Albatross Sampling: Robust and Effective Hybrid Vertex Sampling for Social Graphs

Long Jin[1], Yang Chen[2], Pan Hui[3], Cong Ding[2],
Tianyi Wang[1], Athanasios V. Vasilakos[4], Beixing Deng[1] and Xing Li[1]
[1] Department of Electronic Engineering, Tsinghua University, Beijing, China
[2] Institute of Computer Science, University of Goettingen, Goettingen, Germany
[3] Deutsche Telekom Laboratories/TU-Berlin, Germany
[4] University of Western Macedonia, Greece
E-mail: jin-l06@mails.tsinghua.edu.cn

ABSTRACT

Nowadays, Online Social Networks (OSNs) become dramatically popular and the study of social graph attracts the interests of a large number of researchers. One critical challenge is the huge size of the social graph, which makes the graph analyzing or even the data crawling incredibly time consuming, and sometimes impossible to do to completion. Thus, graph sampling algorithms have been introduced to obtain a smaller subgraph which reflects the properties of the original huge graph well. Breadth-First Sampling (BFS) is very widely used in graph sampling but it is biased towards high-degree vertices during the process of sampling. Besides, Metropolis-Hasting Random Walk (MHRW), which is proposed to get unbiased samples of the social graph, requires the graph to be well connected. In this paper, we propose a vertex sampling algorithm, so-called Albatross Sampling (AS), which introduces adaptive random jump strategy into MHRW during the sampling process. The embedded random jump makes the sampling procedure more flexible and avoids being trapped in some locally well connected part. According to our evaluation, we find that no matter using tightly or loosely connected graphs, AS performs significantly better than MHRW and BFS. On one hand, AS estimates the degree distribution with much lower Normalized Mean Square Error (NMSE) by consuming the same resource budget. One the other hand, to get an acceptable estimation of the degree distribution, AS requires much fewer resource budget.

Categories and Subject Descriptors

G.3 [**Probability and Statistics**]: Statistical computing

General Terms

Measurement

Keywords

Graph Sampling, Online Social Networks, Random Walk, Metropolis-Hasting Algorithm, Adaptive Jump, Degree Distribution, Convergence Time.

1. INTRODUCTION

In recent years, online social networks (OSNs) become dramatically popular all over the world. For example, Twitter, which provides microblogging service, has more than 200 million users around the world and generates 110 million "tweets" per day by January, 2011 [6]. The spreading of OSNs also attracts a large number of researchers to explore and study the newly-built large scale networks and their research topics are diverse and fascinating, such as social interaction [20], information propagation [9] and user behavior characterization [3] in social networks.

Social networks are usually modeled as social graphs for analysis. One of the challenges what researchers face is the huge size of OSNs. Firstly, it is almost impractical to deal with the complete datasets, since crawling large social graphs is incredibly time consuming, and sometimes is impossible to be completed. Moreover, algorithms for analyzing huge social graphs require a large quantity of time to compute, even running on the platform of high-performance computer clusters. Secondly, complete datasets from OSNs are usually not public accessible, due to privacy settings of users and the protecting from OSN companys. Finally, the number of users in OSNs grows rapidly and connections between users vary with time, thus dynamic large graphs cannot be crawled totally. Therefore, attention has been paid on how to shrink a huge social graph to a representative sample, which should have a relatively small size and maintain properties of the original social graph.

Several graph sampling methods have been proposed to obtain samples of OSNs and these graph sampling methods can be classified as edge sampling method and vertex sampling method [10]. Random Vertex Sampling and Random Edge Sampling are two basic and intuitive sampling methods. As their names imply, Random Vertex (Edge) Sampling method samples vertices (edges) randomly in the whole graph. However, these two sampling methods are either resource intensive or impractical to sample OSNs [17].

One practical graph sampling method is Breadth-First Sampling (BFS), which is used for social network analysis in [1, 9, 14, 20]. However, BFS is biased towards high-degree vertices [8]. Another widely used sampling algorithm is Random Walk (RW) and lots of practical graph sampling methods are based on RW.

On the one hand, RW naturally samples edge uniformly in non-bipartite undirected graphs, but not in disconnected graphs [12]. Frontier Sampling (FS) [17], which is an edge sampling method using multidimensional random walkers, is proposed to exhibit lower estimation errors than RW in the presence of disconnected or loosely connected subgraphs. Since FS is an edge sampling method, it is more convenient to estimate edge-centric properties and specific estimators of degree distribution and global clustering coefficient are proposed in [17].

On the other hand, RW method is biased towards high-degree vertices [8] and Metropolis-Hasting Random Walk (MHRW) method is proposed to obtain unbiased samples of social graphs. MHRW is a vertex sampling methods, whose goal is to mimic random vertex sampling by random walking in graphs. However, one design assumptions of MHRW is that the social graph is well connected [7], which results in MHRW not proper for sampling disconnected or loosely connected graphs. What is more, Random Jump (RJ), which may jump to any random vertex with a fixed probability in each step, is proposed in [10] to get rid of the walker being stuck in some locally well connected part of the graph.

In this paper, we focus on proposing an improved vertex sampling method which performs well in sampling social graphs, either tightly or loosely connected. By introducing adaptive jump strategy into MHRW, we propose a hybrid sampling algorithm named Albatross Sampling (AS). According to our evaluation, given the same sampling cost, AS is more robust for estimating degree distribution with lower Normalized Mean Square Error (NMSE) and also more effective for converging more quickly with much smaller convergence time. Therefore, AS is a promising robust and effective hybrid vertex sampling algorithm for social network analysis.

This paper is organized as follows. In Section 2, basic properties about social graphs and measurement of information retrieval cost during the sampling process are introduced. Then, in Section 3, existing graph sampling algorithms mentioned above are discussed in detail and our improved algorithm Albatross Sampling is proposed in Section 4. Finally, the performance of these algorithms for sampling social graphs is evaluated in Section 5 and conclusion is made in Section 6.

2. BACKGROUND

2.1 Properties of Social Graphs

OSNs are usually modeled as social graphs, which is represented as $G = (V, E)$. Here, a vertex v in set V represents a user in the OSN and an edge e in set E represents a friendship link or a "following" relationship between users, which can be either undirected or directed.

For directed graphs, we define $k_{iv}(k_{ov})$ as the in-degree (out-degree) of vertex v and k_v as the degree of vertex v when the directed graph is changed into undirected graph. Then, we define $\theta_{ik}(\theta_{ok})$ as the fraction of vertices with in-degree (out-degree) less than or equal to k and $\widehat{\theta_{ik}}(\widehat{\theta_{ok}})$ as the estimated value through sampling.

In the real world, some large scale social network graphs are not fully connected and may contain disconnected or loosely connected components, e.g. wireless social networks [5]. The simple random walker may be trapped in locally well connected part of the graph, and if the properties of that part differ significantly from those of the whole graph, the sample set cannot represent the original graph well. Therefore, robust and effective methods for sampling disconnected or loosely connected graphs should be studied.

2.2 Measurement of Sampling Cost

In this part, we propose several definitions related to the measurement of information retrieval cost during the sampling process.

Firstly, Total-Cost is defined as the total resource budget (time, bandwidth, or cache) that is used in the process of sampling. Since downloading the profile of a user is much more time-consuming compared to making the choice of the next sampled user, Total-Cost can be viewed as the maximum number of unique users that are visited during sampling.

Secondly, in most social networks, all incoming and outgoing neighbors of a sampled vertex can be learned. For instance, when we visit a user in Twitter, all the followers and followees in the user profile can be known. This fact makes graph sampling methods, such as BFS and RW, feasible and cheap. We define the resource for visiting a new neighbor of the current vertex as Walk-Cost, which can be normalized as 1. We should mention that information of sampled vertices are stored in cache, therefore only visiting new vertices generates sampling cost.

Finally, in different sampling methods, various sampling strategies may be trigged in each step and more resource may be used. For example, in MySpace, each user is given a unique user-ID, thus Random Vertex Sampling can be implemented by selecting a random number from the space of user-IDs in each step. If the selected number is a valid user-ID, the user is sampled, otherwise the number is discarded. However, for most OSNs, valid user-IDs are sparse in the space of user-IDs. In MySpace, sampling a valid user requires nearly 10 attempts on average [16]. Similarly, in Random Jump, we may jump to any vertex to restart the random walker with a fixed probability in each step. To compare different sampling algorithms fairly, we define the resource for choosing a new random vertex in the total graph as Jump-Cost and we choose Jump-Cost as 10 in our evaluation.

3. ANALYSIS OF EXISTING SAMPLING ALGORITHMS

The sampling process of social graphs usually starts from either one or several initial vertices, which can be called seeds. At the beginning, we have Total-Cost resource budget for sampling. After we visit a vertex, its neighbors are all discovered and sampling strategy is used to decide which vertex is sampled next. Then this process is iterated until Total-Cost is used up in the sampling process. Different sampling methods differ in the size of seeds and sampling strategy. In this section, we describe several popular sampling methods in detail.

Breadth-First Sampling (BFS): BFS is a classical graph sampling algorithm, which has been widely applied in studying OSNs. For example, BFS is used for measurement and topological characteristics analysis [1, 14] and user behavior analysis of OSNs [9, 20].

BFS starts from one random seed in the graph and aims at collecting the vertices which are close to the seed. T-wo queues are kept in the sampling process of BFS: queue *Sampled* stores sampled vertices, while queue *Waiting* s-tores vertices that will be sampled. Initially, the seed is put into queue *Waiting*. At each step, the first vertex v in queue *Waiting* is moved to queue *Sampled*, and all the neighbors of vertex v are added into queue *Waiting*, unless the neighbor is already in queue *Waiting* or queue *Sampled*. The process loops until the Total-Cost is used up. During the sampling process, once queue *Waiting* is empty, a random vertex will be selected and inserted into queue *Waiting*.

Recently, [8] points out that BFS is bias towards high-degree vertices. Moreover, due to our evaluation, BFS may only collect some local part of the graph and performs badly in disconnected or loosely connected graphs.

Metropolis-Hasting Random Walk (MHRW): Random Walk is another widely used graph sampling method. In RW, one random vertex is the initial seed and the sampling strategy is that we always choose one of the neighbors (vertex v) of the current sampled vertex (vertex u) as the next vertex, which spends Walk-Cost. Thus, the transition probability from u to v is

$$P_{u,v} = \begin{cases} 1/k_u & \text{if } v \text{ is } u\text{'s neighbor} \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

RW is simple and practical but it is biased towards high-degree vertices. In RW, the next sampled vertex just depends on the current sampled vertex, thus the sampling process can be modeled as a Markov Chain. If the graph is undirected and non-bipartite, the Markov Chain related to RW is ergodic [12]. Therefore, the stationary probability of each edge in undirected graph is equal to $1/|E|$ and each vertex is sampled with the probability $\frac{k_u}{2|E|}$. We can see that, vertices with high-degrees are more likely to be sampled in RW.

To get rid of the bias towards high-degree vertices, the transition probabilities in RW should be modified appropriately and Metropolis-Hastings Random Walk is proposed in [8] to sample vertices uniformly. The Metropolis-Hastings algorithm is a Markov Chain Monte Carlo method to sample from a probability distribution whose direct sampling is difficult [13].

The algorithm of MHRW is as follows. Firstly, choose a vertex u as the initial seed. Secondly, select one neighbor (vertex v) of vertex u and generate a random number p between 0 and 1 uniformly. If $p < k_u/k_v$, v is chosen as the next sampled vertex, otherwise the walker still stays as vertex u. Then, the second step is iterated until Total-Cost is exhausted. The transition probability from u to v is

$$P_{u,v} = \begin{cases} min(\frac{1}{k_u}, \frac{1}{k_v}) & \text{if } v \text{ is } u\text{'s neighbor} \\ 1 - \sum\limits_{w \neq u} min(\frac{1}{k_u}, \frac{1}{k_w}) & \text{if } v = u \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

From the above equation, we can notice that the possibility to visit vertices with high degree is reduced and MHRW is proved to converge to uniform sampling, which means that each vertex in the whole graph is sampled with the probability $1/|V|$ [8].

We should mention that one design assumption of MHRW is that the social graph is well connected. Besides, all duplicated vertices are valid and important in MHRW, since the duplication makes MHRW converge to uniform sampling, inherently.

Random Jump (RJ): Random Jump is a sampling method which is similar to RW. The difference between them is that the walker in RJ can jump to any random vertex in the graph with a fixed probability in each step. The benefit of jumping randomly is getting rid of the random walker being stuck in locally well connected part of the graph. When the jump strategy is trigged, it will spend Jump-Cost. We should mention that RJ is still biased towards high-degree vertices, since if the jump strategy is not trigged, the vertices are selected just like RW.

Jump-Cost may be large in some OSNs. For instance, in Myspace, valid user-IDs are sparse in the space of user-IDs and Jump-Cost is 10. Then, the size of sampled vertices obtained by RJ would be much smaller than the sample size obtained by RW. In practice, we usually prefer to getting a large and fair sample, given the same Total-Cost. However, RJ may not perform well in respect of the size of sampled vertices.

Frontier Sampling (FS): All the sampling methods mentioned above are vertex sampling methods and Frontier Sampling [17], which performs multiple dependent random walkers in sampling graphs, is an edge sampling algorithm. Moreover, vertex sampling performs better than edge sampling in estimating small degree distribution [17], and in most OS-Ns vertices with small degree make up a major portion of the total graph. Thus we are focusing on improving the performance of vertex sampling methods in this paper. N-evertheless, we still introduce FS, which is a representative edge sampling method.

FS works in the following way. Firstly, multiple vertices are selected as the initial seeds. Secondly, choose a vertex u from the set of seeds with the probability proportional to its degree, i.e. $P(u) \propto k_u$. Thirdly, choose an edge (u, v) that starts from vertex u as a sample edge, and then replace u with v in the set of seeds. Repeat step 2 and step 3 until Total-Cost is exhausted.

FS is practical and samples edge uniformly, which indicates that the sample set itself is still biased towards high-degree vertices. Unlike MHRW aims at obtaining an unbiased sample directly, FS corrects the bias by specific estimators. In [17], estimators related to degree distribution and global clustering coefficient, which are based on sampled edges, are proposed. The results show that FS can achieve lower NMSE than RW for estimating degree distribution, especially in disconnected or loosely connected graphs.

RWRW is proposed in [8], and if it is treated as an edge sampling method, RWRW uses exactly the same estimator of FS. The comparisons between MHRW and RWRW [8] are also suitable for comparing MHRW and FS. Firstly, MHRW has the “ready to us” merit, since vertices are sampled uniformly. However, FS is biased towards high-degree vertices and requires re-weighting appropriately. Secondly, specific estimators should be built for estimating different graph properties. However, only estimators of degree distribution and global clustering coefficient are currently available and estimators for purely data-analytic procedures, such as hierarchical clustering or multidimensional scaling, is impossible to be constructed [8]. Thus, MHRW is more simple and versatile than FS in practice.

From the above description, we can conclude that BFS, RW and RJ do not converge to uniform sampling and are biased towards high-degree vertices. MHRW samples vertices uniformly, however, one design assumption of MHRW is that the social graph is well connected. FS, which is an edge sampling method, performs stable in the disconnected or loosely connected graphs, but specific estimators should be built for estimating graph properties. Besides degree distribution and global clustering coefficient, estimators of any other properties are currently unavailable [17].

Algorithm 1 Albatross Sampling

```
cost ← 0
sample set V ← empty
select a random vertex v as the initial seed
while cost < Total-Cost do
  generate α from uniform distribution U[0, 1]
  if α < 1 − (1 − p)^n then
    generate β from uniform distribution U[0, 1]
    if β < q then
      choose a new random vertex u
      add u to the sample set V
    else
      choose a vertex u from the sample set V
      add u to the sample set V
    end if
    if u is visited for the first time then
      cost ← cost+Jump-Cost
    end if
    v ← u
  else
    select an edge (v, w) starting from v randomly
    generate γ from uniform distribution U[0, 1]
    if γ < k_v/k_w then
      add w to the sample set V
      if w is visited for the first time then
        cost ← cost+Walk-Cost
      end if
    else
      remain at v
      add v to the sample set V
    end if
  end if
end while
```

4. ALBATROSS SAMPLING

4.1 Workflow of Albratross Sampling

In this paper, we introduce adaptive jump strategy into MHRW and propose an improved vertex sampling method named Albatross Sampling. The benefit of jump strategy in AS is to get rid of being stuck in some locally well connected part of the whole graph and gathers a "comprehensive" sample of the original graph.

To design our jump strategy, we have the following consideration: (1) As certain vertex is sampled for more times, the probability of jumping away from this vertex should be larger. We should mention that AS may stay at the same vertex in successive steps, which does not increase the sampled times of that vertex. (2) Jump-Cost may be large in some OSNs, resulting in jumping to a new random vertex to explore the graph may be a waste, thus jumping back to sampled vertices can be a cheap and effective way to get rid of being trapped in local part sometimes.

Therefore, our adaptive jump strategy in AS works in the following way. The probability of jumping from the current vertex v is $1-(1-p)^n$, where n is the times that v has been sampled and p is used to control the jump probability. Once the jump strategy is trigged, we can jump to a new random vertex with the probability of q, or a sampled vertex with the probability of $1-q$. The algorithm of AS is described in Algorithm 1.

4.2 Unbiasness of Albatross Sampling

In AS, the transition probability from vertex u to vertex v is

$$P_{u,v} = \begin{cases} min(\frac{\gamma_0}{k_u}, \frac{\gamma_0}{k_v}) + \frac{\alpha_0}{|V|} + \frac{\beta_0 I(v)}{|S|} & \text{if } v \text{ is } u\text{'s neighbor} \\ \gamma_0 - \sum\limits_{w \neq u} min(\frac{\gamma_0}{k_u}, \frac{\gamma_0}{k_w}) + \frac{\alpha_0}{|V|} + \frac{\beta_0 I(v)}{|S|} & \text{if } v = u \\ \frac{\alpha_0}{|V|} + \frac{\beta_0 I(v)}{|S|} & \text{otherwise} \end{cases}$$

where,

$$\alpha_0 = (1-(1-p)^n)q$$

$$\beta_0 = (1-(1-p)^n)(1-q)$$

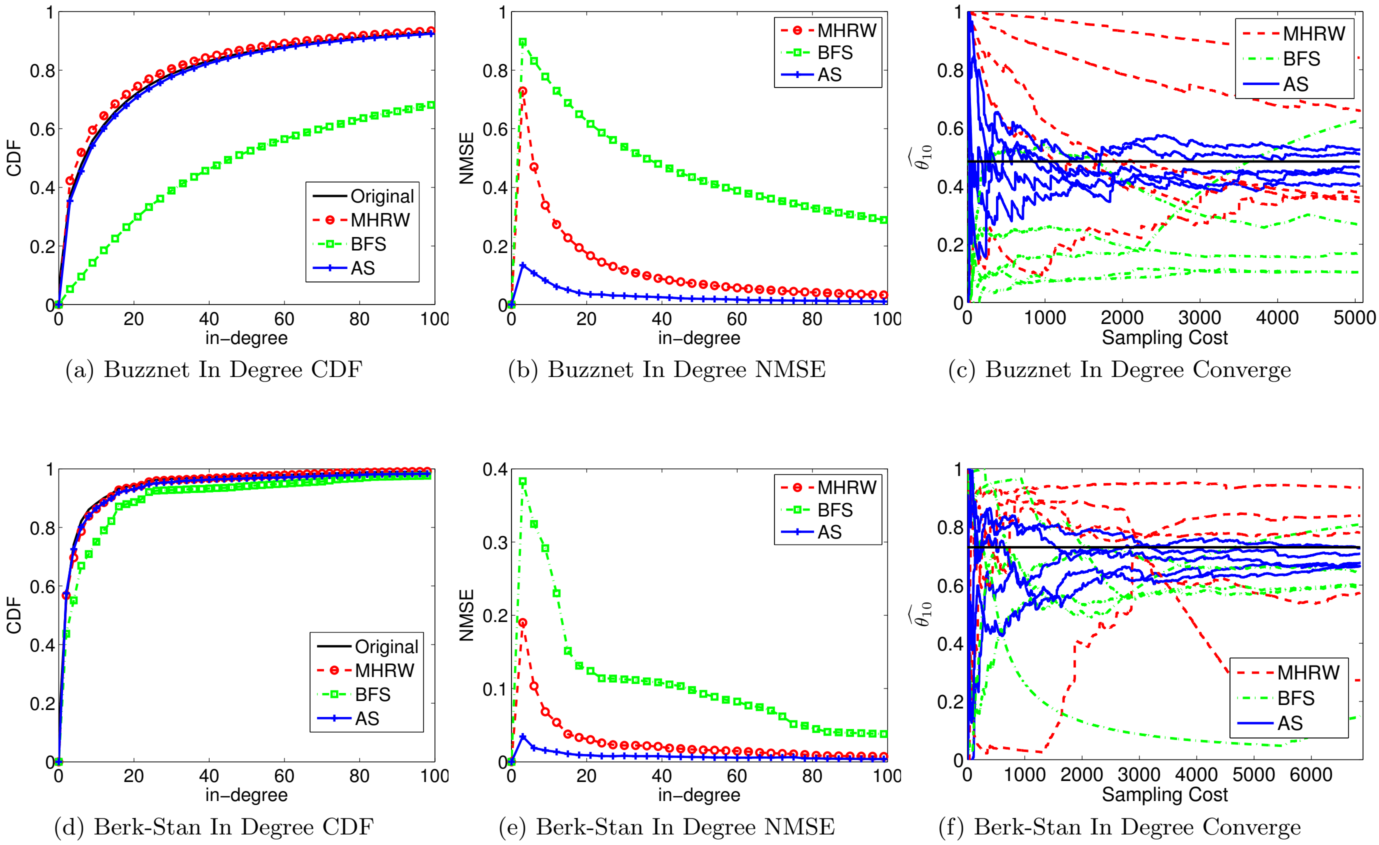
Here, p and q are parameters related to jump strategy and n is the sampled times of vertex u. α_0 represents the probability of jumping to a new vertex in this step and β_0 represents the probability of jumping to a sampled vertex in this step. $\gamma_0 = 1 - \alpha_0 - \beta_0$ represents the probability of walking on in the graph. $|V|$ and $|S|$ represent the number of vertices in the original graph and the sampled set, respectively. $I(v)$ is an indicator for whether v is in the sampled set V.

In AS, the inherent Metropolis-Hasting algorithm guarantees that each vertex is sampled uniformly when jump strategy is not trigged. Since only inconsecutive sampling vertex u increase n (the sampled times of u), n is independent with k_u. From the above equations, we can see that both α_0 and β_0 are only dependent with n, thus they are also independent with k_u. Therefore, each vertex is sampled with the same probability, which indicates that AS is an unbiased sampling methods.

Benefit from adaptive jump strategy, AS avoids being trapped in locally well connected part of the social graph and converge to uniform sampling quickly. Due to our evaluation, AS estimates degree distribution with lower NMSE by consuming the same resource budget and converge quickly with smaller convergence time, even sampling disconnected or loosely connected graphs. Therefore, Albatross Sampling is a promising robust and effective vertex sampling method for social network analysis.

5. EVALUATION

In this section, we evaluate the performance of the sampling algorithms mentioned above. The comparison is focused on the performance of BFS, MHRW and AS. BFS is not an unbiased sampling method, nevertheless, we still compare with BFS, for its widely use in OSNs analysis [1, 9, 14, 20]. Here, we use two criterions to evaluate different sampling algorithms.



(a) Buzznet In Degree CDF (b) Buzznet In Degree NMSE (c) Buzznet In Degree Converge

(d) Berk-Stan In Degree CDF (e) Berk-Stan In Degree NMSE (f) Berk-Stan In Degree Converge

Figure 1: CDF, NMSE and Converge Curves for Buzznet(Cost=|V|/20) and Berk-Stan(Cost=|V|/100)

(1) **The accuracy of estimating degree distribution:** Normalized Mean Square Error (NMSE) is used for evaluating the estimating accuracy of these methods, which is also used in [17]. NMSE for degree k is defined as

$$NMSE(k) = \frac{\sqrt{E[(\widehat{\theta_k} - \theta_k)^2]}}{\theta_k} \tag{3}$$

In this paper, we use NMSE as a criterion to evaluate the robustness of sampling methods. If a sampling algorithm achieves a lower NMSE in estimating degree distribution given the same sampling budget, it is regarded as more robust.

(2) **The converge rate of algorithms:** Firstly, we choose the evolution of estimating θ_{10} for comparison, without loss of generality. Here, θ_{10} means the fraction of vertices with in-degree(out-degree) less than or equal to 10, which is also used in [17]. Moreover, we give out the convergence time of different sampling methods. Similar to the mixing time of a Markov chain [11], we define *convergence time* as the cost c_0 such that

$$|\widehat{\theta_k}(c) - \theta_k| \leq 1/4 \tag{4}$$

for $0 \leq k \leq max(in-degree, out-degree), c_0 \leq c$

Convergence time reflects the fewest budget required for getting an acceptable estimation of the degree distribution.

5.1 Data Set

The data sets used for evaluation are Buzznet [2] and Berk-Stan [18], and their basic information is shown in Table 1. Buzznet is a photo, journal, and video-sharing social media network where vertices represent users in the graph. In Buzznet, user A can subscribe to the updates of user B, which indicates a directed edge from A to B. Berk-Stan is the web graph of Berkeley and Stanford collected in 2002. Vertices represent pages from berkely.edu and stanford.edu domains and hyperlinks between them are represented as directed edges.

Table 1: Basic Information of Data Set

	Type	Nodes	Edges	SCC
Buzznet	Directed	101,169	4,284,534	0.944
Berk-Stan	Directed	685,230	7,600,595	0.489

In table 1, Strongly Connected Components (SCC) represents the fraction of number of vertices in the largest strongly connected component [4]. SCC shows the connectivity of a graph: if the value of SCC is smaller, the graph is more loosely connected. Thus, Buzznet is a tightly connected graph and Berk-Stan is a loosely connected graph. And we use both tightly and loosely connected graphs to compare these algorithms.

5.2 The Accuracy of Estimate

To make the analysis of degree distribution impressive, we plot the Cumulative Distribution Function (CDF) and Normalized Mean Square Error (NMSE) of degree distribution of the sampled vertices. In Figure 1 (a) and (b), the in-degree CDF and NMSE of Buzznet are presented. We choose Total-Cost for sampling as 5% of the total number of vertices of Buzznet. In Figure 1 (d) and (e), we present the in-degree CDF and NMSE of Berk-Stan. We choose Total-Cost for sampling as 1% of the total number of vertices of Berk-Stan. The parameters p and q are chosen as 0.02 and 0.8, respectively.

From these figures, we can see that BFS is biased towards high-degree vertices significantly, whose NMSE is much larger than that of MHRW and AS. Moreover, the CDF curves of MHRW and AS are both almost identical to the original CDF, but AS achieves smaller NMSE than MHRW. From the above description, we can conclude that AS is more robust for estimating degree distribution in both tightly and loosely connected graphs.

5.3 The Converge Rate of Algorithms

To evaluate comprehensively, we also compare the converge rate of these sampling methods. Figure 1 (c) and (f) show five sample paths of the evolution of $\widehat{\theta_{10}}$ as a function of sampling cost over Buzznet and Berk-Stan. And the black lines show the true value of θ_{10}. We can see that four out of five sample paths in BFS underestimate θ_{10}, which indicates that BFS is biased towards high-degree vertices. MHRW performs badly in these two graphs and only one path converge to the true value. However, all five paths of AS converge quickly and stably to the true value.

Moreover, in Table 2, we present the convergence time of different sampling methods in Buzznet and Berk-Stan. The convergence time of AS is 9.2% (Buzznet data set) and 6.1% (Berk-Stan data set) of the convergence time of BFS and 12.0% (Buzznet data set) and 8.4% (Berk-Stan data set) of the convergence time of MHRW. From these simulation results, we find that AS is more effective and reliable for converging quickly and stably.

Table 2: Convergence Time

	Buzznet	Berk-Stan
BFS	4371.2	6355.2
MHRW	3361.4	4652.5
AS	402.9	390.7

6. CONCLUSIONS AND FUTURE WORK

In this paper, we propose an improved vertex sampling sampling algorithm named Albatross Sampling, which introduces adaptive jump strategy into MHRW during the sampling process. Due to our evaluation, AS is more robust for estimating degree distribution with lower NMSE and more effecitve for converging more quickly with much smaller convergence time than MHRW and BFS, given the same sampling cost. Moreover, AS is reliable for sampling both tightly and loosely connected social graphs.

In the future, we will focus on how to select optimal parameters, including p and q, for AS. And the proper value for sampling social graphs should be based on experiments in more large scale social graphs. We will also evaluate these sampling algorithms for estimating other important graph properties, such as Assortativity [15] and Clustering Coefficient [19].

7. REFERENCES

[1] Y.-Y. Ahn, S. Han, H. Kwak, S. Moon, and H. Jeong. Analysis of topological characteristics of huge online social networking services. *WWW*, 2007.

[2] ASU. Social computing data repository at arizona state university. *http://socialcomputing.asu.edu/pages/home.*

[3] F. Benevenuto, T. Rodrigues, M. Cha, and V. Almeida. Characterizing user behavior in online social networks. *ACM SIGCOMM Internet Measurement Conference*, 2009.

[4] R. Diestel. *Graph Theory.* Springer-Verlag, 2010.

[5] N. Eagle, A. S. Pentland, and D. Lazer. Inferring friendship network structure by using mobile phone data. *PNAS*, 106(36):15274–15278, August 2009.

[6] Geek.com. Twitter reaches 200 million users and 110 million tweets per day. *http://www.geek.com/articles/news/twitter-reaches-200-million-users-and-110-million-tweets-per-day-20110120/.*

[7] M. Gjoka. Measurement of online social networks. *UC Irvine PhD Thesis*, 2010.

[8] M. Gjoka, M. Kurant, C. T. Butts, and A. Markopoulou. Walking in facebook: A case study of unbiased sampling of osns. *IEEE Infocom*, 2010.

[9] H. Kwak, C. Lee, H. Park, and S. Moon. What is twitter, a social network or a news media? *WWW*, 2010.

[10] J. Leskovec and C. Faloutsos. Sampling from large graphs. *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2006.

[11] D. A. Levin, Y. Peres, and E. L. Wilmer. *Markov Chains and Mixing Times.* American Mathematical Society, 2008.

[12] L. Lovasz. Random walks on graphs: a survey. *Combinatorics*, 1993.

[13] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller. Equation of state calculation by fast computing machines. *J. Chem. Physics*, 106(21), August 1953.

[14] A. Mislove, M. Marcon, K. P. Gummadi, P. Druschel, and B. Bhattacharjee. Measurement and analysis of online social networks. *ACM SIGCOMM Internet Measurement Conference*, 106(21), October 2007.

[15] M. Newman. Assortative mixing in networks. *Phys. Rev. Lett.*, 89, 2002.

[16] B. Ribeiro, W. Gauvin, B. Liu, and D. Towsley. On myspace account spans and double pareto-like distribution of friends. *UMass Amherst Technical Report*, (UM-CS-2010-001), January 2010.

[17] B. Ribeiro and D. Towsley. Estimating and sampling graphs with multidimensional random walks. *ACM SIGCOMM Internet Measurement Conference*, November 2010.

[18] Stanford. Stanford large network dataset collection. *http://snap.stanford.edu/data/index.html/.*

[19] D. J. Watts and S. Strogatz. Collective dynamics of 'small-world' networks. *Nature*, 393(6684), June 1998.

[20] C. Wilson, B. Boe, A. Sala, K. P. N. Puttaswamy, and B. Y. Zhao. User interactions in social networks and their implications. *ACM EuroSys*, 2009.