Comparing Graph Sampling Methods Based on the Number of Queries

Kenta Iwasaki and Kazuyuki Shudo
Tokyo Institute of Technology

***Abstract*—Random walk-based graph sampling methods can effectively estimate feature values in large-scale social networks wherein the node IDs are unknown. Real social networks are sampled by repeatedly querying their APIs to acquire the lists of adjacent nodes. These queries can then become a bottleneck in the sampling process because nearly all social network services restrict the rate at which queries can be issued. However, most existing graph sampling studies have not focused on the number of queries but have instead compared methods based on sample size. Therefore, such graph sampling methods cannot be recommended for estimating feature values in actual social networks. This study presents an approach to assess graph sampling methods that focus on the number of queries. This describes the time taken by algorithms for typical random walk-based graph sampling methods, such as a simple random walk with re-weighting (SRW-rw), a non-backtracking random walk with re-weighting (NBRW-rw), and a Metropolis - Hastings random walk (MHRW), which require queries. The graph sampling precision was then experimentally evaluated based on sample size and query number standards using actual social networks, and the types of changes that occur were observed.**

***Index Terms*—graph sampling, social graph, random walk**

I. Introduction

The sampling of large-scale networks is a fundamental and important issue in analyzing social networks, such as online social networks (OSNs) and the World Wide Web. Several studies have used sampling for estimating network feature values when working with huge networks, wherein the entire network cannot be considered or when general network information, such as node IDs, cannot be obtained [1] [2] [3]. When sampling is used to extract sub-graphs, (i.e., graphs that model parts of the network structure as sets of nodes and edges), it is called graph sampling [4].

Since general network information, such as node IDs, is not typically published; therefore, estimating social networks' feature values can be difficult. For example, uniform independent sampling, which randomly samples node IDs, cannot be used because the node IDs are unknown. Graph sampling methods that track adjacency relations, known as crawling methods, are therefore more practical [5]. Among these crawling methods, random walk-based methods are particularly effective because their algorithms are simple and easy to implement. In addition, they allow us to conduct unbiased graph sampling using Markov chain analysis. Random walks are not the only possible approach: scanning methods, such as breadth-first sampling (BFS) [6], have also been proposed; however, since their sampling deviations are unknown, they are unsuitable for estimating feature values.

When applying a crawling method to an actual social network, the sampling process involves repeatedly either querying, e.g., the OSN' s API, or scraping. Herein, each query acquires a list of nodes adjacent to the current Internet-connected node. In the example shown in Figure 1, the researcher is using a query to acquire a list of adjacent nodes. As another example, a previous study [2] showed how Facebook could be sampled by repeatedly acquiring the friend lists of Facebook users. In the sampling process, the acquisition of the adjacent node list with a query can become a bottleneck because with many social networks, the number of queries that can be used within a fixed unit of time is restricted. In addition, even if this were not restricted, given the fact that access via communications is slower than the speed of accessing computer memory or a disk, the acquisition of the adjacent node list via the query may still become a bottleneck.

Therefore, the performance of the graph sampling method should be compared based on the number of queries. However, comparisons of the estimation accuracy using the graph sampling method are based on existing studies [7], [8], which often involve experiments based on sample sizes (the length of a sample sequence); thus, recommending an appropriate method for sampling actual social networks can be difficult.

This study proposes a comparison of sampling methods that focuses on the number of queries. This comparison describes the timing within the algorithm for typical random-walk-based graph sampling methods, such as a simple random walk with re-weighting (SRW-rw) [5], [9], a Metropolis - Hastings random walk (MHRW) [5], [9], [10], and a non-backtracking random walk with re-weighting (NBRW-rw) [7], which require queries. The feature values are estimated based on the number of queries for actual social networks, and the performance of the graph sampling methods was compared.

This paper is organized as follows. Section II describes the background for this study and explains the notation and definition of terms. Section III describes the graph sampling method using the random walk method. Section IV explains the computational experiment conducted herein. Section V discusses the related research, and Section VI summarizes the study.

This work was supported by JSPS KAKENHI Grant Numbers 2570008 and 16K12406. This work was supported by New Energy and Industrial Technology Development Organization (NEDO).

Proc. IEEE ISPA-IUCC-BDCloud-SocialCom-SustainCom 2018,
pp.1136-1143, December 2018

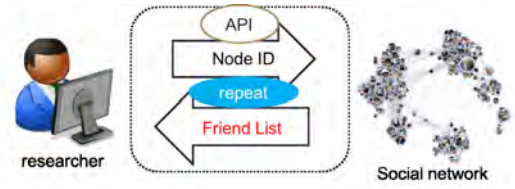


Fig. 1: Example of acquiring an adjacent node list via the API.

II. Preparation

In this section, after explaining the method of notation and the definitions of the graphs, the graph feature values used in this study are described. In addition, existing graph sampling methods and existing methods of estimating clustering coefficients are described.

A. Notation

Herein, social networks are expressed as an undirected graph $G(V, E)$. $V = \{v_1, v_2, ..., v_n\}$ is a set of nodes (term points), and the number of nodes for the graph as a whole is expressed as $n = |V|$. E is the edge set. The adjacent node set to node $v \in V$ is $N(v) = \{w \in V : (v, w) \in E\}$. The degree of node v is set to $d(v) = |N(v)|$.

B. Graph feature values

The typical feature values that characterize complex networks are described herein. Complex networks are defined as real-world networks with a huge, complex structure. Nearly all social networks can be defined as complex networks. The degree distribution and clustering coefficient estimation error are used herein.

1) Degree distribution: If the node ratio for degree k for all nodes is expressed as $p(k)$, the degree distribution can be defined. In social networks, the degree distribution follows the power law. In other words, it becomes $p(k) \propto k^{-\gamma}$. This is called a scale-free property, which shows that although nearly all nodes have a small degree, some nodes have a large degree. These nodes with a large degree are frequently called hubs .

2) Clustering coefficient: The clustering coefficient is an important feature value in the network and is often used in network analyses [2]. In a complex network, a triangle is called a cluster . The term cluster normally refers to a herd or a group and is used in several ways in research analyses, such as cluster analysis or cluster synchronicity. In this study, it is only used to mean triangle. Many clusters are found within most real networks, and this is not limited to networks of human relations . To define a network clustering coefficient, the clustering coefficient $c(v)$ from the number of triangles, including a node v, is first defined. A method of selecting two nodes from v neighboring nodes of which there are $d(v)$ is as shown in $d(v)(d(v) - 1)/2$. Based on this, where there are edges between the two selected nodes, there is one triangle including these two nodes and the node v. Therefore, a maximum of $d(v)(d(v) - 1)/2$ triangles include v Here, if the number of triangles including v is $\triangle_i$, the clustering coefficient $c(v)$ can be defined as follows.

$$c(v) = \begin{cases} 0 & d(v) = 0 or d(v) = 1 \\ \dfrac{2\triangle_i}{d(v)(d(v)-1)} & \text{otherwise} \end{cases} \quad (1)$$

From the definition of the clustering coefficient, $c(v) \in [0, 1]$

The clustering coefficient C for the entire network is defined as the mean value of the clustering coefficients for each node.

$$C = \frac{1}{n} \sum_{v \in V} c(v) \quad (2)$$

where $c(v)$ is a value for the node and C is the value for the graph G.

This is $C \in [0, 1]$ in relation to any network, and this is $C = 1$ in relation to a complete graph. In nearly all actual networks, C is large; this is a characteristic of complex networks . In this study, we estimate C for each random walk using the naive and counting triangles method.

III. Random walk-based graph sampling

This section explains the algorithm and impartiality of the graph sampling method using the random walk method. First, the impartial graph sampling as a base is explained, followed by the algorithm and the method of estimation using each method. In this study, three typical methods of SRW-rw, NBRW-rw, and MHRW are used.

A. Impartial sampling

When estimating the feature values focusing on the social network node or topology, it is necessary to perform impartial sampling using crawling. Thus, uniform node samples are obtained via a random walk. In this section, our goal is to impartially estimate the ratio of nodes with specific features.

In other words, impartial sampling involves constructing a method of sampling using a random walk to obtain the expectation values related to a uniform distribution of an arbitrary function $f : V \to \mathbb{R}$. That is, it is used as a method of sampling, when expressing the uniform distribution as $\mathbf{u} \stackrel{\text{def}}{=} [u(1), u(2), \ldots, u(n)] = [1/n, 1/n, \ldots, 1/n]$, in which

$$\mathbb{E}_{\mathbf{u}}(f) \stackrel{\text{def}}{=} \sum_{v \in V} f(v) \frac{1}{n} \quad (3)$$

produces the expected estimated value. By selecting the function appropriately, it is possible to specify the feature values for the desired node. For example, when estimating the degree distribution ($\mathbb{P}\{D_G = d\}, d = 1, 2, \ldots, n-1$) of a graph G, $f(v) = \mathbb{1}_{\{d(v)=d\}}$ for $v \in V$. We select the function f so that if $d(v) = d$, $f(v) = 1$; otherwise, $f(v) = 0$.

Next, we discuss Markov chain theory, which is the numerical foundation for impartial sampling using a random walk on graph G. A typical random walk on graph G or an irreducible finite Markov chain $\{X_t \in V, t = 0, 1...\}$

with reversibility is defined as having the following transition probability matrix $\mathbf{P} \stackrel{\text{def}}{=} \{P(v,w)\}_{v,w\in V}$

$$P(v,w) = \mathbb{P}\{X_{t+1} = w \mid X_t = v\}, v, w \in V, \tag{4}$$

For $\forall v \in V$, $\sum_{w\in V} P(v,w) = 1$. The transition probability $P(v,w) \geq 0$ is allocated to each edge $(v,w) \in E$, and it is possible for the random walk to transition from node v to node w. In addition, even when there is no self-loop in graph G, the transition to the self-node can be set. For $v \in V$, $P(v,v) > 0$ may exist. However, transitions between nodes without edges are not possible. In other words, $P(v,w) = 0, \forall (v,w) \notin E\,(v \neq w)$.

The stationary distribution is $\boldsymbol{\pi} = [\pi(v), v \in V]$. The following estimator is defined for an arbitrary function $f : V \to \mathbb{R}$.

$$\hat{\mu}_t(f) \stackrel{\text{def}}{=} \frac{1}{t}\sum_{s=1}^{t} f(X_s) \tag{5}$$

Function f expectation values for the stationary distribution $\boldsymbol{\pi}$ are given as follows.

$$\mathbb{E}_{\boldsymbol{\pi}}(f) \stackrel{\text{def}}{=} \sum_{i\in V} \pi(i) f(i). \tag{6}$$

From Ref. [11], where $\{X_t\}$ is a stationary distribution $\boldsymbol{\pi}$ finite, irreducible Markov chain,

$$\hat{\mu}_t(f) \to \mathbb{E}_{\boldsymbol{\pi}}(f)\ almost\,surely\,(a.s.) \tag{7}$$

is formed for an arbitrary initial distribution $\mathbb{P}\{X_0 = v\}, v \in V$, $(t \to \infty)$. However, $\mathbb{E}_{\boldsymbol{\pi}}(|f|) < \infty$.

B. SRW-rw

SRW-rw is a sampling algorithm that includes a method of calculating the estimated value from the sample node collection, whereas simple random walk (SRW) expresses a simple random walk transition algorithm. This method is used based on a sample sequence obtained from SRW and an appropriate re-weighting process for achieving impartial sampling. This is essentially a special case for weighted sampling applied to the random samples generated by the Markov chain. The basic idea of this method is that the deviation in the sampling that occurs because of SRW stationary distribution is corrected via re-weighting.

Here, we explain the method of acquiring graph G of the SRW sample sequence. The Markov chain expressing the sample sequence for the nodes visited by SRW is $\{X_t\}$. If this transition probability matrix is $\mathbf{P}^{SRW} = P^{SRW}(v,w)_{v,w\in V}$, then $P^{SRW}(v,w)$ can be expressed as

$$P^{SRW}(v,w) = \begin{cases} \frac{1}{d(v)} & (v,w) \in E \\ 0 & \text{otherwise} \end{cases} \tag{8}$$

The transition probability $\mathbf{P}^{SRW}$ is irreducible, and the stationary distribution $\pi^{SRW}(v) = d(v)/(2|E|), v \in V$ is known to be reversible.

Let us suppose that t samples $\{X_s\}_{s=1}^{t}$ are obtained from SRW. Then, for the arbitrary function $f : V \to \mathbb{R}$, the weighted function $w : V \to \mathbb{R}$ is determined as follows:

$$\hat{\mu}_t(wf) = \frac{1}{t}\sum_{s=1}^{t} w(X_s) f(X_s) \to \mathbb{E}_{\boldsymbol{\pi}}(wf) = \mathbb{E}_{\boldsymbol{u}}(f)\ a.s. \tag{9}$$

Because this is an irreducible finite Markov chain, when $t \to \infty$,

$$\hat{\mu}_t(wf) = \frac{1}{t}\sum_{s=1}^{t} w(X_s) f(X_s) \to \mathbb{E}_{\boldsymbol{\pi}}(wf) = \mathbb{E}_{\boldsymbol{u}}(f)\ a.s. \tag{10}$$

is a strongly consistent estimator. However, this is not practical because $|E|$ is not usually known in advance; therefore, the following type of estimator is often used. When $t \to \infty$,

$$\frac{\hat{\mu}_t(wf)}{\hat{\mu}_t(w)} = \frac{\sum_{s=1}^{t} w(X_s) f(X_s)}{\sum_{s=1}^{t} w(X_s)} \to \mathbb{E}_{\boldsymbol{u}}(f)\ a.s. \tag{11}$$

Then, by setting $w(v) = 1/d(v)$, an unbiased estimation can be conducted. In this study, $\hat{\mu}_t(wf)/\hat{\mu}_t(w), w(v) = 1/d(v)(v \in V)$ is treated as an unbiased estimator in SRW-rw.

As an example, we estimate the degree distribution using SRW-rw. To estimate the degree distribution $\mathbb{P}\{D_G = d\}$ for the target graph G, we select $f(v) = \mathbb{1}_{\{d(v)=d\}}$ for $v \in V$. Then, for an arbitrary degree distribution d,

$$\frac{\hat{\mu}_t(wf)}{\hat{\mu}_t(w)} = \frac{\sum_{s=1}^{t} \mathbb{1}_{\{d(X_s)=d\}}/d(X_s)}{\sum_{s=1}^{t} 1/d(X_s)} \to \sum_{v\in V} \mathbb{1}_{\{d(v)=d\}} \frac{1}{n}\ a.s.,$$

This demonstrates that the estimator $\hat{\mu}_t(wf)/\hat{\mu}_t(w)$ is a valid impartial estimation of the degree distribution $\mathbb{P}\{D_G = d\}$.

Queries are required within the SRW-rw algorithm when determining the transition destination nodes from the adjacent node list and when using degree information as weighting. Because the degree information is known from the visited node, the number of queries is equal to the number of unique nodes among those visited.

C. Non-backtracking Random Walk with Re-weighting

In this section, we discuss NBRW-rw [7]. NBRW-rw is based on a sample sequence obtained from nonbacktracking random walk (NBRW) and a suitable reweighting process to achieve impartial sampling. This demonstrates that for the reweighting process in the second part, the same process can be applied as in SRW-rw. In addition, we can see that the estimator using NBRW-rw produces a lower distribution value than the estimator using SRW-rw [7]. This section describes the NBRW transition method and provides an overview of the weighting process.

The NBRW transition method uses a random walk to uniformly and randomly select from adjacent nodes while avoiding a transition to the previous node. As an exception, this does not apply when there is an initial node and a degree-one node.

Now, we will explain the NBRW-rw re-weighting process. Let us denote the t step node visited with NBRW as $X'_t \in V$. Determining the next node X'_t from X'_{t+1} depends on X'_t and

on X'_{t-1} because of the algorithm, which avoids the previous node. Therefore, $\{X'_t\}_{t\geq 0}$ is not a Markov chain in the V node state space. However, we can see that this formed via the following formula from Ref. [7].

$$\frac{1}{t}\sum_{s=1}^{t} f(X'_s) \to \mathbb{E}_{\boldsymbol{\pi}}(f) a.s. \tag{12}$$

Because $\boldsymbol{\pi}$ is a stationary distribution of SRW, an impartial estimation can be conducted using the same weighting process as used in SRW [7] Queries are required within the NBRW-rw algorithm when determining the transition destination from the adjacent node list and when using the degree information as weighting. Similar to SRW-rw, because the degree information is known from the visited node, the number of queries is equal to the number of unique nodes among the visited.

D. Metropolis-Hastings Random Walk

Whereas with SRW or NBRW, samples tend to be biased toward high degree nodes, MHRW can appropriately transform the transition probability so that the stationary distribution is a uniform distribution. Because the Metropolis - Hastings (MH) algorithm [12] performs sampling from a complex probability distribution μ, this is a typical Markov chain Monte Carlo (MCMC) method. We can see that if we need to perform sampling from a uniform distribution $\mu_v = \frac{1}{n}$, as on this occasion, we can achieve this by defining the transition probability in the following way.

$$P^{MH}(v,w) = \begin{cases} \min(\frac{1}{d(v)}, \frac{1}{d(w)}) & (v,w) \in E \\ 1 - \sum_{y \neq v} P^{MH}(v,y) & w = v \\ 0 & \text{otherwise} \end{cases} \tag{13}$$

Then, the stationary distribution is $\pi^{MH}(v) = \frac{1}{n}$, which is a uniform distribution. Contrary to SRW, MHRW may sometimes transition to its own node. In this case, we add a new sample sequence. We can express this MH algorithm as in Algorithm 1. Then, $X_t \in V$ is the t-th node of MHRW, and X_0 is selected arbitrarily.

Algorithm 1 MH algorithm in MHRW (at time t).

randomly select node w from the adjacent node list $N(X_t)$
generate $p \sim U(0,1)$
if $p \leq \frac{d(X_t)}{d(w)}$ **then**
  $X_{t+1} \leftarrow v$
else
  $X_{t+1} \leftarrow X_t$
end if

It is notable that Algorithm 1 does not require obtaining its own transition probability $P^{MH}(v,v)$ and does not require knowing the degree of the nodes in the t step node X_t adjacent node list. Instead, if only the degree information for node w is selected at random, it is possible to select whether to transition to w.

Impartial estimations using MHRW, unlike SRW-rw, do not require reweighting calculations because the MHRW stationary distribution is a uniform distribution. If we assume that t samples $\{X_t\}_{s=1}^{t}$ are obtained from MHRW because there is an irreducible finite Markov chain for an arbitrary function $f : V \to \mathbb{R}$, when $t \to \infty$,

$$\frac{1}{t}\sum_{s=1}^{t} f(X_t) \to \mathbb{E}_{\boldsymbol{u}}(f)\, a.s., \tag{14}$$

Note that, as expressed in Algorithm1, in the case of self-transition, it is necessary to add one sample.

Queries are required within the MHRW algorithm when stopping at node v, when acquiring the node v adjacent list, and when used to obtain the degree information from the node v transition destination nodes. Therefore, the number of queries is basically the number of unique visited nodes; however, where a self-loop occurs, additional queries are required.

E. Graph sampling focusing on the number of queries

To estimate the feature values, graph sampling uses two types of timings at which the adjacent node list can be acquired as follows.

- When determining the next transition node using the crawling algorithm
- When calculating function $f(v)$ to estimate feature values, (from formula 5)

Crawling algorithm requires an adjacent node list acquisition query to determine the next transition destination node. In other words, for nodes that have been already visited, a query is always used once to acquire the adjacent node list of that node. With SRW and NBRW, the number of specific visited nodes is equal to the number of queries in the crawling algorithm. In the case of MHRW, when a self-loop occurs, there is a possibility that the query will be wasted; therefore, the number of specific visited nodes $\leq$ number of queries in the crawling algorithm. The adjacent node list of the nodes that are visited once can be reused when revisiting and when estimating feature values.

The number of queries when calculating function $f(v)$ to estimate feature values differs according to the feature values that are to be estimated. This can be broadly divided into cases wherein feature value estimation is possible by reusing the adjacent node list obtained via the query used when running the crawling algorithm, as well as feature values for which further queries are required. The former is a case wherein $f(v)$ can be calculated using the information from the list adjacent to node v. For example, clustering coefficient estimation uses average degree, degree distribution, and the counting triangles method. The latter equates to naive clustering coefficient estimation.

Figure 2 is an example of the node range using queries for calculation when performing naive clustering coefficient estimation, i.e., $f(v) = c(v)$. The blue nodes for the query range used within the crawling algorithm should acquire adjacent nodes with queries up to the yellow range to calculate the clustering coefficients. In this way, the necessary query

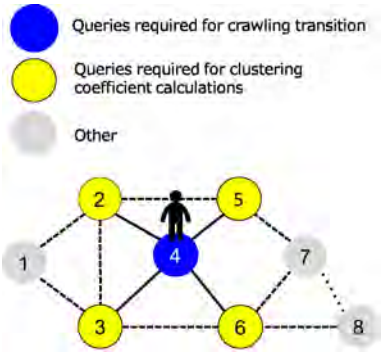
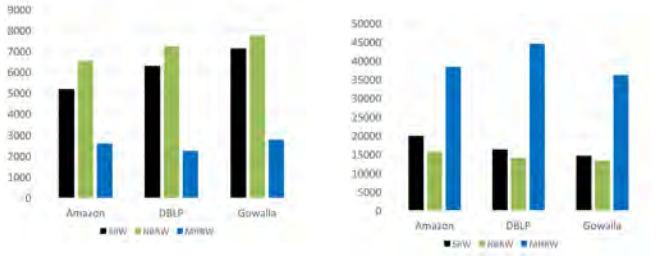


Fig. 2: Example of query range for obtaining adjacent node list.



(a) Mean number of queries required for a 10,000-sample size.

(b) Mean sample size per 10,000 queries.

Fig. 3: Relation between number of queries and sample size by random walk for each network.

range differs according to the feature value that has to be obtained via graph sampling. Incidentally, the range between the focal adjacent node, as well as the adjacent nodes of that adjacent node, is called the ego network.

In contrast, when estimating the same feature value, if the method of setting $f(v)$ changes, the necessary query range may change. To use the example of clustering coefficients, using the counting triangles method, this is $f(v) = \phi_k \cdot w(v)$ [8]. During this time, when ϕ_k is the k-th step, if an edge exists between the node in the k+1 step and the k - 1 node, the value of ϕ_k will be 1; and if the edge does not exist, the value will be 0. The definition of $w(v)$ changes based on the random walk; however, this is a function that determines the degree information. Therefore, the clustering coefficients can be estimated using only the queries required in the crawling algorithm.

IV. Experiments

The SRW-rw, NBRW-rw, and MHRW performance with the query number standard and sample size standard are described herein.

A. Data set

We performed an experiment using the social network/citation network publicized with the Stanford Network Analysis Project (SNAP) [13]. Table I shows an overview of each data set. Cases wherein graph sampling can practically occur are those in which the target social network is unknown; however, we performed a simulation in relation to graph data where the overall image is known.

TABLE I: Network statistical volume.

Network	Number of nodes	Average degrees	Average clustering coefficient
Amazon	334,863	5.530	0.3967
DBLP	317,080	6.622	0.6324
Gowalla	196,591	9.668	0.2367

B. Number of Queries by Random Walk

Figure 3 shows the mean number of queries and the sample size when simulating 100 random walks each for SRW, NBRW, and MHRW. Figure 6a shows the average number of queries required for the sample size (length of sample node sequence) to reach 10,000 times when sampling SRW, NBRW, and MHRW. Figure 6b shows the mean sample size to acquire 10,000 queries for each random walk. Based on the ascending width size value, these queries are NBRW, SRW, and MHRW.

C. Clustering coefficient estimation

In this section, the two types of clustering coefficient estimation experiments performed in Figures 4 and 5 are described. In both the experiments, we plotted the normalized root-mean-square error (NRMSE) [14] with the horizontal axis as the sample size (left side) and number of queries (right side). The lower the NRMSE value, the greater the estimation accuracy. NMRSE can be calculated $\frac{1}{C}\sqrt{\mathbb{E}[(\hat{C} - C)^2]}$, when $\hat{C}$ is the clustering coefficient estimated value.

In Figure 4, we fix the sampling method as SRW-rw. Then, we compare NRMSE as an approximate method using the naive estimation and counting triangles method. With Amazon, DBLP, we selected 100 as the starting point and performed the simulations independently. With Gowalla, we selected 10 as the starting point and performed the simulation independently. The naive method is defined as $f(v) = c(v)$, $v \in V$ in relation to function $f : V \to \mathbb{R}$ in formula 5. Here, $c(v)$ is the function defined in formula 1. In other words, we calculate the clustering coefficients as defined for each node visited in the random walk and perform impartial estimation using SRW-rw. To calculate a clustering coefficient for a certain node the adjacent node lists for that node and for those adjacent nodes should be acquired; consequently, queries other than random walk transitions occur.

In contrast, the counting triangles method is a method to estimate clusters without additional queries in relation to the queries required for random walk transitions. The details are described in Appendix A. In Figure 5, we fix the clustering coefficient estimation method as the counting triangles method and compare the NEMSE when changing the transition method via random walk. For each network, we selected 100 as the starting point and performed the simulation independently. We combined the SRW-rw, NBRW-rw, MHRW, and counting triangles method; in addition, since a combination of the

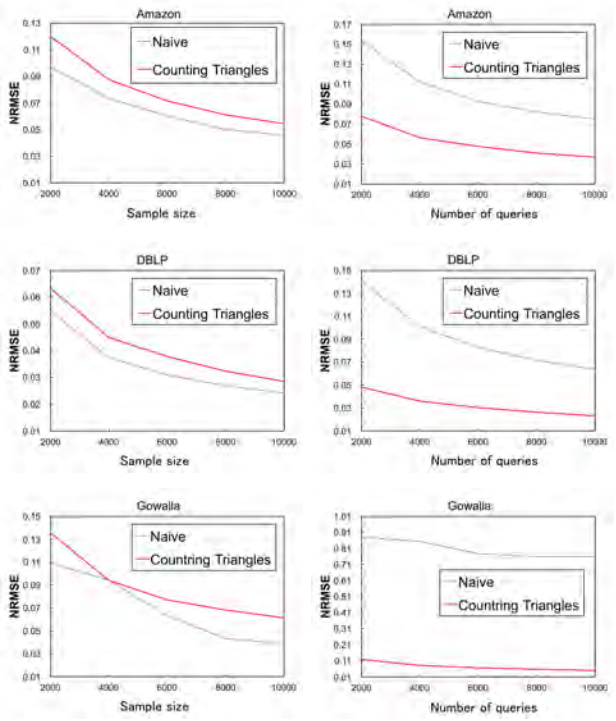


Fig. 4: NMRSE for clustering coefficient estimation using naive method and Counting Triangles method.

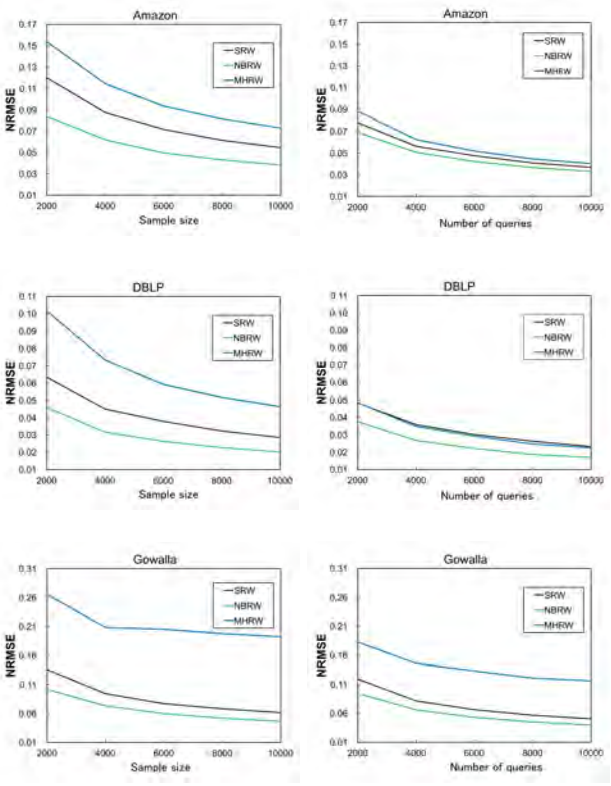


Fig. 5: NRMSE when estimating the clustering coefficient for random walk sequence using the Counting Triangles method.

MHRW and counting triangles method has not been proposed in the past, a new algorithm in this study has been devised herein, which is described in Appendix A.

D. Degree distribution estimation error

Degree distribution was estimated via random walk. We evaluated the complementary cumulative distribution function $\mathbb{P}\{D_g > d\}$(CCDF) for each respective network and compared SRW-rw, NBRW-rw, and MHRW. To estimate $\mathbb{P}\{D_g > d\}$, we defined $f(v) = \mathbb{1}_{\{d(v)>d\}}, v \in V$ and estimated them. Similar to the method for clustering coefficients, we compared estimation accuracy by calculating NRMSE. Here, the method of calculating NRMSE is $\frac{1}{x}\sqrt{\mathbb{E}[(\hat{x}(t)-x)^2]}$. $\hat{x}(t)$ is the estimated value when taking t samples, and x is the true value when taking t samples. During this time, in case of impartial estimation, $x = \lim_{t\to\infty} \hat{x}(t)$.

Figure 6 shows a simulation performed independently from a starting point of 100 for each network and plots the NRMSE for each method. The sample size standard accuracy is on the left side, whereas the query number standard accuracy is on the right side. The smaller the value, the greater the accuracy; NRMSE is calculated in the same way for MHRW estimation accuracy. However, since poor results were obtained when SRW-rw and NBRW-rw were separated, we displayed a graph comparing only SRW-rw and NBRW-rw.

E. Observations

As described in Section III-E, graph sampling methods were compared by focusing on the number of queries. From Figure 3 ,we can compare the number of queries required in the crawling algorithm for each random walk. Figure 4 fixes the crawling algorithm and shows an example of when the $f(v)$ setting is changed. Figures 4 and 5 show examples of fixing $f(v)$ and changing the crawling algorithm. From Figure 4, it is evident that the accuracy is reversed. With a query number standard, the NRMSE for the counting triangles method is smaller than that for the naive method; therefore, accuracy is determined to be good, whereas with a sample size standard, the naive method exhibits better accuracy because the counting triangles method calculates $f(v)$ in a probabilistic way. However, with the naive method, the clustering coefficient calculation is performed as defined because this method requires additional queries for calculating $f(v)$; thus, the counting triangles method produced better results compared with the query number standard. In this case, when considering sampling in actual social networks, the experiment results from the query number standard should be used.

The graphs on the left side of Figures 5 and 6 show a comparison of random walks based on the sample size standard, and for each result, the accuracy improved in the order of NBRW, SRW, and MHRW. The results of NBRW vs SRW for the sample size standard are described in [7], [8]. The sample size standard degree distribution NRMSE in Figure 6 generally has a lower value for NBRW than SRW, thus

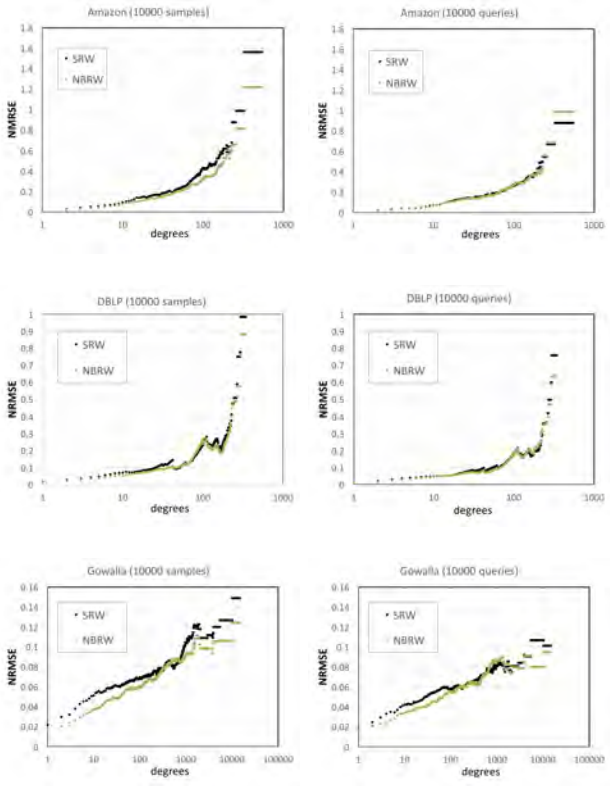


Fig. 6: NRMSE per degree d when estimating $\mathbb{P}\{D_g > d\}$.

leading to the conclusion that NBRW is more accurate than SRW. However, when looking at the query standard results, the NBRW and SRW results appear to have an antagonistic relation. As can be seen from Figure 3, the sample size per query is larger for SRW compared with that NBRW. In other words, when considering the query number standard, the SRW accuracy is reduced in relation to NBRW compared to the case of the sample size standard; notably, the SRW accuracy may even be reversed. In the same way in Figure 5, the accuracy of NBRW and SRW in the query number standard is smaller than in the case of sample size standard. Furthermore, the difference between the accuracy of MHRW and SRW is the same. With the sample size standard in Figure 5, SRW clearly exhibits a higher accuracy than MHRW; however, from the results of the query number standard, the difference in their accuracy is reduced and actually reversed with the experiment on DBLP.

V. Related Work

Herein, we describe related studies for graph sampling. Several studies have achieved impartial sampling from the network. Gjoka et al. [2] compared SRW-rw that achieves impartial sampling by reweighting using the standard distribution via SRW and MHRW using the MHRW algorithm, thereby demonstrating that SRW-rw shows superior accuracy. Lee et al. [7] proposed NBRW-rw derived from SRW-rw and showed that from a logical and experimental perspective, that it is superior to NBRW-rw impartiality and SRW-rw. In addition, several studies provide sampling techniques with the objective of estimating graph feature values. For methods specializing in the estimation of clustering coefficients, Hardiman and Katzir [15] [16] were the first to propose the counting triangles method as a method of approximation, as well as proposed a method combined with SRW-rw. Iwasaki et al. [8] proposed a graph sampling coefficient estimation method in combination with NBRW-rw and demonstrated that it was superior to SRW-rw. Combination of MHRW and the counting triangles method have not been previously proposed; thus, this is a novel study that proposes the combination of MHRW and counting triangles method with an overview of the algorithm shown in Appendix A. Chiericetti et al. [17] described query complexity in uniform sampling. Therein, rejection and maximum-degree sampling was used for MHRW and graph preinformation, which is different from the target method used in this study.

VI. Conclusion

For graph sampling in actual social networks, performance comparisons with the query number standard are important. However, experiments using query number standards were not performed previously because the methods recommended in those studies may not be valid methods in actual social networks. In addition, when proposing a new graph sampling method in the future, it will be necessary to incorporate the results of the performance comparison with query number standards.

In this study, after demonstrating the focus points when considering graph sampling based on the query number standard, we demonstrated the difference between accuracy using the traditional sampling size standard and the query number standard using an experiment. We also demonstrated an example of performing a query number standard experiment in relation to the new method of estimating clustering coefficients using a combination of MHRW and the counting triangles method. For the previous graph sampling methods, such as SRW-rw, NBRW-rw, and MHRW, with the sample size standard, the accuracy was higher in NBRW-rw, SRW-rw, and MHRW; however, by changing the query standard order, the difference between each method became smaller. It was demonstrated that this may be reversed depending on the target group and estimated feature values. Furthermore, we showed an example in which, even when the feature value estimation function $f(v)$ is changed by method, the results are reversed based on the sample size standard and query number standard.

Future issues include the logical expression of the relationship between the number of queries and sample size. The relationship between the sample size number and feature value estimation accuracy has been evidenced by past studies; so, it is expected that these results can be used to obtain the relation between the number of queries and feature value estimation accuracy.

References

[1] Y.Y. Ahn, S. Han, H. Kwak, S. Moon, and H. Jeong. Analysis of topological characteristics of huge online social networking services. In

Proceedings of the 16th international conference on World Wide Web, pp. 835–844. ACM, 2007.
[2] M. Gjoka, M. Kurant, C.T. Butts, and A. Markopoulou. Walking in Facebook: A case study of unbiased sampling of OSNs. In *Proceedings IEEE Infocom*, pp. 1–9. IEEE, 2010.
[3] A. Mislove, M. Marcon, K. P. Gummadi, P. Druschel, and B. Bhattacharjee. Measurement and analysis of online social networks. In *Proceedings of the 7th ACM SIGCOMM conference on Internet measurement*, pp. 29–42. ACM, 2007.
[4] J. Leskovec and C. Faloutsos. Sampling from large graphs. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 631–636. ACM, 2006.
[5] M. Gjoka, M. Kurant, C. T. Butts, and A. Markopoulou. Practical recommendations on crawling online social networks. *Selected Areas in Communications, IEEE Journal on*, Vol. 29, No. 9, pp. 1872–1892, 2011.
[6] M. Kurant, A. Markopoulou, and P.dd Thiran. Towards unbiased bfs sampling. *Selected Areas in Communications, IEEE Journal on*, Vol. 29, No. 9, pp. 1799–1809, 2011.
[7] C. H. Lee, X. Xu, and D. Y. Eun. Beyond random walk and metropolis-hastings samplers: why you should not backtrack for unbiased graph sampling. In *ACM SIGMETRICS Performance Evaluation Review*, Vol. 40, pp. 319–330, 2012.
[8] K. Iwasaki, K. Shudo. Estimating the clustering coefficient of a social network by a non-backtracking random walk. In *IEEE BigComp 2018*, pp. 114–118. IEEE, 2018.
[9] A. H. Rasti, M. Torkjazi, R. Rejaie, N. Duffield, W. Willinger, and D. Stutzbach. Respondent-driven sampling for characterizing unstructured overlays. In *INFOCOM 2009, IEEE*, pp. 2701–2705. IEEE, 2009.
[10] M. Al Hasan and M. J. Zaki. Output space sampling for graph patterns. *Proceedings of the VLDB Endowment*, Vol. 2, No. 1, pp. 730–741, 2009.
[11] G. L. Jones, et al. On the markov chain central limit theorem. *Probability surveys*, Vol. 1, pp. 299–320, 2004.
[12] W. K. Hastings. Monte carlo sampling methods using markov chains and their applications. *Biometrika*, Vol. 57, No. 1, pp. 97–109, 1970.
[13] Stanford large network dataset collection. https://snap.stanford.edu/data/.
[14] K. Avrachenkov, B. Ribeiro, and D. Towsley. Improving random walk estimation accuracy with uniform restarts. In *International Workshop on Algorithms and Models for the Web-Graph*, pp. 98–109. Springer, 2010.
[15] S. J. Hardiman and L. Katzir. Estimating clustering coefficients and size of social networks via random walk. In *Proceedings of the 22nd international conference on World Wide Web*, pp. 539–550. International World Wide Web Conferences Steering Committee, 2013.
[16] L. Katzir and S. J. Hardiman. Estimating clustering coefficients and size of social networks via random walk. *ACM Transactions on the Web (TWEB)*, Vol. 9, No. 4, p. 19, 2015.
[17] F. Chiericetti, A. Dasgupta, R. Kumar, S. Lattanzi, and T. Sarlós. On sampling nodes in a network. In *Proceedings of the 25th International Conference on World Wide Web*, pp. 471–481. International World Wide Web Conferences Steering Committee, 2016.

APPENDIX

In this appendix, we describe the basic thinking underlying the counting triangles method as an approximation algorithm for clustering coefficient estimation using experiments in this study. In addition, we propose a counting triangles method through MHRW, which has not been proposed until now. The counting triangles method is a technical method to estimate clustering coefficients by investigating the triangular structures that emerge during the random walk. Until now, we have proposed a method by which the counting triangles method can be applied to SRW and NBRW [8], [15]. A salient aspect of the counting triangles method is that it does not require additional queries, as the queries required for random walk transitions do. As a method of calculating the clustering coefficients as defined, additional queries were required to calculate clustering coefficients [2]. Therefore, it is an important point whether additional queries are required or not.

The basic thinking behind the counting triangles method is to uniformly and randomly select two nodes $v1$, $v2$ from v adjacent nodes in relation to degree 2 or more nodes v visited during the random walk. If an edge exists between $v1, v2$, this is counted as a triangular structure. Here, the probability expectation value that a triangular structure exists is defined with a weighted coefficient, so that the clustering coefficients of node v are equal. In case of SRW, the weighted coefficient is $d(v)/(d(v) - 1)$, and in case of NBRW, the weighted coefficient is 1. Nodes wherein the degree is 1 or below have a clustering coefficient of 0; therefore, there is no need to confirm the triangular structures. When implementing this, $v_1 \in N(v_2)$ or $v_2 \in N(v_1)$ can be confirmed. Thus, it is necessary to obtain the $v1$ or $v2$ adjacent node list. With the counting triangles method using SRW and NBRW, by defining $v1 = \{$node visited one step before $v\}$ and $v2 = \{$node visited 1 step after $v\}$, two nodes can be uniformly and randomly selected from the adjacent node list; this satisfies the state of knowing the adjacent node list of one of the nodes without requiring any additional queries.

Next, the counting triangles method with MHRW is described. Owing to the fact that the MHRW transition destination node is the node selected uniform and randomly from the adjacent node list, this cannot be defined as $v1 = \{$node visited one step before v$\}$and $v2 = \{$ node visited 1 step after v$\}$, as in SRW or NBRW. Therefore, it is necessary to devise a way of selecting $v1, v2$. In this study, in relation to node v of degree 2 or greater visited in MHRW, we define $v1$ as $\{$ transition destination candidate node w within MHRW transition Algorithm 1 $\}$ and $v2$ as the $\{$ node selected in uniform random from list $N(v)/\{v_1\}$ after $v1$ has been removed from the v adjacent node list $\}$. At this time, the probability expectation value $v_2 \in N(v_1)$ is equal to the clustering coefficient of node v. Furthermore, with MHRW, because it is necessary to know the degree of the transition destination candidate nodes to obtain the receive rate to the transition destination candidate nodes within the transition algorithm, the $v1$ adjacent node list is obtained within the transition algorithm. Therefore, an additional query is not required when applying the counting triangles method.