Monotone Sampling of Networks*

Tim Grube[1], Benjamin Schiller[1], and Thorsten Strufe[2]

[1] Technische Universität Darmstadt, Hochschulstraße 10, 64289 Darmstadt, Germany
lastname@cs.tu-darmstadt.de
[2] Technische Universität Dresden, Nöthnitzer Straße 46, 01187 Dresden, Germany
firstname.lastname@tu-dresden.de

Abstract. Determining the graph-theoretic properties of large real-world networks like social, computer, and biological networks, is a challenging task. Many of those networks are too large to be processed efficiently and some are not even available in their entirety. In order to reduce the size of available data or collect a sample of an existing network, several sampling algorithms were developed. They aim to produce samples whose properties are close to the original network. It is unclear what sample size is sufficient to obtain a sample whose properties can be used to estimate those of the original network. This estimation requires sampling algorithms that produce results that converge smoothly to the original ones since estimations based on unsteady data are unreliable. Consequently, we evaluate the monotonicity of sampled properties while increasing the sample size. We provide a ranking of common sampling algorithms based on their monotonicity of relevant network properties using the results from four nework classes.

1 Introduction

Today's networks are quite large, in many cases too large to understand the network or to compute its properties. We have to reduce the complexity and therefore the size of networks to use the network for analyses and research. We can reduce the size by using graph coarsening or sampling techniques. Graph coarsening and some sampling techniques require the availability of the complete network. This constraint is rarely satisfied. Sampling by exploration allows to gain knowledge about the unavailable network, but it will usually distort properties as the sampling process can be biased. There are two large classes of sampling algorithms, we can sample using a breadth first sampling (BFS) approach, constructing the sample from the local area first, or we can use a random walk (RW) approach, traversing along random paths of nodes and constructing the sample with nodes from deeper areas of the network. The convergence behavior of network properties like the degree distribution depend highly on the underlying network and the used sampling algorithm.

* This research was partly funded by the MSIP (Ministry of Science, ICT & Future Planning), Korea in the ICT R&D Program 2014 and the Software Campus by the German Federal Ministry of Education and Research (BMBF) under grant no. "01IS12054"

Many work has been done to overcome these biases and many specialized algorithms were developed. These algorithms produce samples, whose properties converge faster to the original networks properties, but as the properties of the original network are typically unknown, it is undecidable whether the quality demands or original properties are met or not.

Our approach is another way of solving this problem. We are proposing a new metric, which allows to develop an estimator for network properties in future work. This estimator should deliver the properties of the original network if it gets the network properties of the sample, the specification of the sampling algorithm, and the assumed size of the original network. For the development of such an estimator, we need the sampling algorithm to produce a sample with monotone converging network properties. In this paper, we are investigating the convergence monotonicity of the network properties on sampled networks.

The rest of the paper is structured as follows: Section 2 presents the related work about sampling and the evaluation of newly developed sampling algorithms. We present in Section 3 the desired behavior of sampling algorithms. We present the results and the discussion of our work in Section 4 and conclude with a summary and outlook in Section 5.

2 Related Work

The related work lists two typical classes of sampling techniques. The first one is the deletion of nodes or edges. Node deletion techniques use the complete network as basis and are deleting nodes until the network size is reduced to the desired size. Edge deletion uses a similar technique, instead of removing nodes, these algorithms remove edges and under algorithm specific circumstances the attached nodes. The complete network has to be available to apply these two techniques. The second technique is sampling by exploration, using this technique, the sampling algorithm traverses from a start node into the network and collect the nodes to the sample. This technique is interesting for at least two reasons: First, it is easy to use by instrumenting crawlers. Second, we do not have this dependency on the availability of the complete network. Sampling by exploring is the common technique to reduce the complexity of networks and gain an excerpt of the whole network.

A lot of research has been done into the direction of developing more sophisticated sampling algorithms in this area. These algorithms are developed to produce samples, whose property values converge faster towards the property values of the

original network. The sampling algorithms can be classified into breadth first approaches and random walk approaches. Table 1 shows the classification, based on the type of walking, and the abbreviations of the analyzed algorithms.

Breadth First Sampling

The BFS algorithms traverse through the network by focusing on the local neighborhood first. The simplest implementation is a classical BFS which visits all the neighbors of node. Krishnamurthy, Leskovec and Stutzbach [4,7,13] use the BFS in their work. Goodman et al. [2] introduced snowball sampling (SS), which is a variation of the BFS and visits only a specifiable number of new, still unseen neighbors. Lee et al. [6] have evaluated this variation. Similar to SS, respondent-driven sampling (RDS) is developed by Heckarthorn et al. [3] and analyzed by Rasti and Kurant [10,5]. RDS visits a specifiable number of random neighbors, ignoring if these neighbors are already known or not. Forest fire (FF), introduced by Leskovec et al. [7], is the last BFS derivate. It collects all neighbors with a certain probability into a walking queue.

Random Walk Sampling

The second class is the one of the random walk algorithms. The simplest one is the random walk sampling (RW). This algorithm traverses through the network by exploring along random neighbors. This sampling algorithm is well studied, e.g. by Stutzbach, Leskovec and Ribeiro [13,7,11]. Intuitively, and mathematically proveable, the RW sampling is biased towards nodes with higher degrees in the network. To overcome this bias, and to collect a representative sample of the network, two methods of correcting the probability for the next node were introduced. Stutzbach et al. [13] called their version random walk with degree correction (RW-DC). Rasti et al. [10] proposed a slightly different correction and named it metropolized hastings random walk (RW-MH). Both approaches depend on the degree of the potential next node on their exploration way.

Stutzbach et al. [13] showed two further variations of the RW: The first one is the random stroll (RS). This variation is moving like the RW but skips intermediate nodes from adding them to the sample. The second algorithm is the combination of RS and RW-DC. Random stroll with degree correction (RS-DC) skips intermediate nodes like the RS and moves with a degree correction like the RW-DC.

Leskovec et al. [7] introduced another variation of the RW. In particular, they introduced a jump probability in the random jump (RJ). This probability allows to move to a farther area of the network to avoid getting caught in a small area

of the network.

Ribeiro et al. [11] developed a variation of a random walk with multiple instances. Since a simple parallel execution would suffer from the same problems like the classical RW, they introduced a dependency between the instances. Only one instance, in the default setting the one with the highest node degree on the current position, is allowed to move through the network. The active instance is picked every round again. This sampling algorithm is called frontier sampling (FS).

Another algorithm, similar to the RW, is the depth first sampling (DFS). This algorithm moves to the first neighbor and collects the remaining neighbors in a queue. This queue affects the behavior if the algorithm is getting caught. Even though there are similarities between RW and DFS, the results are quite different due to the impact of the queue in DFS. This algorithm is often used as a kind of baseline, e.g. by Krishnamurthy and Leskovec [4,7].

Table 1. Analyzed sampling algorithms.

Class	Algorithm	Abb.	Related Work
Breadth First Sampling	Breadth First Sampling	BFS	[4,7,13]
	Forest Fire	FF	[7]
	Respondent-driven Sampling	RDS	[3,10,5]
	Snowball Sampling	SS	[2]
Random Walk Sampling	Depth First Sampling	DFS	[4,7]
	Frontier Sampling	FS	[11]
	Random Jump	RJ	[7]
	Metropolized Hastings Random Walk	RW-MH	[10]
	Random Stroll	RS	[13]
	Random Stroll with Degree Correction	RS-DC	[13]
	Random Walk	RW	[13,7,11]
	Random Walk with Degree Correction	RW-RDC	[13]

Network Properties

We analyze the common network properties. The earlier introduced sampling algorithms are partially evaluated with these metrics in their corresponding papers. There are two types of properties: The first one is a single scalar value, for example a floating-point number or an integer. The second type is a distribution, which provides for each possible value a certain probability to find this value in the network. We concentrate our work in this paper on the single scalar values.

The assortativity coefficient (AC) is a measure for the correlation of connected nodes. We use the definition from Newman et al. [9,8]. We analyze the clustering coefficient (CC) in both characteristics, the transitivity which is for example used by Chakrabarti et al. [1] and the average clustering coefficient which is for example used by Krishnamurthy et al. [4]. The degree distribution (DD) allows the derivation of multiple single scalar values. We inspect the average node degree, the average in-degree, the average out-degree and the maximum degree of the networks. We are deriving multiple single scalar values from the shortest path length distribution (SP), too. The characteristic path length is a measure for the expected path length in the network, the diameter is the maximum shortest path length in the network. As the diameter is prone to distortions, we are analyzing the effective diameter of the 90% quantile. This metric computes the maximum shortest path length for the part of 90% connected nodes in the network. Our implementation uses the definition from Chakrabarti et al. [1]. By ignoring the end of the heavy tail, we remove the sensitivity for deformed networks. Table 2 lists the used metrics, submetrics and abbreviations for the metrics.

Table 2. Evaluated network properties, analyzed submetrics.

Metric	**Submetrics**	**Abb.**
Assortativity	Assortativity Coefficient	AC
Clustering	Average Clustering Coefficient Transitivity	CC
Degree	Average Degree (avg) Average In-Degree (avg_{in}) Average Out-Degree (avg_{out}) Maximum Degree (max)	DD
Shortest Path	Characteristic Path Length (cpl) Diameter (diam) Effective Diameter, 90% (effectiveDiam)	SP

3 Monotonicity

We propose a new approach to circumvent the nescience of the original networks properties. Instead of evaluating the sampling algorithms with respect to the speed of convergence, we evaluate the monotonicity of the property convergence along increasing sample sizes. The monotonicity is an important property to support the development of estimators to project the properties of the sampled network to the original network. We rank the well known and commonly used explorative

sampling algorithms with respect to the monotonicity properties of their samples.

The looked-for algorithm produces smoothly approximating properties among increasing sampling sizes. We count the direction changes of the property convergence while sampling with increasing sample sizes. Figure 1 shows the aggregrated AC of a webgraph, sampled 100 times with DFS (in blue) and RW (in red). The original networks assortativity coefficient is plotted as a green line. The AC sampled with RW is obviously slower converging than the AC sampled with DFS, even though, the smooth progression allows develop a simpler estimator for the original property values. We are computing the monotonicity by comparing the property

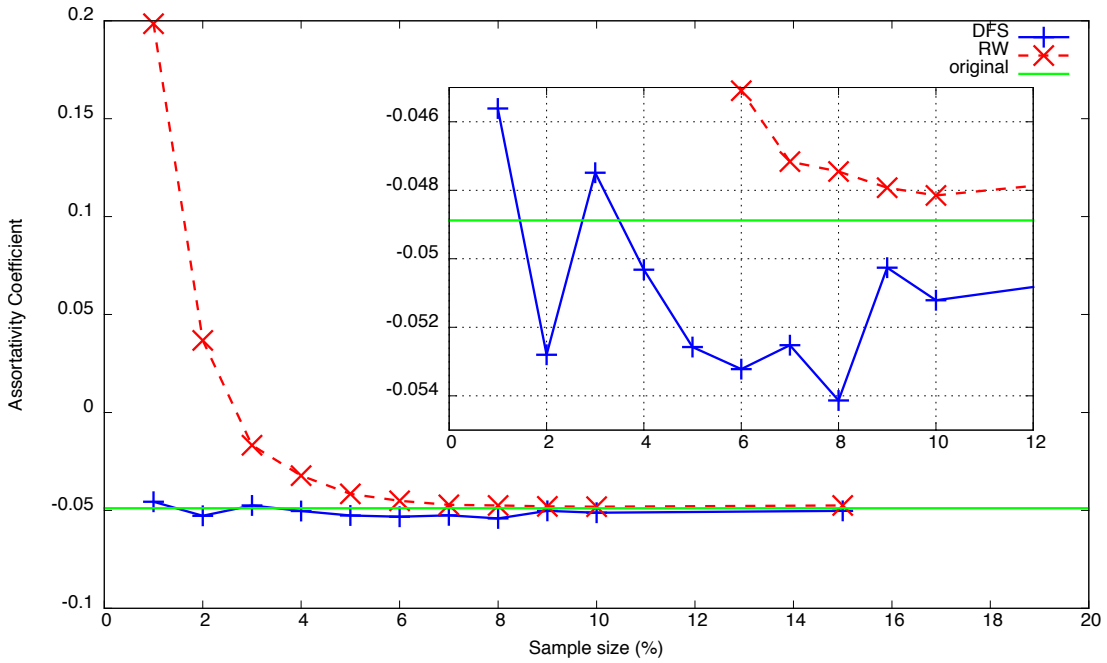


Fig. 1. Monotonicity of the AC on a webgraph, sampled with DFS (blue) and RW (red), original value in green.

values of consecutive samples according to Eq. (1). v_i are the values of a network property in the sample i, the sample $i+1$ is the next larger sample, the last v_i yields the property value of the original network.

$$monotonicity_i = \begin{cases} \uparrow_i & : v_i < v_{i+1} \\ =_i & : v_i = v_{i+1} \\ \downarrow_i & : v_i > v_{i+1} \end{cases} \tag{1}$$

Intuitively, a change of monotonicity is defined as a change in the direction of convergence as in Eq. (2) at position i to $i+1$. An intermediate equality of successive property values does not cause a change in monotonicity.

$$\uparrow_1 \ \ldots \ \uparrow_i \downarrow_{i+1} \downarrow_{i+2} \ \ldots \tag{2}$$

A *good* sampling algorithm approximates the inspected properties with as few changes in the monotonicity progress as possible. This property definition of a good sampling algorithm is also intuitive as it matches the perception of a good approximation.

4 Evaluation

We used GTNA [12] to implement and compute the sampling process. GTNA allows to integrate the network generation, the application of the sampling algorithms and the computation of the graph-theoretic metrics. We initialized the sampling algorithms with random start nodes, and executed 100 runs for the networks with $\leq 500,000$ nodes and 20 runs for the networks with $> 500,000$ nodes to reduce the impact of randomness in the sampling process. We sampled *1%* - *10%* in *1%* steps and *15%* on all networks, networks with $\leq 500,000$ nodes are also sampled with *20%* and *25%*.

We selected networks based on the, in Section 2 presented, sampling algorithm introducing related work. The analyzed networks are listed in Table 3. The networks are available at the SNAP project[3]. We identified four groups of networks based on their network type: social networks in a directed and undirected form, directed p2p networks and a directed webgraph. We calculated the earlier presented network properties (AC, CC, DD, and SP) to provide a useful evaluation for the property monotonicity of the sampled networks. We show the results in Table 4. The sampling algorithms are ranked according to their monotonicity properties. We ignored the transitivity in this evaluation, since all algorithms perform very good with respect to the monotonicity of the progression of this metrics values. As introduced in Section 3, we are concentrating on single scalar properties, and therefore we are deriving the submetrics as listed in Table 2. These single scalar values are combined as in Eq. (3) to have a single value per property. To avoid an overweighting of one of these submetrics, we are normalizing by the number of submetrics.

$$SP = \frac{(\text{cpl} + \text{diam} + \text{effectiveDiam})}{3}; \quad DD = \frac{(\text{avg} + \text{avg}_{\text{in}} + \text{avg}_{\text{out}} + \text{max})}{4} \tag{3}$$

[3] Stanford Large Network Dataset Collection, available at `http://snap.stanford.edu/data/`

Table 3. Evaluated networks, classified by their context.

Class	Type	Network	Nodes
social	directed	cit-HepPh	34,546
		cit-HepTh	27,770
		soc-Epinions1	75,879
social	undirected	ca-GrQc	5,242
		com-Youtube	1,134,890
p2p	directed	p2p-Gnutella30	36,682
		p2p-Gnutella31	62,586
webgraph	directed	web-Google	875,713

We want to be able to provide a recommendation of sampling algorithms for the complete network class. Therefore, we are summing up the single network properties of each network instance in a group to have a cumulated monoticitiy value per network property. To be able to compare these values over the borders of a group, we are normalizing the values by the number of networks within the group. This is formulated in Eq. (4). (*group* represents the network groups, listed in Table 3). The function M returns the value of a certain metric, computed on the given network. M_{group} collects the normalized, added values of the analyzed metrics. M is a placeholder for the computed metrics, listed in Table 2.

$$M_{group} = \frac{1}{|group|} \sum_{nw \in group} M(nw) \; ; \quad M \in \{AC, CC, DD, SP\} \tag{4}$$

To be able to provide a recommendation not only on the basis of a single property but for a monotone sampling of the complete network, we are cumulating the values of the single network properties of a group and are normalizing them by the number of metrics. This is shown in Eq. (5) and described by ν.

$$\nu_{group} = \frac{AC_{group} + CC_{group} + DD_{group} + SP_{group}}{4} \tag{5}$$

The Uniform Sampling (US), which is a random node selection algorithm, is used as ground truth. This algorithm is not practicable in real world applications, but as it produces a real random sample it is a good baseline to compare the monotonicity of the analyzed algorithms with. We are providing in Table 4 the by ν sorted ranking of sampling algorithms. Table 4 shows the domination of the random walk algorithms on the directed social network and the webgraph, the BFS algorithms are dominating the undirected social network and the P2P networks. An interesting fact is the stable presence of the US in the upper half of the ranking. Besides the webgraph, the simple algorithms are not far behind the forefront.

Table 4. Summed monotonicity rankings per network group

Rank	ν - directed social		ν - undirected social		ν - p2p		ν - webgraph	
1	RW	0.74	US	1.63	BFS	1.44	RW-DC	0.83
2	RW-DC	0.94	DFS	1.69	US	1.79	RS-DC	0.83
3	RJ	1.01	RDS	1.71	RS-DC	1.83	RJ	0.83
4	RW-MH	1.13	FS	1.91	FF	1.88	RW-MH	0.96
5	FS	1.51	BFS	1.98	RW-MH	1.92	FS	1.42
6	US	1.58	RS-DC	2.08	RS	2.04	RW	1.58
7	RS	1.63	SS	2.10	RJ	2.13	US	1.83
8	RS-DC	2.13	RS	2.43	RW-DC	2.21	RS	1.92
9	DFS	2.25	RJ	2.63	DFS	2.58	FF	2.69
10	SS	2.28	RW-MH	2.71	RDS	2.71	SS	3.27
11	RDS	2.42	RW	3.17	SS	3.17	BFS	3.52
12	FF	2.46	RW-DC	3.25	FS	3.33	RDS	3.69
13	BFS	2.92	FF	3.28	RW	3.96	DFS	3.71

Their is typically either BFS or RW within the best five sampling algorithms. We are showing in Figure 2(a) the plotted values of the results for the directed social network. We have built groups for AC, CC, DD, and the SP, the last group is showing the ν-values of the column from Table 4. The AC is completely monotone sampled by the RW and the RW-MH, the other algorithms are including changes in the monotonicity, a high amount of changes is especially visible at the group of BFS algorithms. The plot of the CC values showing similar results, the advantage of the random walk group is even higher, besides the DFS, all algorithms are better than the BFS algorithms. The monotonicity analysis of the DD metric shows similar monotonicity values for all sampling algorithms. The advantage of the RW algorithms is not distinctly present. The SP metric is well preserved by the BFS, the advanced algorithms of the BFS group and the group of RW algorithm produce similar monotonicity values. The advantage of the BFS is intuitive, as the shortest path properties are constantly converging by extending the exploration of the direct neighborhood in rings. The RW algorithms are traversing into the deep of the network and are usually producing longer paths. Beside the SP property, the results for the webgraph are similar. Figure 2(d) shows the plot for the webgraph. The advantage of the BFS on the SP property is not present on the webgraph.

The undirected social network, shown in Figure 2(b), is similarly sampled with all algorithms. The monotonicity of all samples is similar for the combined metrics of the network, DD and SP. The CC has a negative outlier with RW-DC which is not monotonously converging. The AC has two positive outliers BFS and RDS, which produce very monotone AC values.

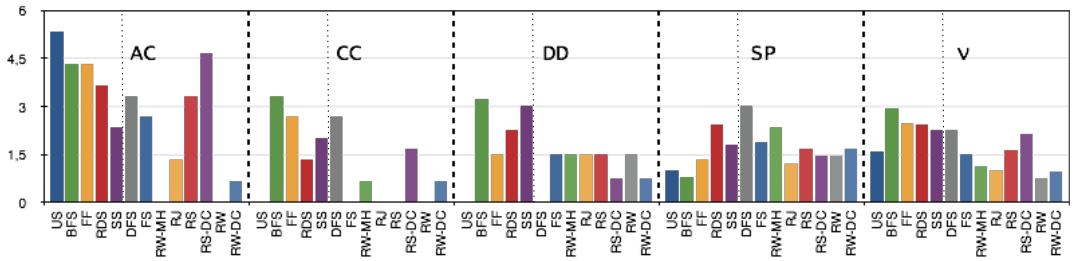
The P2P network, shown in Figure 2(c), is the only network with different monotonicity results, as shown in Table 4. The advantage of the RW algorithms is not present. The AC is dominated by BFS and FF, while the CC is dominated by BFS, RS and RS-DC. The DD is nearly equal monotone for all sampling algorithms, only the BFS has a change in the monotonicity but these values can be neglected with a value of 0.25. The SP property is slightly dominated by RW algorithms. Combined to the ν value, the algorithms perform with similar monotonicity for the complete network. There is no predominant algorithm for this group of P2P networks.

5 Conclusion

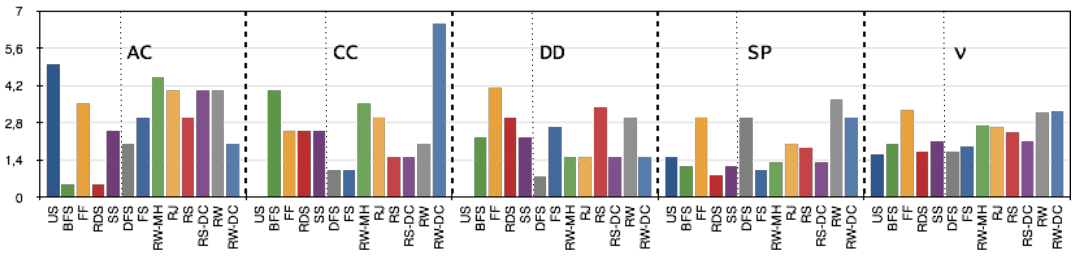
Today's networks are large, often too large to understand and process them directly. The computation of graph-theoretical properties on these large networks is a challenging task. We need to reduce the networks complexity and therefor the size of the network. The main technique for achieving this reduction of complexity is sampling by exploration. These sampling algorithms traverse through the network and collect the sample. Due to the network structure, some algorithms distort the networks properties. Many improved sampling algorithms were introduced to overcome these biased sampling processes. The properties of the sampled network are highly depending on the underlying network and the used sampling algorithm.

As the original networks properties are mostly unknown, we are not able to compare the sample properties with them. Therefore, it is undecidable if or when the quality demands are met. We propose another way to overcome this problem. We evaluate the convergence monotonicity to allow the development of an estimator for the common original network properties. The common properties are e.g. the degree distribution, and the shortest path distribution. To be able to provide a useful monotonicity evaluation, we chose networks based on the related work, which analyzes the newly developed sampling algorithms. To evaluate the convergence monotonicity, we are sampling multiple times and compute the properties of the samples. The convergence along increasing sample sizes is collected and aggregated. We rank the sampling algorithms with respect to the monotonicity values of their samples.

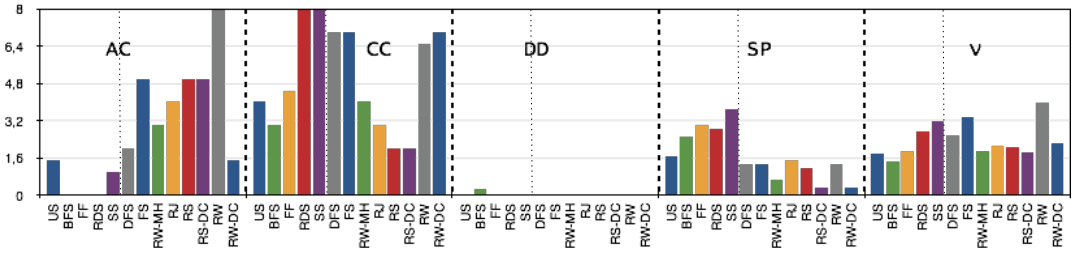
The main results of our evaluation are as follows: the complex algorithms enhancing the simple basic algorithms are not necessarily better in our monotonicity metric. Moreover, the simple algorithms random walk and breadth first sampling are the best algorithms of their group or at least at the forefront of their groups. The random walk algorithms are typically outperforming their breadth first sampling counterparts. The breadth first sampling algorithms are only similar good as the random walk algorithms on the P2P and undirected social networks. The



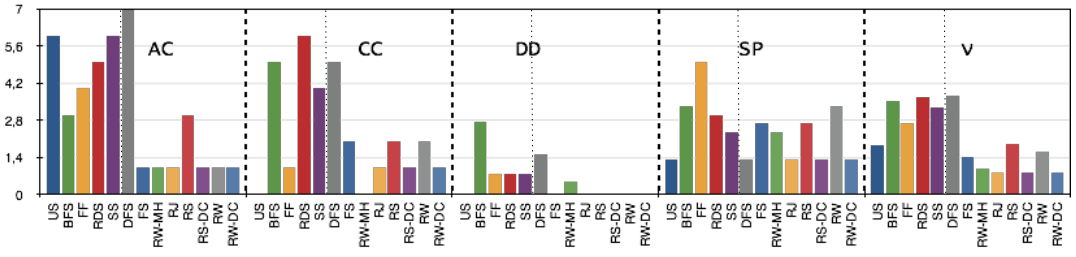
(a)



(b)



(c)



(d)

Fig. 2. Results of the monotonicity of the analyzed sampling algorithms: (a) directed social networks, (b) undirected social networks, (c) p2p networks, (d) webgraphs.

shortest path properties are well preserved by the breadth first sampling, only on the webgraph and the P2P networks are the monotonicity values not as good as the random walk algorithms. An interesting fact is that the monotonicity of the uniform sampling which is used as ground truth: The uniform sampling is never the most monotone algorithm but is in the better half of the ranking on all networks.

In the future work, we will develop a metric to measure the monotonicity of network properties which are not describable with a single scalar value, but with a distribution. The open question regarding this metric is: What is a monotonely converging distribution and how to measure this monotonicity? After answering this question, we want to develop estimators to assess the properties of the original network by analyzing sampled networks.

References

1. Chakrabarti, D., and Faloutsos, C. Graph Mining : Laws , Generators , and Algorithms. *ACM Computing Surveys 38*, March (2006).
2. Goodman, L. A. Snowball Sampling. *The Annals of Mathematical Statistics 32*, 1 (Mar. 1961), 148–170.
3. Heckathorn, D. D. Respondent-driven sampling: a new approach to the study of hidden populations. *Social problems 44* (1997), 174–199.
4. Krishnamurthy, V., Sun, J., Faloutsos, M., and Tauro, S. Sampling Internet Topologies : How Small Can We Go ? In *International Conference on Internet Computing* (2003).
5. Kurant, M., Markopoulou, A., and Thiran, P. On the bias of BFS (Breadth First Search). In *2010 22nd International Teletraffic Congress (lTC 22)* (Sept. 2010), IEEE, pp. 1–8.
6. Lee, S., Kim, P.-J., and Jeong, H. Statistical properties of sampled networks. *Physical Review E 73*, 1 (Jan. 2006), 016102.
7. Leskovec, J., and Faloutsos, C. Sampling from large graphs. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining* (New York, NY, USA, 2006), KDD '06, ACM, pp. 631–636.
8. Newman, M. Mixing patterns in networks. *Physical Review E 67*, 2 (Feb. 2003), 026126.
9. Newman, M. J. Assortative Mixing in Networks. *Physical Review Letters 89*, 20 (Oct. 2002), 208701.
10. Rasti, A. H., Torkjazi, M., Rejaie, R., Stutzbach, D., Duffield, N., and Willinger, W. Evaluating Sampling Techniques for Large Dynamic Graphs. Technical Report CIS-TR-08-01, Department of Computer and Information Science, University of Oregon, http://mirage.cs.uoregon.edu/pub/tr08-01.pdf, Sept. 2008.
11. Ribeiro, B., and Towsley, D. Estimating and sampling graphs with multidimensional random walks. In *Proceedings of the 10th annual conference on Internet measurement - IMC '10* (New York, New York, USA, Nov. 2010), ACM Press, p. 390.
12. Schiller, B., Bradler, D., Schweizer, I., Mühlhäuser, M., and Strufe, T. GTNA: a framework for the graph-theoretic network analysis. In *Proceedings of the 2010 Spring Simulation Multiconference* (San Diego, CA, USA, 2010), SpringSim '10, Society for Computer Simulation International, pp. 111:1—-111:8.
13. Stutzbach, D., Rejaie, R., Duffield, N., Sen, S., and Willinger, W. Sampling Techniques for Large, Dynamic Graphs. In *INFOCOM 2006. 25th IEEE International Conference on Computer Communications. Proceedings* (Apr. 2006), Ieee, pp. 1–6.