NRMSE of our methods for estimating concentrations of CIS classes of specific size k, in comparison with that of two state-of-the-art algorithms FANMOD [Wernicke 2006] and GUISE in [Bhuiyan et al. 2012] under the constraint that the number of queries cannot exceed B^*. As mentioned earlier, issues arise when comparing our methods of estimating concentrations of CIS classes of a specific size k to that of the method in [Bhuiyan et al. 2012], as the latter wastes most queries to sample CISes of size not equal to k. To address this problem, we use adapt the MHRW of GUISE [Bhuiyan et al. 2012] to focus on subgraphs of size k, which we name metropolis-Hastings subgraph random walk (MHSRW). Later we compare MSS $k = 3, 4, 5$ directly to GUISE [Bhuiyan et al. 2012] showing that MSS is significantly more accurate than GUISE.

To sample k-node CISes, MHSRW works as follows: At each step, MHSRW randomly selects a k-node CIS y from $X(x)$, the set of neighbors of the current k-node CIS x on CIS relationship graph $G^{(k)}$, and accepts the move with probability $\min\left\{1, \frac{d^{(k)}(y)}{d^{(k)}(x)}\right\}$. Otherwise, it remains at x. MHSRW samples k-node CISes uniformly when it reaches the steady state. Based on CIS samples s_j $(1 \leq j \leq B)$, MHSRW estimates the concentration of subgraph class $C_i^{(k)}$ for graph G_d as

follows,

$$\breve{\omega}_i^{(k)} = \frac{1}{B} \sum_{j=1}^{B} \mathbf{1}(C(s_j) = C_i^{(k)}), \qquad 1 \le i \le T_k.$$

4.3. Results of estimating 3-node CIS class concentrations

In this subsection we show results of estimating 3-node CIS class concentrations for undirected, directed, and signed graphs respectively.

1) **3-node undirected CIS classes**: Fig. 8 shows the results of estimating $\omega_2^{(3)}$, the concentration of the 3-node undirected CIS class 2 (or the triangle as shown in Fig. 2 (a)) for Flickr, Pokec, LiveJournal, and YouTube graphs, where B^* is the number of queries, i.e., the number of distinct nodes required to query in the original graph G. The true value of $\omega_2^{(3)}$ for Flickr, Pokec, LiveJournal, and YouTube are 0.0404, 0.0161, 0.0451, and 0.0021 respectively. The results show that PSRW exhibits the smallest errors, which are almost an order of magnitude less than errors of MHSRW and FAN-MOD for Flickr and Pokec graphs. SRW is more accurate than MHSRW and FANMOD but less accurate than PSRW. Note that PSRW uses only $B^* = 3 \times 10^3$ queries and still exhibits smaller errors than the other methods that use one order of magnitude more queries $B^* > 3 \times 10^4$. Hence, PSRW reduces more than 10-fold the number of queries required to achieve the same estimation accuracy. Meanwhile we observe that an order of magnitude increase in B^* roughly decreases the error by $1/\sqrt{10}$ for all methods studied. Fig. 9 plots the evolution of $\omega_2^{(3)}$ estimates as a function of B (the number of sampling steps) for one run. We observe that PSRW *converges* to the value of $\omega_2^{(3)}$ when 10^3, 10^3, 2×10^5, and 10^4 CISes are sampled for Flickr, Pokec, LiveJournal, and YouTube respectively, and is much more quickly than the other methods. To compare the performances of these methods after the random walks have entered the stationary regime, Fig. 10 shows the results of estimating $\omega_2^{(3)}$ based on 10^4 CISes sampled after 10^6 steps. We can see that PSRW outperforms the other methods at steady state.

2) **3-node directed CIS classes**: Fig. 11 shows the concentrations of 3-node directed CIS classes for Flickr, Pokec, LiveJournal, and YouTube graphs, and the subgraph classes and their associated IDs are listed in Fig. 3. The total numbers of 3-node CISes are 1.4×10^{10}, 2.0×10^9, 6.9×10^9, and 1.5×10^9 for Flickr, Pokec, LiveJournal, and YouTube respectively. Fig. 12 compares concentrations estimates of 3-node directed CIS classes for different methods under the same number of queries $B^* = 10,000$. The results show that subgraph classes with smaller concentrations have larger NRMSEs. PSRW is significantly more accurate than the other methods for most subgraph classes. SRW is not shown in the plots but its performance lies again somewhere between MHSRW and FANMOD.

3) **3-node signed and undirected CIS classes**: Figure 13 shows the concentrations of 3-node signed and undirected CIS classes (as listed in Fig. 4) for Epinions and Slashdot graphs. Epinions and Slashdot graphs have 1.7×10^8 and 6.7×10^7 signed and undirected 3-node CISes respectively. Fig. 14 shows the estimated concentrations of signed and undirected 3-node CIS classes for different methods under $B^* = 2,000$ queries. The results show that subgraph classes with smaller concentrations have larger NRMSEs. All NRMSEs given by PSRW are much smaller than one for all subgraph classes. PSRW is almost four times more accurate than MHSRW and FANMOD. MHSRW exhibits slightly smaller errors than FANMOD for most subgraph classes.

4.4. Results of estimating 4-node CIS class concentrations

Figure 15 shows the concentrations of 4-node undirected CIS classes (as listed in Fig. 2 (b)) for Epinions, Slashdot, and Gnutella graphs. Epinions, Slashdot, and Gnutella graphs have 2.5×10^{10}, 2.1×10^{10}, and 1.0×10^7 undirected four-node CISes respectively. Fig. 16 shows the estimated concentrations of undirected four-node CIS classes for different methods under $B^* = 2,000$ queries.
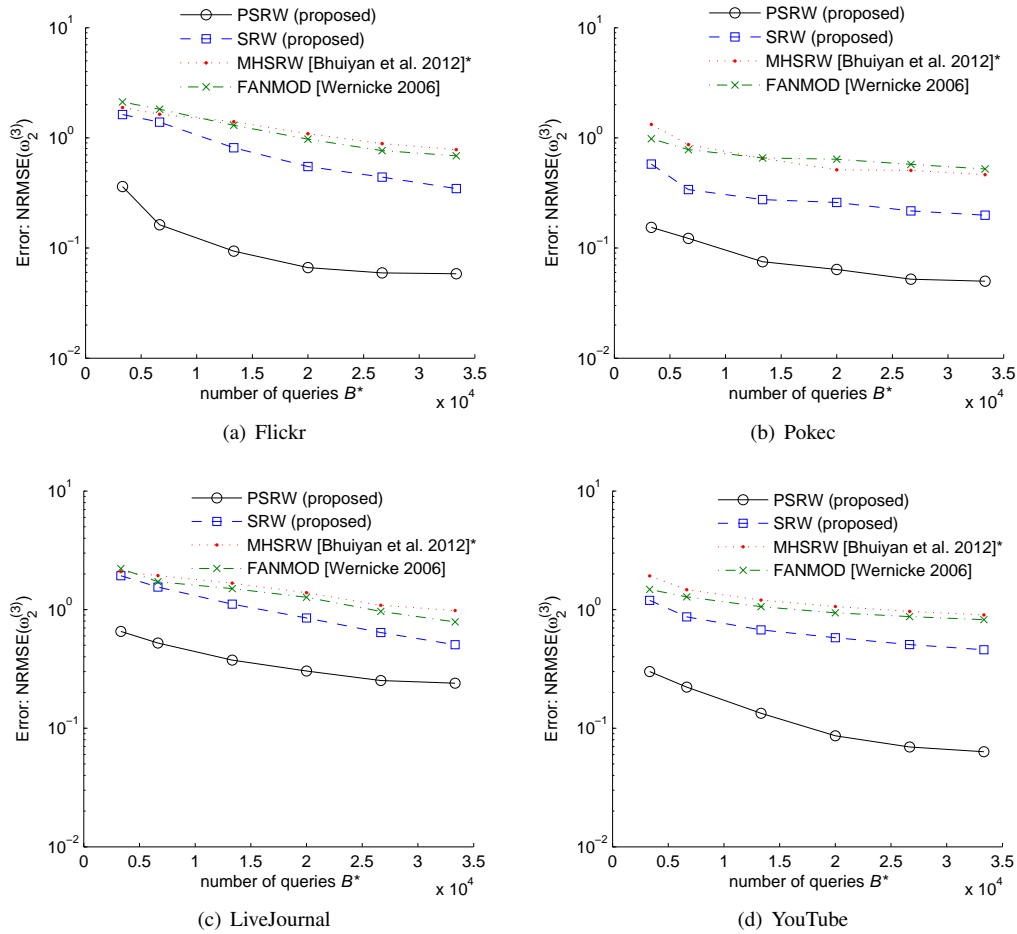
Fig. 8. (Flickr, Pokec, LiveJournal, and YouTube) Compared NRMSEs of concentration estimates of 3-node undirected CIS classes for different methods.

The results show that all NRMSEs given by PSRW are smaller than 0.4 for subgraph classes 1 to 5. PSRW is significantly more accurate than the other methods.

4.5. Results of estimating 5-node and 6-node CIS class concentrations

Because the number of k-node CISes exponentially increases with k, it is computationally intensive to calculate the ground-truth of k-node CIS classes' when $k \geq 5$. Nevertheless, we proceed to evaluate our methods based on a relatively small graph Gnutella which has 6,299 nodes and 20,776 edges for $k = 5$ and $k = 6$. Gnutella has 3.9×10^8 five-node CISes and 1.7×10^{10} six-node CISes. It takes almost one day to obtain all these subgraphs using the software provided in Kashtan et al. [Kashtan et al. 2004]. Fig. 17 shows NRMSEs of concentration estimates of one five-node undirected CIS class and one five-node undirected CIS class for Gnutella graph. The five-node undirected CIS class we studied is topologically equivalent to a five-node tree with depth one. The true value of its concentration is 0.183 for Gnutella graph. The results show that PSRW is nearly four times more accurate than MHSRW and FANMOD. The six-node undirected CIS class we studied is topologically equivalent to a six-node tree with depth one. The true value of its concentration is

(a) Flickr, $\omega_2^{(3)} = 0.0404$.

(b) Pokec, $\omega_2^{(3)} = 0.0161$.

(c) LiveJournal, $\omega_2^{(3)} = 0.0451$.
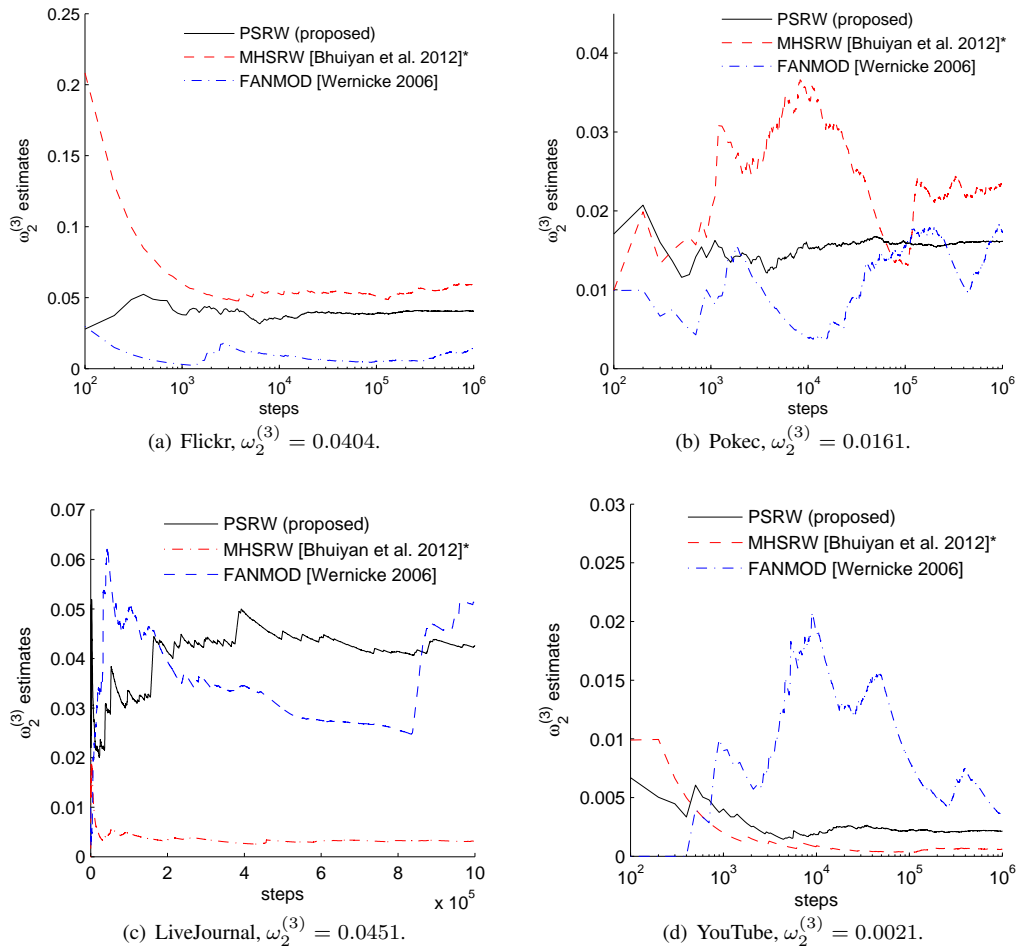
(d) YouTube, $\omega_2^{(3)} = 0.0021$.

Fig. 9.   (Flickr, Pokec, LiveJournal, and YouTube) Compared $\omega_2^{(3)}$ estimates of 3-node undirected CIS classes for different methods.

0.0589 for the Gnutella graph. The results show that PSRW is nearly twice as accurate as MHSRW and FANMOD.

4.6. Time cost of sampling CISes

The time cost of sampling a k-node CIS consists of two parts: 1) computational time, and 2) the query response time. We observe that the computation times increase with k for PSRW, MHSRW, and FANMOD, and the computation times are smaller than 0.1 second for $k \leq 5$, which is usually smaller than the query rate limits for querying a node imposed by OSNs. Thus, we can easily find that PSRW is computationally more efficient than MHSRW, since PSRW samples k-node CISes from graph $G^{(k-1)}$, while MHSRW samples k-node CISes from graph $G^{(k)}$. We compare the performances of different methods under the same time budget T. We do simulations to evaluate the performance of different methods for two cases: 1) the graph is stored in a local database with near zero query delay, and 2) the graph is stored in a remote database with 100 milliseconds query delay. Fig. 18 shows the NRMSEs of estimates of $\omega_2^{(3)}$ under T=200, 400, 600, 800, and 1,000 seconds for the Flickr graph. The results show that PSRW is four and five times as accurate as the other methods
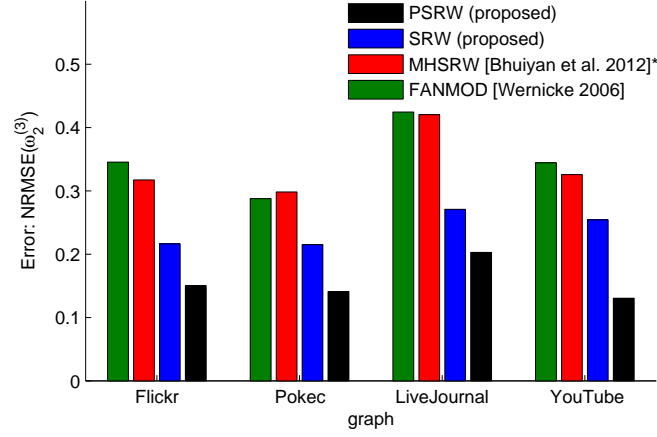
Fig. 10. (Flickr, Pokec, LiveJournal, and YouTube) Compared NRMSEs of concentration estimates of 3-node undirected CIS classes for different methods at the stationary state.
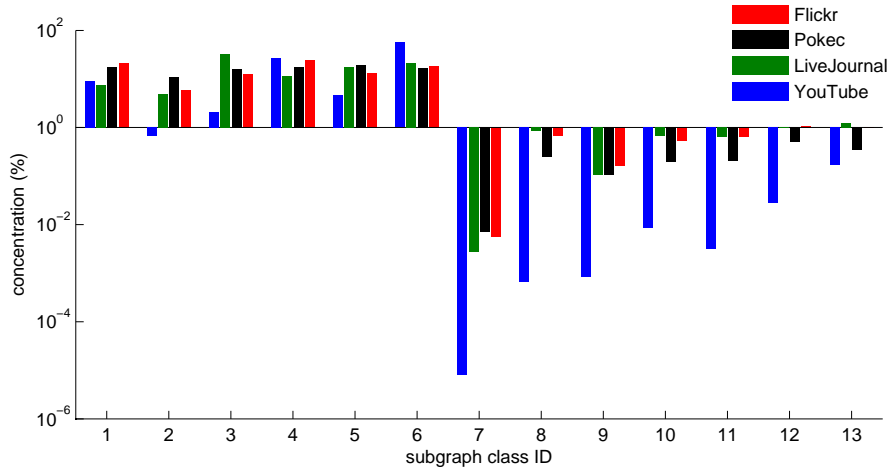

Fig. 11. (Flickr, Pokec, LiveJournal, and YouTube) Concentrations of 3-node directed CIS classes.

for the same time budget T for the local and remote databases respectively. The results for other graphs are similar, so we omit them here.

4.7. Comparison with GUISE

Next, we evaluate the performance of our method MSS for the special case of simultaneous estimation of CISes $k = 3, 4, 5$ concentrations as in GUISE [Bhuiyan et al. 2012]. Let $\omega^{(k)} = (\omega_1^{(k)}, \ldots, \omega_{T_k}^{(k)})$ and $\hat{\omega}^{(k)} = (\hat{\omega}_1^{(k)}, \ldots, \hat{\omega}_{T_k}^{(k)})$, where $\omega_i^{(k)}$ is the concentration of subgraph class $C_i^{(k)}$ and $\hat{\omega}_i^{(k)}$ is an estimate of $\omega_i^{(k)}$. We define the root mean square error (RMSE) as:

$$\mathrm{RMSE}(\hat{\omega}^{(k)}) = \sqrt{\mathrm{E}[\sum_{i=1}^{T_k}(\hat{\omega}_i^{(k)} - \omega_i^{(k)})^2]}, \qquad k = 3, 4, \text{ and } 5,$$
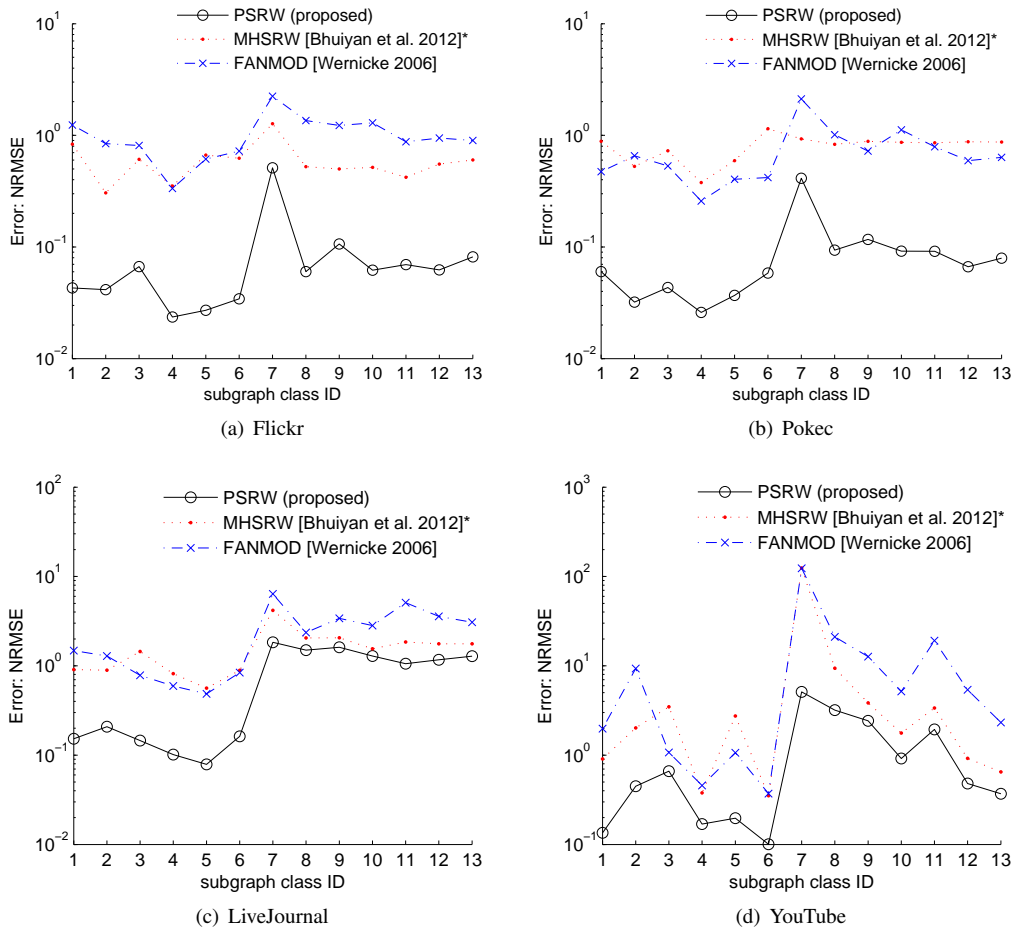
Fig. 12.   (Flickr, Pokec, LiveJournal, and YouTube) Compared NRMSEs of concentration estimates of 3-node directed CIS classes for different methods under the same number of queries $B^* = 10,000$.
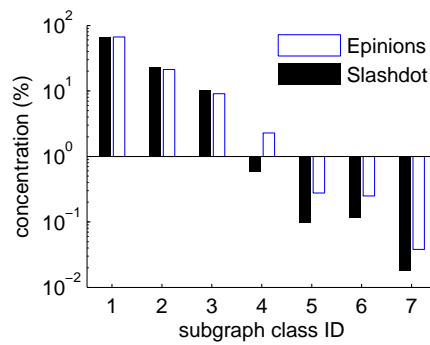
Fig. 13.   (Epinons and Slashdot) Concentrations of 3-node signed and undirected CIS classes.

(a) Epinons
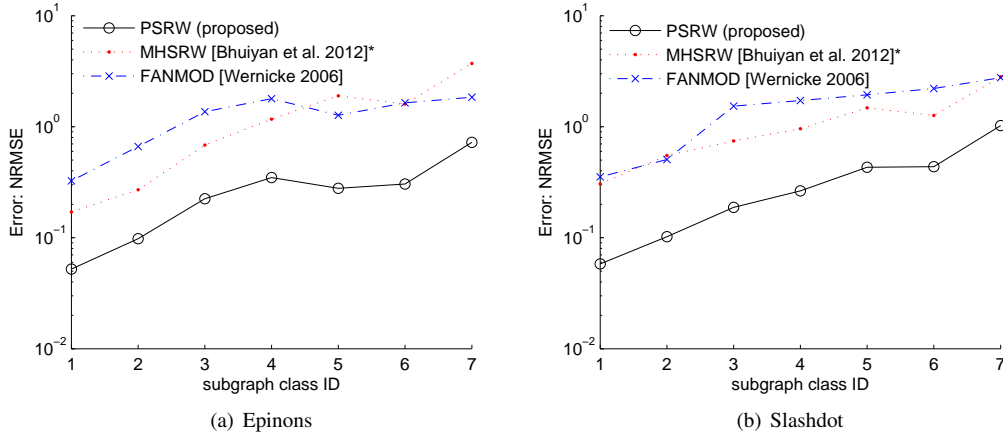
(b) Slashdot

Fig. 14. (Epinons and Slashdot) Compared NRMSEs of concentration estimates of 3-node signed and undirected CIS classes for different methods under the same number of queries $B^* = 2,000$.
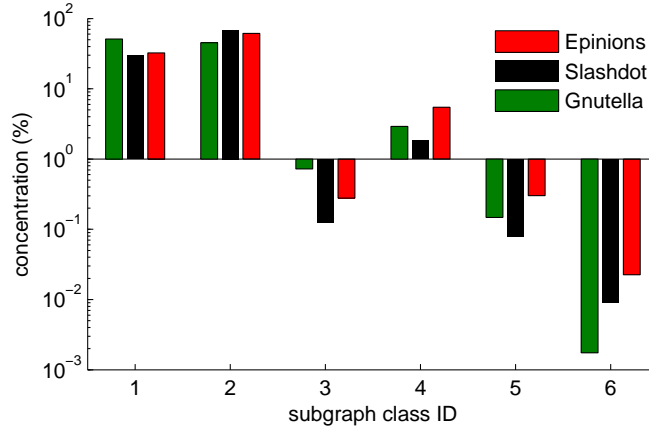
Fig. 15. (Epinons, Slashdot, and Gnutella) Concentrations of the 4-node undirected CIS classes.

which measures the error of the estimate $\hat{\omega}^{(k)}$ with respect to its true value $\omega^{(k)}$. Note here the variable $\hat{\omega}^{(k)}$ of function RMSE($\hat{\omega}^{(k)}$) is a *vector*. In our experiments, we average the estimates and calculate their RMSEs over 1,000 runs. Fig. 19 shows RMSEs of estimates of $\omega^{(3)}$, $\omega^{(4)}$, and $\omega^{(5)}$ for different methods under $B^* = 3,000$ queries. Besides MSS and GUISE, we also use PSRW to estimate $\omega^{(3)}$, $\omega^{(4)}$, and $\omega^{(5)}$ respectively. For simplicity, we provide PSRW a budget of 1,000 queries for each value of k, since it is hard to determine the optimal budget allocation for PSRW to jointly estimate $\omega^{(3)}$, $\omega^{(4)}$, and $\omega^{(5)}$, The results show that MSS is more accurate than PSRW, and is nearly three times more accurate than GUISE.

(a) Epinons

(b) Slashdot

(c) Gnutella

Fig. 16. (Epinons and Slashdot) Compared NRMSEs of concentration estimates of 4-node undirected CIS classes for different methods under the same number of queries $B^* = 2,000$.
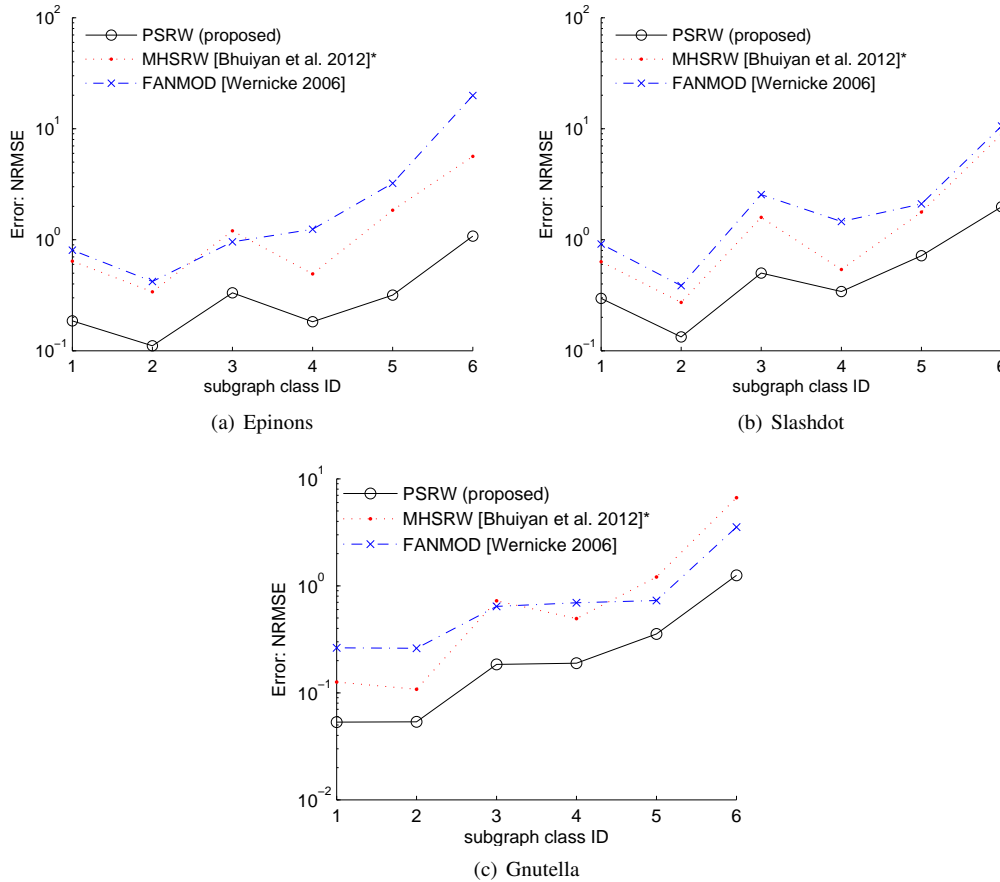
5. APPLICATIONS

In this section, we apply our methods to understand intrinsic properties of some large OSNs. We conduct experiments on Chinese OSNs Sina microblog[3] and Douban[4]. Sina microblog is the most popular Chinese microblog service, and has many features similar to Twitter. It has more than 300 million registered users as of February 2012. Douban provides an exchange platform for reviews and recommendations on movies, books, and music albums. It has approximately 6 million registered users as of 2009 [Zhao et al. 2011]. Douban and Sina microblog can be both modeled as directed graphs, where edges are formed by users' following and follower relationships. We conducted experiments in September 2012 on Sina microblog and Douban. By using PSRW, we sampled approximately 500,000 3-node CISes from Sina microblog and Douban respectively. Fig. 20 (a) shows the estimated concentrations of 3-node directed CIS classes. It shows that closed subgraph classes (classes 8–13) have much lower concentrations than unclosed subgraph classes (classes 1–6), which indicates that Douban and Sina microblog have a small fraction of closed triangles, and thus they have small clustering coefficients, unlike the friendship relationship based OSNs such as Facebook.

---

[3]www.weibo.com

[4]www.douban.com

(a) 5-node tree with depth one
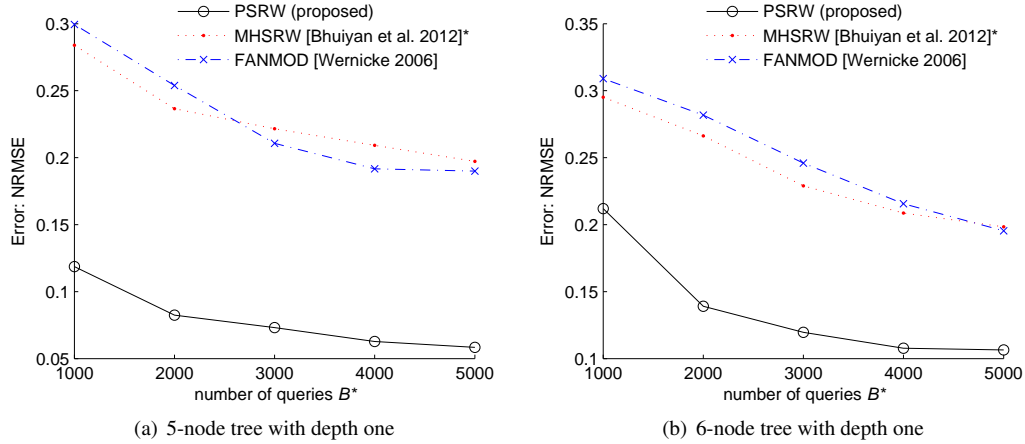
(b) 6-node tree with depth one

Fig. 17.   (Gnutella) Compared NRMSEs of concentration estimates of one 5-node undirected CIS class and one 6-node undirected CIS class for different methods.

(a) Local database (near zero query delay)

(b) Remote database (100 milliseconds of query delay)
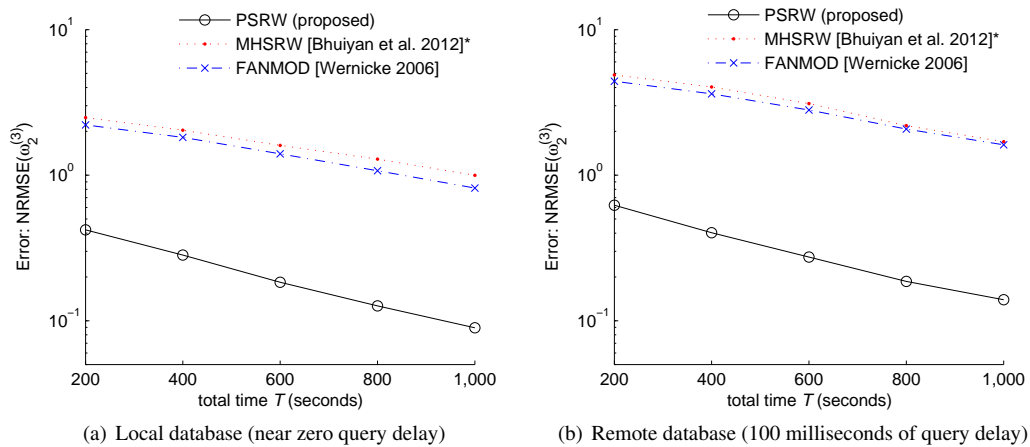
Fig. 18.   (Flickr) NRMSEs of error estimates of $\omega_2^{(3)}$ under for computer time T.

We observe that the concentration of subgraph class 7 is almost zero, and is omitted from the concentration results shown in Fig. 20. From Fig. 3, we observe that subgraph class 7 is a directed circle of three nodes, which corresponds to three persons A, B, C, with A following B, B following C, and C following A. A concentration of zero might be explained by the asymmetry of following relationships, where a following edge usually indicates statuses of the two end users, e.g., an edge from a low-status user to high-status user such as a celebrity. Therefore, three users with different statuses are unlikely to form a closed circle.

Next, we study the Z-scores of these subgraph classes. The Z-score of each subgraph class $C_i^{(k)}$, $k > 1$, is defined as

$$Z_i^{(k)} = \frac{\omega_i^{(k)} - \mu_i^{(k)}}{\sigma_i^{(k)}}, \qquad 1 \leq i \leq T_k, \qquad (4)$$

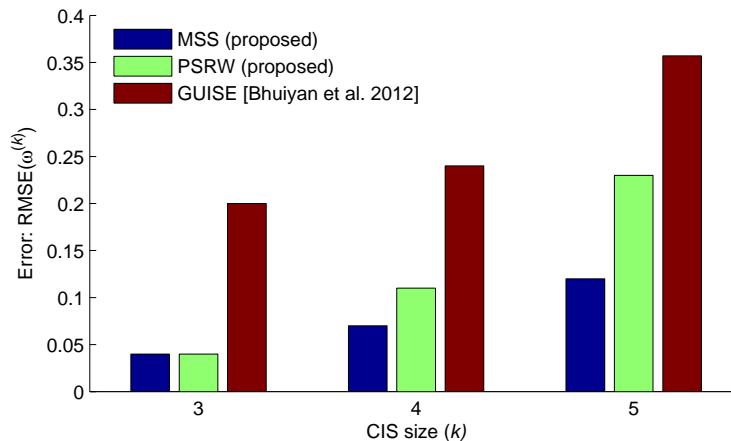Fig. 19. (Gnutella) Compared errors of characterizing 3-node, 4-node, and 5-node undirected CIS classes simultaneously for different methods under the same number of queries $B^* = 3,000$.

where $\mu_i^{(k)}$ and $\sigma_i^{(k)}$ are the mean and the standard deviation of the concentration of $C_i^{(k)}$ for random graphs with the same in-degree and out-degree sequence as G_d. Clearly the Z-score of $C_i^{(k)}$ is a qualitative measure on the significance of $C_i^{(k)}$ [Milo et al. 2002].

We propose a method to estimate $\mu_i^{(k)}$ and $\sigma_i^{(k)}$ as follows: First, we use graph sample methods such as RW to estimate the joint in-degree and out-degree distribution $\phi = (\phi(i,j) : i,j \geq 0)$, where $\phi(i,j)$ is the fraction of nodes in G_d with in degree i and out degree j. In essence, this is similar to the problem of estimating node label densities as studied in our previous work [Ribeiro and Towsley 2010]. We use the configuration model [Molloy and Reed 1995] to generate random networks according to the estimated joint in-degree and out-degree distribution $\hat{\phi}(i,j)$. To generate a random graph, we first generate $|V|$ nodes, and the in-degree and out-degree of each node are randomly selected according to $\hat{\phi}(i,j)$, where the graph size $|V|$ can be estimated by sampling methods proposed in [Katzir et al. 2011]. We then use the configuration model [Molloy and Reed 1995] to generate a group of random graphs. Algorithm 2 describes the pseudo-code of our method for generating a random graph. Finally, we compute the mean and standard deviation of the subgraph class concentration based on randomly generated graphs.

Algorithm 2: Pseudo-code of random graph generation algorithm.

1: **Step 1**: Assign each node v with $d_I(v)$ incoming edge stubs (in-stubs) and $d_O(v)$ outgoing edge stubs (out-stubs).
2: **Step 2**: Pick an unconnected in-stub randomly from all nodes' in-stubs. Denote the associated node of selected in-stub as v_i.
3: **Step 3**: Pick an unconnected out-stub randomly from all nodes' out-stubs. Denote the associated node of selected out-stub as v_o. Repeat this step when $v_o = v_i$ or there already exists an edge from v_i to v_j.
4: **Step 4**: Connect the selected in-stub and out-stub.
5: Repeat Step 2 to Step 4 until no unconnected in-stub or out-stub remains.

Using the above method, we estimate the joint degree distribution based on nearly one million unique nodes sampled by RW for Sina microblog and Douban respectively, and then generate 1,000 random graphs to compute the mean and the standard deviation of 3-node subgraph classes' concentrations, which are used for estimating Z-scores shown in Eq. (4). Fig. 20 (b) shows estimated

Z-scores of 3-node directed CISes. We find that subgraph classes 1 and 3 have higher Z-scores in Sina microblog than Douban, where subgraph class 1 can be viewed as a *listening type*, i.e., users follow many celebrities, and subgraph class 3 can be viewed as a broadcast type, i.e., celebrities have many fans. This indicates that Sina microblog acts more like a news media than an OSN, which is similar to Twitter as observed in [Kwak et al. 2010]. Subgraph class 6 has a higher Z-score in Douban than Sina microblog. It may be because Douban is an interest-based network, where an edge between two users with many common interests is more likely to be symmetric than asymmetric.

(a) concentrations

(b) Z-scores
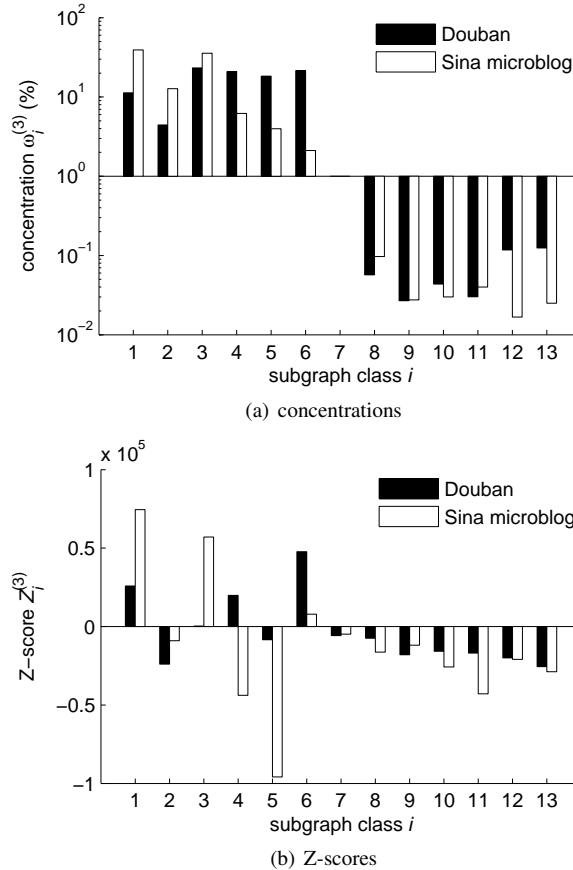
Fig. 20.   Results of real applications for all 3-node directed CISes.

6. RELATED WORK

In this paper we aim to characterize small subgraphs in *a single large graph*, which is a very different problem than that of estimating the number of subgraph patterns appearing in *a large set of graphs* studied in [Hasan and Zaki 2009]. Our problem can be directly solved by methods of enumerating all subgraphs of a specific size and type, such as triangle listing [Chu and Cheng 2012] and maximal clique enumeration [Cheng et al. 2011]. There are several subgraph concentration computation methods for motif discovery using different subgraph enumeration and counting methods [Chen et al. 2006; Kashani et al. 2009]. However these methods need to process the whole graph and are

computationally hard for large graphs. Meanwhile most of these methods are difficult to combine with sampling techniques. OmidiGenes et al. [Omidi et al. 2009] proposed a subgraph enumeration and counting method using sampling. However this method suffers from unknown sampling bias. To estimate subgraph class concentrations, Kashtan et al. [Kashtan et al. 2004] proposed a connected subgraph sampling method using random edge sampling. However their method is computationally expensive when calculating the weight of each sampled subgraph, which is used for correcting bias introduced by edge sampling. To address this drawback, Wernicke [Wernicke 2006] proposed a new method named FANMOD based on enumerating subgraph trees to detect network motifs. To sample a k-node CIS, their method needs to explore more than k nodes, which is expensive when exploring graph topology via crawling. Neither the method proposed in [Kashtan et al. 2004] nor [Wernicke 2006] can be applied to detect motifs in OSNs without the complete knowledge of the graph topology, since they rely on uniform edge sampling and uniform node sampling techniques respectively, which may not be feasible because these sampling functions are not supported by most OSNs.

Similar to estimate subgraph class concentrations, Bhuiyan et al. [Bhuiyan et al. 2012] propose a method GUISE for estimating 3-node, 4-node, and 5-node subgraph frequency distribution, that is, ($\frac{n_x}{N}$: x is a 3-node, 4-node, or 5-node undirected and connected subgraph class), where n_x be the number of undirected CISes in subgraph class x, and N is the total number of 3-node, 4-node, and 5-node undirected CISes. GUISE builds a new graph G_{mix}, whose node set consists of all 3-node, 4-node, and 5-node CISes. For a 3-node CIS, all 3-node and 4-node CISes having 2 and 3 nodes in common respectively are its neighbors in G_{mix}. For a 4-node CIS, all 3-node, 4-node, and 5-node CISes having 3, 3, and 4 nodes in common respectively are its neighbors in G_{mix}. For a 5-node CIS all 4-node and 5-node CISes with 4 nodes in common are its neighbors in G_{mix}. To estimate subgraph frequency distribution, GUISE performs a Metropolis-Hastings based sampling method over G_{mix}. Hardiman and Katzir [Hardiman and Katzir 2013] propose random walk based sampling methods for estimating the network average and global clustering coefficients. Gjoka et al. [Gjoka et al. 2013] propose a uniform node sampling based method for estimating the clique (i.e., complete subgraph) size distribution. The methods in [Hardiman and Katzir 2013; Gjoka et al. 2013] are difficult to extend to measure concentrations of subgraph classes.

7. CONCLUSIONS

In this paper we propose two random walk based sampling methods to estimate subgraph class concentrations when the complete graph topology is not available. The experimental results show that our methods PSRW and SRW only need to sample a very small fraction of subgraphs to obtain an accurate and unbias estimate, and significant reduces the number of samples required to achieve the same estimation accuracy of state-of-the-art methods such as FANMOD. Also, simulation results show that PSRW is much more accurate and computational efficient than SRW.

Appendix

LEMMA 7.1. *When a graph $G = (V, E)$ is connected, for each node $v \in V$, we can generate a $(k + 1)$-node tree with a root v that contains $min\{d(v), k\}$ neighbors, where $d(v)$ is the degree of v in graph G, and $1 \leq k \leq |V| - 1$.*

PROOF. One can use breadth-first search (BFS) to traverse G starting from v, then build a tree from the first k nodes visited by BFS, where $2 \leq k \leq |V|$. This tree clearly contains $min\{d(v), k\}$ neighbors of v and the root node is v. $\square$

LEMMA 7.2 ([ROBERTS AND ROSENTHAL 2004; JONES 2004; LEE ET AL. 2012]).   *Let $G = (V, E)$ be connected and non-bipartite. Let u_j be the j-th node sampled by a RW on G, where $1 \leq j \leq B$ and B be the number of samples. Denote by $\boldsymbol{\pi} = (\pi_v, v \in V)$ the stationary distribution, where $\pi_v = \frac{d_v}{2|E|}$. Then, for any function $f(v) : V \to \mathbb{R}$, where $\sum_{\forall v \in V} f(v) < \infty$, we*

have

$$\lim_{B \to \infty} \frac{1}{B} \sum_{j=1}^{B} f(u_j) \xrightarrow{a.s.} \frac{1}{|V|} \sum_{\forall v \in V} f(v)\pi_v.$$

LEMMA 7.3 ([MEYN AND TWEEDIE 2009; RIBEIRO AND TOWSLEY 2010]). *Let* $G = (V, E)$ *be an undirected graph which is connected and non-bipartite. Let* (u_j, v_j) $(1 \le j \le B)$ *be the j-th edge sampled by a RW, where B is the number of sampled edges. Denote function* $f(u, v) : V \times V \to \mathbb{R}$. *Then, we have*

$$\lim_{B \to \infty} \frac{1}{B} \sum_{j=1}^{B} f(u_j, v_j) \xrightarrow{a.s.} \frac{1}{|E|} \sum_{\forall (u,v) \in E} f(u, v),$$

for any function f with $\sum_{\forall (u,v) \in E} f(u, v) < \infty$. □

7.1. Proof of Theorem 3.1

We use induction to prove $G^{(k)}$ is connected.

Initial Step. Since G is connected, clearly there exists a path (edge sequence) between any two disconnected edges. Therefore $G^{(2)}$ is connected. *Inductive Step*. Our inductive assumption is that $G^{(k)}$ is connected, $2 \le k \le |V| - 2$. We now prove that $G^{(k+1)}$ is also connected. For any two different CISes $x^{(k+1)}$ and $y^{(k+1)}$ in $C^{(k+1)}$, from Lemma 7.1 we can easily show that there exists a k-node CIS $x^{(k)}$ contained by $x^{(k+1)}$, and a k-node CIS $y^{(k)}$ contained by $y^{(k+1)}$. When $x^{(k+1)}$ and $y^{(k+1)}$ are not connected, our inductive assumption shows that there exists a k-node CIS sequence $s_i^{(k)}$ $(1 \le i \le l)$ in graph $G^{(k)}$, where $s_1^{(k)}$ connects to $x^{(k)}$, $s_l^{(k)}$ connects to $y^{(k)}$, and two adjacent k-node CIS $s_i^{(k)}$ and $s_{i+1}^{(k)}$ are connected, where $2 \le i < l$. Denote by $s_1^{(k+1)}$ the $(k+1)$-node CIS consisting of $k+1$ different nodes appearing in $s_1^{(k)}$ and $x^{(k)}$, $s_{l+1}^{(k+1)}$ the $(k+1)$-node CIS consisting of $k+1$ different nodes appearing in $s_l^{(k)}$ and $y^{(k)}$, and $s_i^{(k+1)}$ the $(k+1)$-node CIS consisting of $k+1$ different nodes appearing in $s_i^{(k)}$ and $s_{i+1}^{(k)}$, where $2 \le i < l$. In graph $G^{(k+1)}$, we can easily find that $s_1^{(k+1)}$ connects to $x^{(k+1)}$, $s_{l+1}^{(k+1)}$ connects to $y^{(k+1)}$, and two adjacent $(k+1)$-node CISes $s_i^{(k+1)}$ and $s_{i+1}^{(k+1)}$ $(1 \le i \le l)$ are connected. This shows that there exists a path between any two disconnected nodes $((k+1)$-node CISes) in $G^{(k+1)}$. Therefore graph $G^{(k+1)}$ is connected.

7.2. Proof of Theorem 3.2

Denote by v the node with degree larger than two. Lemma 7.1 indicates that there exists a k-node tree t with root v which contains at least three neighbors of v, where $4 \le k \le |V|$. We easily find that t has at least three leaves. Since t is still connected after we remove any leaf, there exist at least three different $(k-1)$-node CISes consisting of $k-1$ nodes in t obtained by removing one leaf of t, and these CISes are connected to each other in graph $G^{(k-1)}$. Similarly there exist at least three $(k-2)$-node CISes consisting of $k-2$ nodes in t by excluding two leaves of t, which are connected to each other in $G^{(k-2)}$. Therefore, each $G^{(k)}$ $(2 \le k < |V|)$ is non-bipartite since it has at least one odd length loop. When G has no node with degree larger than two, since G is connected and non-bipartite, we can easily show that G is a $|V|$-node circle and $|V|$ is odd. For each node $v \in V$, we can generate a k-node CIS consisting of v and $k-1$ nodes close to v in clockwise direction, where $2 \le k < |V|$. Finally there are $|V|$ different k-node CISes, and they form an odd length loop in graph $G^{(k)}$. Therefore $G^{(k)}$ is non-bipartite.

7.3. Proof of Theorem 3.4

SRW can be viewed as a regular RW over graph $G^{(k)}$, $2 \le k < |V|$. For each $\omega_i^{(k)}$, $1 \le i \le T_k$, we then obtain following equations from Lemma 7.2 and Theorem 3.3 for non-bipartite and connected

$G^{(k)}$,

$$\lim_{B \to \infty} \frac{1}{B} \sum_{j=1}^{B} \frac{\mathbf{1}(C(s_j) = C_i^{(k)})}{d^{(k)}(s_j)}$$

$$\xrightarrow{a.s.} \frac{1}{|C^{(k)}|} \sum_{\forall s \in C^{(k)}} \frac{\mathbf{1}(C(s) = C_i^{(k)})}{d^{(k)}(s)} \pi^{(k)}(s)$$

$$= \frac{1}{|C^{(k)}| \sum_{t \in C^{(k)}} d^{(k)}(t)} \sum_{\forall s \in C^{(k)}} \mathbf{1}(C(s) = C_i^{(k)})$$

$$= \frac{\omega_i^{(k)}}{\sum_{t \in C^{(k)}} d^{(k)}(t)}.$$

Similarly, we have

$$\lim_{B \to \infty} \frac{1}{B} \sum_{j=1}^{B} \frac{1}{d^{(k)}(s_j)} \xrightarrow{a.s.} \frac{1}{\sum_{t \in C^{(k)}} d^{(k)}(t)}.$$

Thus, we can easily find that $\hat{\omega}_i^{(k)}$ ($1 \le i \le T_k$) is an asymptotically unbiased estimator of $\omega_i^{(k)}$.

7.4. Proof of Theorem 3.5

To estimate $\tilde{\omega}_i^{(k)}$, $1 \le i \le T_k$, $2 \le k < |V|$, PSRW can be viewed as a regular RW over the graph $G^{(k-1)}$. Denote $s_{(u,v)}^*$ as the k-node CIS generated by $(u,v) \in R^{(k-1)}$, an edge in $G^{(k-1)}$, where $u, v \in C^{(k-1)}$ are $(k-1)$-node CISes. For each $\omega_i^{(k)}$, $1 \le i \le T_k$, we then obtain following equations from Lemma 7.3,

$$\lim_{B \to \infty} \frac{1}{B-1} \sum_{j=1}^{B-1} \frac{\mathbf{1}(C(s_j^*) = C_i^{(k)})}{I^{(k-1)}(s_j^*) \left(I^{(k-1)}(s_j^*) - 1 \right)}$$

$$\xrightarrow{a.s.} \frac{1}{|R^{(k-1)}|} \sum_{\forall (u,v) \in R^{(k-1)}} \frac{\mathbf{1}(C(s_{(u,v)}^*) = C_i^{(k)})}{I^{(k-1)}(s_{(u,v)}^*) \left(I^{(k-1)}(s_{(u,v)}^*) - 1 \right)}$$

$$= \frac{1}{2|R^{(k-1)}|} \sum_{\forall s \in C^{(k)}} \mathbf{1}(C(s) = C_i^{(k)})$$

$$= \frac{\omega_i^{(k)} |C^{(k)}|}{2|R^{(k-1)}|}.$$

The last equation holds because the k-node CIS s is generated by $\frac{\left(I^{(k-1)}(s) \right)\left(I^{(k-1)}(s) - 1 \right)}{2}$ edges in $R^{(k-1)}$. Similarly, we have

$$\lim_{B \to \infty} \frac{\sum_{j=1}^{B-1} \frac{1}{I^{(k-1)}(s_j^*)\left(I^{(k-1)}(s_j^*) - 1 \right)}}{B-1} \xrightarrow{a.s.} \frac{|C^{(k)}|}{2|R^{(k-1)}|}.$$

Thus, we can easily find that $\tilde{\omega}_i^{(k)}$ ($1 \le i \le T_k$) is an asymptotically unbiased estimator of $\omega_i^{(k)}$.

7.5. Proof of Theorem 3.6

For each $\omega_i^{(k-1)}$, $1 \leq i \leq T_{k-1}$, we obtain following equations from Lemma 7.2 and Theorem 3.3 for non-bipartite and connected $G^{(k)}$,

$$\lim_{B \to \infty} \frac{1}{B} \sum_{j=1}^{B} \frac{1}{d^{(k)}(s_j)} \sum_{s' \in C^{(k-1)}(s_j)} \frac{\mathbf{1}(C^{(k-1)}(s') = C_i^{(k-1)})}{|O^{(k)}(s')|}$$

$$\xrightarrow{a.s.} \frac{1}{|C^{(k)}|} \sum_{\forall s \in C^{(k)}} \frac{\pi^{(k)}(s)}{d^{(k)}(s)} \sum_{s' \in C^{(k-1)}(s)} \frac{\mathbf{1}(C^{(k-1)}(s') = C_i^{(k-1)})}{|O^{(k)}(s')|}$$

$$= \frac{1}{|C^{(k)}| \sum_{t \in C^{(k)}} d^{(k)}(t)} \sum_{\forall s \in C^{(k)}} \sum_{s' \in C^{(k-1)}(s)} \frac{\mathbf{1}(C^{(k-1)}(s') = C_i^{(k-1)})}{|O^{(k)}(s')|}$$

$$= \frac{1}{|C^{(k)}| \sum_{t \in C^{(k)}} d^{(k)}(t)} \sum_{\forall s' \in C^{(k-1)}} \sum_{s \in O^{(k)}(s')} \frac{\mathbf{1}(C^{(k-1)}(s') = C_i^{(k-1)})}{|O^{(k)}(s')|}$$

$$= \frac{\omega_i^{(k-1)} |C^{(k-1)}|}{|C^{(k)}| \sum_{t \in C^{(k)}} d^{(k)}(t)}.$$

Similarly, we have

$$\lim_{B \to \infty} \frac{\sum_{j=1}^{B} \frac{1}{d^{(k)}(s_j)} \sum_{s' \in C^{(k-1)}(s_j)} \frac{1}{|O^{(k)}(s')|}}{B} \xrightarrow{a.s.} \frac{|C^{(k-1)}|}{|C^{(k)}| \sum_{t \in C^{(k)}} d^{(k)}(t)}.$$

Thus, we can easily find that $\breve{\omega}_i^{(k-1)}$ ($1 \leq i \leq T_{k-1}$) is an asymptotically unbiased estimator of $\omega_i^{(k-1)}$.

References

István Albert and Réka Albert. 2004. Conserved network motifs allow protein–protein interaction prediction. *Bioinformatics* 4863, 13 (2004), 3346–3352.

Mansurul A Bhuiyan, Mahmudur Rahman, Mahmuda Rahman, and Mohammad Al Hasan. 2012. GUISE: Uniform Sampling of Graphlets for Large Graph Analysis. In *Proceedings of IEEE ICDM 2012*. 91–100.

Jin Chen, Wynne Hsu, Mong-Li Lee, and See-Kiong Ng. 2006. NeMoFinder: dissecting genome-wide protein-protein interactions with meso-scale network motifs. In *Proceedings of ACM SIGKDD 2006*. 106–115.

James Cheng, Yiping Ke, Ada Wai-Chee Fu, Jeffrey Xu Yu, and Linhong Zhu. 2011. Finding maximal cliques in massive networks. *ACM Transactions on Database Systems* 36, 4 (dec 2011), 21:1–21:34.

Shumo Chu and James Cheng. 2012. Triangle listing in massive networks. *ACM Transactions on Knowledge Discovery from Data* 6, 4, Article 17 (December 2012), 32 pages.

Hyunwoo Chun, Yong yeol Ahn, Haewoon Kwak, Sue Moon, Young ho Eom, and Hawoong Jeong. 2008. Comparison of Online Social Relations in Terms of Volume vs. Interaction: A Case Study of Cyworld. In *Proceedings of ACM SIGCOMM Internet Measurement Conference 2008*. 57–59.

Minas Gjoka, Maciej Kurant, Carter T Butts, and Athina Markopoulou. 2011. Practical recommendations on crawling online social networks. *Selected Areas in Communications, IEEE Journal on* 29, 9 (2011), 1872–1892.

Minas Gjoka, Emily Smith, and Carter T. Butts. 2013. Estimating Clique Composition and Size Distributions from Sampled Network Data. *ArXiv e-prints* (Aug. 2013).

Stephen J. Hardiman and Liran Katzir. 2013. Estimating clustering coefficients and size of social networks via random walk. In *Proceedings of the 22nd international conference on World Wide Web (WWW 2013)*. 539–550.

Mohammad Al Hasan and Mohammed J. Zaki. 2009. Output Space Sampling for Graph Patterns. In *Proceedings of the VLDB Endowment 2009*. 730–741.

Shalev Itzkovitz, Reuven Levitt, Nadav Kashtan, Ron Milo, Michael Itzkovitz, and Uri Alon. 2005. Coarse-Graining and Self-Dissimilarity of Complex Networks. *Physica Rev.E* 71 (2005), 016127.

Galin L. Jones. 2004. On the Markov chain central limit theorem. *Probability Surveys* 1 (2004), 299–320.

Zahra Razaghi Moghadam Kashani, Hayedeh Ahrabian, Elahe Elahi, Abbas Nowzari-Dalini, Elnaz Saberi Ansari, Sahar Asadi, Shahin Mohammadi, Falk Schreiber, and Ali Masoudi-Nejad. 2009. Kavosh: a new algorithm for finding network motifs. *BMC Bioinformatics* 10 (2009), 318.

Nadav Kashtan, Shalev Itzkovitz, Ron Milo, and Uri Alon. 2004. Efficient sampling algorithm for estimating subgraph concentrations and detecting network motifs. *Bioinformatics* 20, 11 (2004), 1746–1758.

Liran Katzir, Edo Liberty, and Oren Somekh. 2011. Estimating Sizes of Social Networks via Biased Sampling. In *Proceedings of WWW 2011*. 597–606.

Jérôme Kunegis, Andreas Lommatzsch, and Christian Bauckhage. 2009. The slashdot zoo: mining a social network with negative edges. In *Proceedings of WWW 2009*. 741–750.

Haewoon Kwak, Changhyun Lee, Hosung Park, and Sue Moon. 2010. What is Twitter, a Social Network or a News Media?. In *Proceedings of WWW 2010*. 591–600.

Chul-Ho Lee, Xin Xu, and Do Young Eun. 2012. Beyond Random Walk and Metropolis-Hastings Samplers: Why You Should Not Backtrack for Unbiased Graph Sampling. In *Proceedings of ACM SIGMETRICS/Performance 2012*. 319–330.

Jure Leskovec, Kevin J. Lang, Anirban Dasgupta, and Michael W. Mahoney. 2009. Community Structure in Large Networks: Natural Cluster Sizes and the Absence of Large Well-Defined Clusters. *Internet Mathematics* 6, 1 (2009), 29–123.

L. Lovász. 1993. Random walks on graphs: a survey. *Combinatorics* 2 (1993), 1–46. Issue Paul Erdös is Eighty.

Brendan D. McKay. 1981. Practical Graph Isomorphism. *Congressus Numerantium* 30 (1981), 45–87.

Brendan D. McKay. 2009. *nauty User's Guide, Version 2.4*. Technical Report. Department of Computer Science, Australian National University.

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

R. Milo, Et Al, and Cell Biology. 2002. Network motifs: Simple building blocks of complex networks. *Science* 298, 5549 (October 2002), 824–827.

Alan Mislove, Massimiliano Marcon, Krishna P. Gummadi, Peter Druschel, and Bobby Bhattacharjee. 2007. Measurement and Analysis of Online Social Networks. In *Proceedings of ACM SIGCOMM Internet Measurement Conference 2007*. 29–42.

Michael Molloy and Bruce Reed. 1995. A critical point for random graphs with a given degree sequence, Random Structures & Algorithms. *Random Structures and Algorithms* 6, 2-3 (1995), 161–179.

Saeed Omidi, Falk Schreiber, and Ali Masoudi-nejad. 2009. MODA: An efficient algorithm for network motif discovery in biological networks. *Genes and Genet systems* 84, 5 (2009), 385–395.

Bruno Ribeiro and Don Towsley. 2010. Estimating and Sampling Graphs with Multidimensional Random Walks. In *Proceedings of ACM SIGCOMM Internet Measurement Conference 2010*. 390–403.

Bruno Ribeiro, Pinghui Wang, Fabricio Murai, and Don Towsley. 2012. Sampling Directed Graphs with Random Walks. In *Proceedings of IEEE INFOCOM 2012*. 1692–1700.

Bruno F. Ribeiro and Don Towsley. 2012. On the estimation accuracy of degree distributions from graph sampling. In *CDC*. 5240–5247.

Matthew Richardson, Rakesh Agrawal, and Pedro Domingos. 2003. Trust Management for the Semantic Web. In *Proceedings of the 2nd International Semantic Web Conference*. 351–368.

Gareth O. Roberts and Jeffrey S. Rosenthal. 2004. General state space Markov chains and MCMC algorithms. *Probability Surveys* 1 (2004), 20–71.

Shai S. Shen-Orr, Ron Milo, Shmoolik Mangan, and Uri Alon. 2002. Network motifs in the transcriptional regulation network of Escherichia coli. *Nature Genetics* 31, 1 (May 2002), 64–68.

Lubos Takac and Michal Zabovsky. 2012. Data Analysis in Public Social Networks.. In *International Scientific Conference and International Workshop Present Day Trends of Innovations*. 1–6.

Johan Ugander, Lars Backstrom, and Jon Kleinberg. 2013. Subgraph frequencies: mapping the empirical and extremal geography of large graph collections. In *Proceedings of the 22nd international conference on World Wide Web (WWW 2013)*. 1307–1318.

Sebastian Wernicke. 2006. Efficient Detection of Network Motifs. *IEEE/ACM Transactions on Computational Biology and Bioinformatics* 3, 4 (2006), 347–359.

Junzhou Zhao, John C. S. Lui, Don Towsley, Xiaohong Guan, and Yadong Zhou. 2011. Empirical Analysis of the Evolution of Follower Network: A Case Study on Douban. In *Proceedings of IEEE INFOCOM NetSciCom 2011*. 941–946.