Evaluation of Graph Sampling: A Visualization Perspective

Yanhong Wu, Nan Cao, Daniel Archambault, Qiaomu Shen, Huamin Qu, and Weiwei Cui

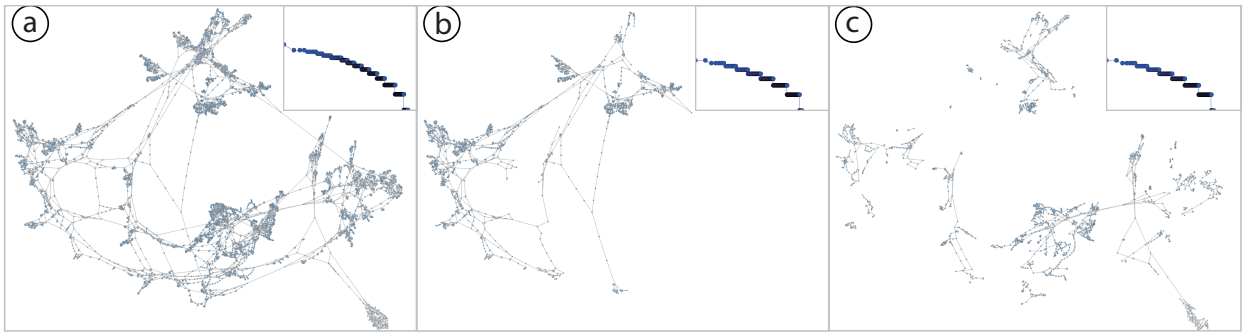

Figure 1. Comparison between two sampling strategies: (a) The original graph, (b) the graph sampled using *Random Walk*, and (c) the graph sampled using *Forest Fire*. The upper right corner of each subfigure shows the degree distribution. Subfigures (b) and (c) have similar degree distributions and an average degree of 2.4. However, the resulting visualizations of the samples are extremely different, indicating that the sampling strategy chosen can greatly influence the visual features present in the graph.

Abstract— Graph sampling is frequently used to address scalability issues when analyzing large graphs. Many algorithms have been proposed to sample graphs, and the performance of these algorithms has been quantified through metrics based on graph structural properties preserved by the sampling: degree distribution, clustering coefficient, and others. However, a perspective that is missing is the impact of these sampling strategies on the resultant visualizations. In this paper, we present the results of three user studies that investigate how sampling strategies influence node-link visualizations of graphs. In particular, five sampling strategies widely used in the graph mining literature are tested to determine how well they preserve visual features in node-link diagrams. Our results show that depending on the sampling strategy used different visual features are preserved. These results provide a complimentary view to metric evaluations conducted in the graph mining literature and provide an impetus to conduct future visualization studies.

Index Terms—Graph visualization, graph sampling, empirical evaluation

1 Introduction

As we enter the big data era, our capacity to collect and store networks has provided unprecedented opportunities to gain insight into our world. For example, as of late 2015, Facebook reported 1.6 billion monthly active users. Researchers can validate the "six degrees of separation" theory [36] and explore patterns in communication by analyzing social networks, such as Facebook. However, large-scale networks also pose unprecedented challenges to the fields of data mining and visualization. Graph mining algorithms typically exhibit high computational complexity. Graph visualization methods are also inherently limited by the complexity of algorithms used, screen space, visual clutter, and human perceptual capabilities when reading the data.

Many algorithms have been designed to address this scalability issue [22, 44]. Sampling is a commonly used technique in both data mining [35] and visualization [16, 45] because of its simplicity and efficiency. A number of sampling strategies have been developed, which range from simple node-based to advanced traversal-based schemes, have been developed based on different statistical models. Graph mining experts have evaluated the performance of these sampling strategies [35, 41] through metrics quantifying the topological properties preserved. These results have determined that no single strategy can preserve all the structural properties of the original network. Thus, guidelines have been proposed for selecting an appropriate graph sampling algorithm [32, 41].

Although sampling techniques have been thoroughly evaluated in the graph mining community, these studies have only considered the perspective of metrics (e.g. degree distribution and clustering coefficient) [35]. An important perspective that has not been considered is how the graph sampling affects the perception of graph visualizations. Such effects cannot be studied through metric evaluations alone. For example, often as part of a graph visualization task, a user may need to identify nodes with an abnormally high number of connections. Metric evaluations quantify how many original high degree nodes remain after sampling, but whether these nodes will still be *perceived* as high degree by a human user remains unknown. As another example, consider Fig. 1: In this figure, (a) shows the unsampled graph, whereas (b) presents *Random Walk* sampling, and (c) presents *Forest Fire* sampling. The two sampled graphs both preserve the power-law degree distribution of the original graph and have the same average node degree of 2.4. However, these graphs are visually different. Thus, the sampling algorithms may not influence the visualization equally in a perceptual sense, and studies are required to investigate this effect.

In this work, we study the influence of sampling strategies on node-link visualizations. In particular, we raise two research questions:

- *What are the visual factors that must be retained to make sampled graphs representative from the perspective of visualization?*
- *How do sampling strategies preserve these visual factors?*

For the first question, our pilot study finds that three visual factors significantly influence the representativeness of sampled node-link diagrams: cluster quality, high degree nodes, and coverage area. Targeting these visual factors, we conduct three experiments to evaluate the influence of sampling strategies on the perception of these visual

- *Y. Wu, Q. Shen, and H. Qu are with the Hong Kong University of Science and Technology. E-mail: {ywubk, qshen, huamin}@ust.hk*
- *N. Cao is with New York University Shanghai. E-mail: nan.cao@gmail.com*
- *D. Archambault is with Swansea University. E-mail: d.w.archambault@swansea.ac.uk*
- *W. Cui is with Microsoft Research Asia and is the corresponding author. E-mail: weiwei.cui@microsoft.com*

Manuscript received xx xxx. 201x; accepted xx xxx. 201x. Date of Publication xx xxx. 201x; date of current version xx xxx. 201x. For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org. Digital Object Identifier: xx.xxxx/TVCG.201x.xxxxxxx

factors. Evaluating all the algorithms is impractical for graph sampling because many algorithms have been proposed. Thus, we conduct a comprehensive survey of these algorithms and select five popular sampling strategies from the literature: *Random Node*, *Random Edge Node*, *Random Walk*, *Random Jump*, and *Forest Fire*. In addition, we discuss the differences between the findings of our studies and compare them to those in the graph mining literature. Our results show that *Random Edge Node* sampling and *Random Jump* best preserve global structure and cluster quality, whereas *Random Walk* helps users perceive high degree nodes. The performance of *Random Walk* and *Forest Fire* sampling is also sensitive to increased modularity.

2 Related Work

2.1 Node-Link Diagrams

Graph visualization is an active research area [25, 39, 49]. Node-link diagrams are among the most popular techniques for visualizing graphs because these diagrams are intuitive and space-efficient. However, previous research [19] has shown that the readability of node-link diagrams decreases with increasing size or density because of elevated visual clutter and occlusion. To increase the scalability of node-link diagrams, various techniques have been proposed which can be classified into two major categories: clustering and filtering.

Graph Clustering. Graph clustering methods abstract away groups of nodes and edges to reduce the visual complexity of the graph. In particular, node-based methods frequentely organize a graph into hierarchical structures by merging neighboring nodes together. For example, ASK-GraphView [1] enables users to explore a hierarchically organized network. Muelder and Ma [44] adopted treemaps to hierarchically organize graphs, so that they can be efficiently mapped onto 2D layouts. GrouseFlocks and TugGraph [4, 5, 6] support interactions to add and remove aggregated nodes on demand, which provides users with more flexibility to organize node clusters. Vehlow et al. [48] represented communities as abstract nodes and highlighted nodes between communities through partially aggregated graphs. Zinsmaier et al. [52] introduced a graph rendering technique, based on kernel density estimation, to efficiently cluster nodes. Glyphs are also widely used as a space-efficient way to represent multiple types of community information [11, 51]. Although these methods reduce the number of nodes, edge density may be higher than that in the original graph [30].

Edge-based clustering methods focus directly on the over-plotted lines in node-link diagrams. The nodes in these approaches are not hidden or aggregated in visualizations. Instead, edges with similar directions are visually bundled together in order to reduce edge crossings and emphasize directional patterns [18, 29]. Edge similarities in existing approaches are typically determined via explicit hierarchies [26], control meshes [13], force systems [27], clustering methods [18], and other approaches.

For example, Holten and van Wijk [27] applied attractive forces between edges to curve them into bundles. Gansner et al. [18] adopted multilevel clustering techniques to increase the scalability of bundling techniques. The main disadvantage of these approaches is that edge bundles can change the semantics of graphs and may fail to reflect the properties of the original graph structure.

Graph Filtering. Filtering-based methods preserve graph semantics by extracting subgraphs from an original graph. Stochastic and deterministic are two common types of filtering used in visualization [49]. Stochastic filtering, or sampling, randomly selects nodes or edges from a graph. Rafiei and Curial [45] compared the performance of three basic sampling strategies for large graph visualization. By contrast, deterministic filtering removes nodes and edges based on specific topological properties. For example, Jia et al. [30] filtered graphs by removing edges with low betweenness centrality, thus preserving connectedness and other graph features, such as cliques. Hennessey et al. [24] also adopted graph metrics, such as number of shortest paths and distance to the central node, to simplify graphs and obtain representative skeletons.

Deterministic methods often require some knowledge about the data and predefined thresholds. By contrast, stochastic filtering methods do not hold underlying assumptions about the data and are prevalent in the graph mining literature. Thus, we focus on stochastic filtering methods (i.e., sampling) in this study.

2.2 Graph Sampling Strategies

Graph sampling techniques randomly select nodes or edges to construct a subgraph that represents the original unfiltered graph. These methods have been studied in many fields, including statistics [23], data mining [22], and visualization [45]. To preserve particular graph properties for domain-specific goals, many sampling methods have been proposed [28]. These sampling techniques can be categorized into three main groups: node-based, edge-based, and traversal-based.

Node-Based Sampling. *Random Node (RN)* sampling [35] is the most common method; it selects a set of nodes uniformly at random from the graph. Using this set, an induced subgraph can be created by including every edge that connects a pair of nodes in the set. Although *RN* is simple and efficient, Stumpf et al. [47] demonstrated that it would not always preserve degree distribution in scale-free networks. Instead of sampling uniformly, advanced node-based methods select nodes with different probabilities based on graph properties. For example, in *Random Degree Node (RDN)* sampling [35], a node is selected with a probability that is proportional to its degree. Thus, *RDN* favors high degree nodes.

Edge-Based Sampling. *Random Edge (RE)* sampling [35] generates an induced subgraph by selecting edges uniformly at random. Several variants of *RE* exist. For example, *Random Node-Edge (RNE)* sampling [35] randomly selects a node and then randomly chooses an adjacent edge. *Random Edge-Node (REN)* sampling [45] first obtains a set of nodes from a uniform set of random edges. An induced subgraph is then computed by adding all the edges whose nodes are present in this set. Previous studies [35] have demonstrated that neither *RE* nor *RNE* preserves community structures because the resulting sampled graphs are often sparsely connected. Meanwhile, both *RE* and *REN* slightly favor high degree nodes because the probability of selecting a node increases with its degree.

Traversal-Based Sampling. Given the limitations of node-based and edge-based methods, researchers have investigated a third category of sampling techniques: traversal-based sampling. These methods are also known as topology-based sampling or sampling by exploration. An advantage of these methods is that connected graphs remain connected after sampling. *Depth First (DF)* and *Breadth First (BF)* sampling are two basic traversal-based sampling methods [14]. Starting from a randomly chosen node, *DF* selects nodes in depth first order [12]. Similarly, *BF* selects nodes in breadth first order [12]. In both *DF* and *BF*, only the starting node is randomly selected and the rest of the process is deterministic. Previous studies [31] have shown that *BF* and *DF* favor nodes with high degree and high page rank. *Snow-Ball (SB)* sampling, which is similar to *BF*, selects a fixed fraction of neighbors visited at each iteration. Lee et al. [34] suggested that *SB* suffers from a *boundary bias*, thereby making peripheral nodes (i.e., nodes sampled in the last round) miss a number of neighbors. *Forest Fire (FF)* sampling [37] is a probabilistic version of *SB* that randomly selects a seed node with incident edges and adjacent nodes getting "burned" away recursively with a probability p.

Another popular traversal-based sampling approach is *Random Walk (RW)* sampling [40]. Starting from a randomly selected node, *RW* selects the next node at random from the neighbors of the currently selected node. A problem occurs in *RW* when the graph has multiple connected components. In this case, components that do not contain the starting node will never be sampled. *Random Jump (RJ)* sampling circumvents this problem by randomly jumping to another node in the graph with a certain probability in each iteration. In general, *RW* and *RJ* favor high degree nodes.

A number of studies have evaluated the performance of these algorithms by measuring the properties of sampled graphs [3, 35, 41]. For example, Maiya and Berger-Wolf [41] evaluated several traversal-based sampling methods via measuring degree, clustering, and network reach. Ahmed et al. [3] extended these metrics and applied them to streaming graphs. However, these evaluations have all been performed from the perspective of metrics. The effect of these strate-

gies on the perception of node-link visualizations remains unknown. As graph mining becomes commonplace, this effect must be understood in the context of the visualizations. In this work, we aim to fill in this gap by conducting a study that directly addresses this question.

3 Definitions and Preliminaries

In this section, we present the definitions used in this paper and introduce the sampling strategies and data used in our study.

3.1 Notations and Definitions

Based on previous graph sampling studies [3, 35, 41], we adopt the following notation:

- A *graph* or *network* is represented by $G = (V,E)$ with the node set $V = \{v_1, v_2, \ldots, v_N\}$ and the edge set $E \subseteq V \times V$. $N = |V|$ is the number of nodes.
- A *node sample* S is a subset of the nodes of G, $S \subset V$.
- A *sampled graph* $G_S = (S, E_S)$ is the induced subgraph of G based on a node sample $S = \sigma(G)$ using sampling strategy σ with node set $S \subset V$ and edge set $E_S = (S \times S) \cap E$.
- A *sampling rate* $\phi = |S|/|V|$ is the percentage of node sampled S from V.

3.2 Sampling Strategies

Given the number of sampling strategies that exist, it is prohibitive to evaluate all of them. Thus, we choose five methods based on a survey of related work. Specifically, we select *Random Node (RN)*, *Random Edge Node (REN)*, *Random Walk (RW)*, *Random Jump (RJ)*, and *Forest Fire (FF)*. These methods cover the three major sampling categories defined by Hu and Lau [28]. For node and edge based sampling, we choose the most popular techniques [41, 47]. For traversal-based sampling, we evaluate all the three techniques selected by Leskovec and Faloutsos [35] because they differ substantially.

Node-Based Sampling. *RN* is selected as it is the most commonly used method for this category [9, 47]. *RN* selects a uniform set of random nodes and includes all the edges between any pair of nodes in this set. Thus, *RN* preserves node distributions because nodes are selected with equal probability. However, *RN* does not always preserve the degree distribution for scale-free networks [47]. This study can help understand the impact of this drawback on node-link visualizations.

Edge-Based Sampling. *RE* is the most basic edge-based sampling technique, but *REN* is more frequently used in metric studies [35, 41]. *REN* collects a random set of edges and the induced subgraph of the nodes present in this set. Thus, *REN* is selected from the edge-based sampling techniques to align with previous studies in graph mining.

Traversal-Based Sampling. Traversal-based sampling is a large and diverse category of sampling techniques. Thus, we select three representative schemes from this category: *RW*, *RJ*, and *FF*. These three techniques were used in the evaluation conducted by Leskovec and Faloutsos [35].

RW is the most commonly used traversal-based strategy [38]. It starts by randomly selecting a node and then performing a random walk on the graph from this node. However, *RW* cannot work directly on graphs with multiple connected components and can get trapped in dense clusters. A variant of this approach selects a node at random if the random walk does not discover a new node after a fixed number of iterations. In our experiments, we use the threshold proposed by Leskovec and Faloutsos [35], which is $100 * |V|$.

RJ is a widely used variant of *RW*. In this strategy, the random walker can jump to a random node $u \in V$ with probability c. In this study, we set $c = 0.15$: a value used in previous studies [35]. *RJ* partially circumvents the drawbacks of *RW*. However, *RJ* favors high-degree nodes and dense components. Although existing work [35] shows that *RJ* has a similar performance to *RW*, *RJ* is selected to determine if this property holds from a perceptual perspective.

FF is also selected. Unlike *RW* and *RJ*, *FF* does not favor high-degree nodes. *FF* avoids selecting nodes that are previously traversed by the algorithm, which causes it to behave differently. *FF* uniformly selects a random *seed* node and "burns" its x adjacent edges. The value of x is a geometrically distributed random number with a mean of $p_f/(1-p_f)$. In this work, p_f is set to 0.7, which is the value used in the graph mining literature [35]. After burning the edges, the endpoints are collected and the process is repeated until sufficient nodes have been visited.

BF, *DF*, and *SB* are deterministic, and thus, are excluded from our evaluation because these methods significantly depend on the starting node selected.

3.3 Graph Types and Layout Algorithms

Most real-world networks, such as social networks and biological networks, have a heavy-tailed degree distribution. Barabási and Albert [7] found that many of these *"scale-free networks"* share similar characteristics. First, the node degrees of scale free networks typically follow a power law, and a few nodes in these networks with considerably high degree. Also, these networks generally include central nodes that bridge two or more densely connected clusters. Due to their prevalence, many evaluations have been conducted to understand the properties of scale-free networks [21, 32, 34]. We also focus on scale-free networks in this study to stay aligned with the existing literature.

As our study focuses on the perceptual effect of sampling techniques, we need to control for the graph layout in our experiments. Many techniques have been proposed to produce graph layouts with good readability. In our study, we choose the approach of Dwyer [15] to achieve a good layout within a reasonable amount of time.

When developing a graph visualization system, graph sampling can be used as pre-processing before producing a layout or as post-processing to maintain a stable drawing when performing interactive exploration. If we use graph sampling as pre-processing, the time required for generating layouts is reduced. On the other hand, by sampling as a post-process, visual features present in the unsampled graph have the potential to be more readily preserved. For example, preserving node relationships is hard in a sampled graph as the incident edges can be filtered out. However, as many graph layout algorithms, such as force-directed methods, often place nodes in the same cluster close to each other, node relationships can be estimated by the distances between nodes. By keeping the node positions, people can infer node relationships even with few edges remained after sampling. Also, the experiments can be better controlled by avoiding potential layout confound effect. For these reasons, we draw the graph first and apply sampling as a post-process in our experiments.

4 Pilot Study

This study aims to understand the effect of our five sampling strategies on the perception of node-link visualizations. As there is no prior work on the topic of how important visual factors can affect graph similarity perception, we first conducted a pilot study to identify the important visual factors that should be preserved by graph sampling. This pilot study also helped determine a universal sampling rate, allowing us to fairly compare across data sets and techniques.

4.1 Participants and Apparatus

We recruited 10 participants (7 males, 3 females; aged 23-29 years, median: 25) for this study. All participants were students with a computer science background in our local university. The study was performed on a laptop computer with an external 23-inch display with a resolution of 1920 $\times$ 1080 pixels and 60 Hz refresh rate. Each participant took approximately 20 minutes to complete the pilot study.

4.2 Testing Data

In our pilot study, we used five real-world graphs (Table 1): a power-grid network (PowerGrid [50]), a hyperlink network (PoliticalBlogs [2]), and three social networks (Google+ [42], ResidentRating [17], and AdolescentHealth [43]). Graph generation models randomly insert specific patterns into graphs. As we did not want to bias the participants towards these features when judging similarity, we considered only real-world networks in our pilot study. All networks were regarded as undirected and unweighted graphs.

Network	N	D	AD	CC	PL
ResidentRating (*RR*)	217	0.1002	21.6	0.50	1.9
PoliticalBlogs (*PB*)	1,222	0.0220	27.4	0.32	2.7
AdolescentHealth (*AH*)	2,539	0.0054	13.7	0.33	2.3
PowerGrid (*PG*)	4,941	0.0005	1.3	0.08	19.0
Google+ (*G+*)	23,613	0.0001	3.3	0.17	4.0

Table 1. Network properties of the five data sets, where N is the number of nodes, D is the graph density, AD is the average degree, CC is the local clustering coefficient, and PL is the average shortest path length.

Figure 2. Pilot study interface. Original unsampled PG is visualized (top-left) with sampled versions at the following rates: 5%, 10%, 20%, 30%, and 40%. In this figure, PG is sampled with *Random Walk*.

4.3 Study Conditions

We used a within-subject design with three factors: data sets, sampling strategies, and random seeds for sampling strategies (Table 2).

	5	real-world data sets (*RR*, *PB*, *AH*, *PG*, *G+*)
	5	sampling strategies (*RN*, *REN*, *RW*, *RJ*, *FF*)
×	3	random seeds (3 different seeds)
	75	trials per participant
×	10	participants
	750	**trials in total**

Table 2. Conditions of pilot study and number of trials.

All participants viewed samples involving the same random seeds to ensure a fair comparison. The study began with a questionnaire about the participant's background. Then we explained the requirements and procedure to the participant. During this stage, the participants were encouraged to ask questions if they encountered problems. The study trials were divided into five blocks: one for each data set. The data sets were randomized within each block and the study blocks were counterbalanced using Latin squares.

4.4 Tasks and Procedure

This pilot study aims to collect two pieces of information to be used by the formal studies: a fixed sampling rate to be used by all the strategies and the visual factors to be investigated in the study.

T1: Determining the appropriate sampling rate. Sampling rate is an important parameter that significantly influences the sampling results. We assume that a "good" sampling technique can preserve the visual properties of graphs at a low rate when compared with a "bad" one. Therefore, we aim to find a sufficiently low sampling rate to distinguish the performance of these strategies without making the sampled graphs completely unidentifiable. For example, sampling a graph with a rate of 5% will produce an extremely sparse result, making comparisons with the original unsampled graph unnecessarily difficult. An existing study [37] suggested that sampling rates over 50% cause the strategies to behave the same. Therefore, in our pilot study, we test a set of sampling rates: 5%, 10%, 20%, 30%, and 40%, to identify an acceptable rate to use in the formal study.

T2: Identifying the most influential visual factors. In the pilot study, we also aim to identify important visual factors that influence the representativeness of the sampled graphs. Lee et al. [33] listed

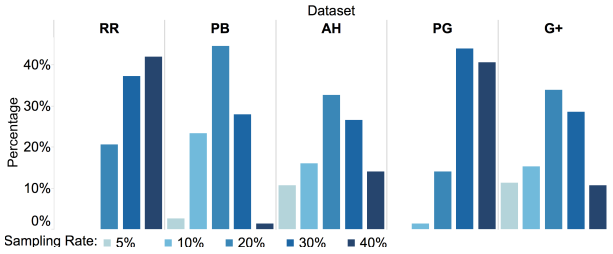

Figure 3. T1 results. This chart shows the percentage of votes for each sampling rate grouped by data set.

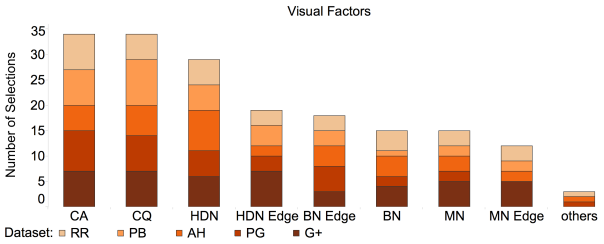

Figure 4. T2 results. This chart shows the number of votes for each visual factor and the vote distributions grouped by data set.

ten fundamental graph visualization tasks. Among them, we consider two tasks that do not have a numerical ground truth and heavily rely on human perception:

- Anomalies: graph elements that have extreme statistical values.
- Clusters: graph elements that share similar attribute values.

Other tasks, such as path finding, that could be performed by using automatic methods are excluded in this study. We compiled a list of eight different visual factors (Table 3) drawn from this category. We considered three types of anomalies: high degree nodes (*HN*), margin nodes (*MN*), and boundary nodes (*BN*). High degree nodes are the nodes that have considerably high degree compared with the average degree of the data set. Margin nodes are the nodes of considerably low degree that are not central to the graph structure. Boundary nodes are the nodes that bridge between two or more clusters in the data. Consequently, an edge that participates in a *HN*, *MN*, or *BN* relationship is considered an edge level anomaly of its corresponding type. Cluster quality (*CQ*) is also selected since clusters play an important role in terms of landmarks in graph analysis. Moreover, we include the "coverage area" (*CA*) as an overview task (Lee et al. [33], Section 4.4) as graph shape in the drawing can largely influence graph similarity perception. *CA* is defined as the overview of global structure provided by the sample when compared to the original unsampled graph. One can view it as the shape of the graph or the amount of area covered by the node-link diagram. In Fig. 2, when the sampling rate increases, we can observe a larger *CA* as the original graph's outline becomes clearer.

Network Level	Node Level	Edge Level
Coverage Area (*CA*)	High Degree Nodes (*HN*)	Edges Linking *HN*
Cluster Quality (*CQ*)	Margin Nodes (*MN*)	Edges Linking *MN*
	Boundary Nodes (*BN*)	Edges Linking *BN*

Table 3. The eight visual factor candidates that may influence the perceived representativeness of the sampled graphs.

For each set of study factors, we produced five sampled graphs using rates of 5%, 10%, 20%, 30%, and 40%. The corresponding original graph and the sampled graphs were visualized in a 2 × 3 grid (Fig. 2). No rotation or scaling was applied as each graph was presented once. All the graph samples were presented simultaneously. The original unsampled graph was presented in the top-left corner, and the sampled graphs were shown in the remaining grid cells. The participants were asked to select the lowest sampling rate that still preserved the unsampled graph structure. Simultaneously, the participants were required to name visual factor(s) that most influenced their decisions from the options listed in Table 3. The participants can enter their own visual factors if they are not listed. An informal post-study interview was conducted to understand the reasoning behind their selections.

4.5 Results and Discussion

Fig. 3 shows the results of T1. Generally, a small sampling rate was selected when the graph was large In particular, approximately 40% of the votes were given to either 5% or 10% rates for *G+*, which was the largest testing testing graph. By contrast, no participant believed that a sampling rate below 20% would be useful for *RR*, which was the smallest testing graph. An exception was *PG*, where a higher sampling rate was preferred despite its size. The post-study interview indicated that even for large sampling rates, a large area of *PG* was missed when using certain sampling strategies (Fig. 2), causing poor perceived coverage area. A possible explanation is the low clustering coefficient and high average shortest path length in this data set.

Compared with previous works, which either simply suggest a "one-size-fits-all" sampling rate [8] or describe strategies for selecting an appropriate sampling rate [41], the present study shows that the lowest acceptable rate, from a perceptual perspective, depends on graph properties, such as graph size and structure. To conduct a fair comparison among all strategies across data sets, we chose to fix the sampling rate. Our pilot study determined that a sampling rate of 20% or 30% could maintain the shape of graphs with a thousand or more nodes. Thus, to best discriminate among the different strategies, we selected a sampling rate of 20% for our formal study.

Fig. 4 shows the number of votes for each of the visual factors in our pilot study. Many of these visual factors could have been chosen for our experiment, but in order to have a focused study we select the three most important factors as judged by our participants. Therefore, in the next section, we present the results of three formal experiments based on the top three visual factors found by our pilot study: high degree nodes, cluster quality, and coverage area.

5 Formal Study

We aim to understand the effects of the five aforementioned sampling strategies on the three visual factors identified in our pilot study. Thus, we conducted three controlled experiments to test the effects of these strategies on these visual factors respectively.

5.1 Experiment I: Perception of High Degree Nodes

This experiment focuses on how sampling influences the perception of high degree nodes in the graph. Nodes with more incident edges are visually salient in node-link visualizations. However, sampling can influence the perception of these nodes by changing node degrees in a graph. The influence of sampling on node degree distributions has been considered in earlier works. However, no study has yet been conducted on how graph sampling influences the visual prominence of high degree nodes.

Hypotheses. Although the four sampling strategies, namely, *REN*, *RW*, *RJ*, and *FF* favor high degree nodes, we believe that *RW* can best preserve the visual prominence of these nodes. By visual prominence, we mean that if a node is considered high degree in the graph, it should still be perceived as a high degree node in the sample. The walker used by *RW* can be easily trapped inside dense local structures and has to move restrictively between adjacent nodes. Therefore, the incident edges of a high degree node have a higher probability to be selected, which enables the high degree nodes to retain more edges and be more identifiable. Although *REN*, *RJ*, and *FF* also favor high degree nodes, their incident edges are more likely to be filtered out. We conjecture that these nodes could be perceived as of lower degree after sampling. We also conjecture that *RN* is unlikely to preserve high degree nodes as it selects nodes randomly and uniformly. Finally, we believe this performance holds across all the data sets. Thus, we have the following hypotheses:

- **H1** It will be easier to perceive high degree nodes in the samples produced by *RW*.
- **H2** It will be more difficult to perceive high degree nodes in the samples produced by *RN*.
- **H3** H1 and H2 hold across data sets.

Testing Conditions and Data Generation. We conducted a within-subject study to compare the five sampling strategies. Four additional factors were considered in the experiment, namely, graph size, average degree of the high degree nodes, sampling strategy, and random seeds (Table 4).

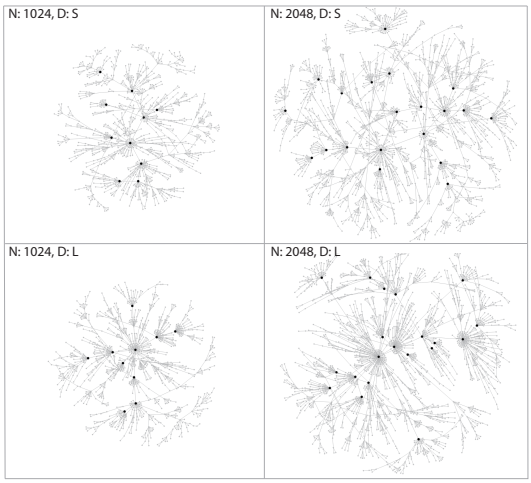

Figure 5. Four scale-free networks generated by Barabási-Albert model. Two graph sizes and two average degrees of high degree nodes are shown. The top 1% nodes of high degree nodes are highlighted.

	2	graph sizes (small=1024 nodes, large=2048 nodes)
	2	average degrees of high degree (small, large)
	5	sampling strategies (*RN*, *REN*, *RW*, *RJ*, *FF*)
	3	random seeds (3 different seeds)
×	3	repetitions
	180	trials per participant
×	20	participants
	3,600	**trials in total**

Table 4. Conditions of Experiment I and the total number of trials.

We create the test data sets (Fig. 5) using the Barabási-Albert model [7], a widely adopted scale-free network generator. This generator guarantees that nodes of high degree are embedded into the graphs that produce our stimuli. This model has two parameters: the number of nodes and the average node degree. A pilot study with four participants was conducted to determine the appropriate parameters to use in the experiment. Two graph sizes, i.e., 1,024 nodes for small and 2,048 nodes for large, were selected so that they would be visible on the screen while still having distinct high degree nodes. The average node degree was set to two to avoid unnecessary visual clutter.

To determine the two levels of the average degree for high degree nodes, we randomly generated 100,000 graphs for each size (1,024 nodes and 2,048 nodes). For each graph, the top 1% of nodes with high degree were considered high degree nodes. The graphs were sorted by the average degree of this 1% from smallest to largest. For the small and large graph data sets, two graphs were randomly selected from the first and last third of this sorted list as having "small" and "large" average high degree. Thus, four data sets were used (2 graph sizes × 2 average high degree) for our experiment (Fig. 5).

Task and Procedure. This experiment aims to test if high degree nodes in the original data sets are still perceived as high degree nodes in the samples produced by each sampling strategy. The participants were required to compare the degree of the highlighted node to other nodes in this graph and state if the highlighted node could be perceived as high degree. Each stimulus was a sampled graph in which the entire set of high degree nodes of the original graph were highlighted. As all the original high degree nodes are highlighted, it limits priming effects. If the participant still perceived a highlighted node as high degree, he or she clicked on the node to report it. If the participant felt that none of the highlighted nodes could be perceived as high degree, they could click on a button to indicate this response.

The study began with a training session in which the participants can familiarize themselves with the system and the task. The experiment was divided into two blocks (large graph and small graph), each consisting of 90 tasks. Trials were randomized within each block and

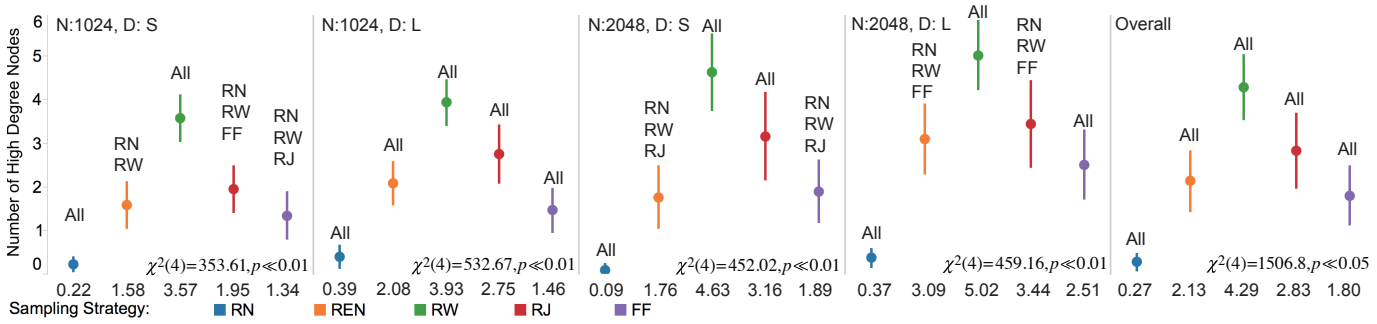

Figure 6. The average number (bottom number of each column) of selected high degree nodes for each graph condition. Error bars indicate 95% confidence intervals. Pairwise significant differences are indicated above each bar. Friedman test statistics appear at the bottom-right corners.

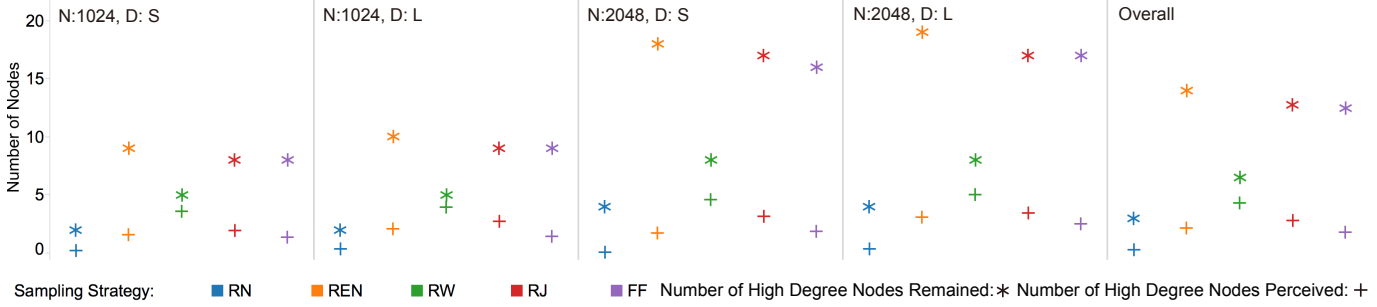

Figure 7. The numbers of selected high degree nodes and actual high degree nodes for different sampling strategies under each graph condition.

blocks were counterbalanced between the participants. The stimuli were mirrored and flipped to ensure a fair comparison and to avoid memorization of previous answers. The entire experiment took approximately 30 minutes to finish, and the participants could rest after finishing a study block.

Participants and Apparatus. We recruited 20 participants (8 females, aged 22 to 30 years (mean = 24.8, SD = 2.1)) with normal or corrected-to-normal vision. All the participants were undergraduate students, graduate students, or staff in the computer science department of our local university. The study was conducted on a laptop computer with an external 23 inch display with a resolution of 1920 $\times$ 1080 and a refresh rate of 60Hz. The sampled graph was displayed in a window with 1000 $\times$ 1000 pixels (26.5 cm $\times$ 26.5 cm). The highlighted nodes had a radius of 5 pixels and were black, whereas all the other nodes had a radius of 2 pixels and were filled in gray. The highlighted nodes turned red when clicked. All the edges were gray lines with a width of 1 pixel. Participants used a mouse for interaction.

Results. First, we analyzed the performance of the different sampling strategies for all the graph conditions. To check for normality, we ran the Shapiro-Wilk test. The result showed that the data was not always normal. Thus, we use a Friedman test to determine the statistical significance with a standard statistical level $\alpha = 0.05$. Post-hoc analysis was conducted with a Nemenyi-Damico-Wolfe-Dunn test.

We then analyzed how the strategies performed for each graph condition. When dividing the data by graph size and average high degree, we applied a Bonferroni correction, reducing the significance level to $\alpha = 0.0125$. The post-hoc analysis was conducted as above.

Fig. 6 shows the average number of selected high degree nodes with pairwise significant differences indicated above each error bar. Significant differences were identified overall and for each graph condition. Statistics related to these tests are reported in the bottom right corner of each subfigure.

Discussion. Our results provide evidence that the sampling technique chosen influences how high degree nodes in the graph are perceived in the sample. The participants tended to correctly perceive high degree nodes in the samples produced by *RW*, confirming **H1**. Fig. 7 compares the results of our experiment with counts of the number of high degree nodes that remain in the sample. In these counts, *REN* preserves the most high degree nodes and *RW* the fewest. Our experiment shows that for *RW*, however, most of these remaining high degree nodes are still perceived as high degree by the participant. Although other strategies may preserve more of these nodes, the participants tended not to perceive them as high degree.

A possible explanation is that *RW* often collects edges incident to high degree nodes as it is not forbidden from revisiting previously selected nodes. *RJ*, *REN*, and *FF* perform similar in this study. Despite preserving more high degree nodes than *RW* in terms of numbers (Fig. 7), these methods performed worse for various reasons. *RJ* allows the walker to escape from "traps", which causes the incident edges of high degree nodes to be less likely selected. *REN*, an edge-based strategy, does not favor the incident edges to high degree nodes. *FF* does not allow node revisit, which also reduces the probability of selecting the incident edges of high degree nodes. *RN* does not perform well in either experiment as all the nodes have the same probability of being selected, which supports **H2**. This trend was seen across all the four graph conditions, which also verifies **H3**.

In addition, all five sampling strategies have large confidence intervals except for *RN*. From our post-study interview, we found that participants often have different standards when identifying a node as high degree. Some participants chose a constant value (degree 5 or 10) and considered anything above this threshold high degree. Others, used different thresholds depending on the condition.

Our experiment provides evidence that the choice of sampling strategy can influence the perception of high degree nodes in graph visualizations. This human centered perspective compliments the results of metric experiments performed in the graph mining community.

5.2 Experiment II: Perception of Cluster Quality

A cluster is a group of densely connected nodes with only a few edges that connect nodes from the cluster to nodes that are not in the cluster. Clusters are more frequent in real-world graphs than in random graphs. As clusters provide important visual landmarks, they should be preserved in node-link visualizations after sampling. However, different sampling strategies may variably influence the node and edge distributions in different ways, thereby impacting the legibility of clusters in sampled graphs. Existing work in the graph mining community [20] studies clusters from the perspectives of metrics, such as modularity, which can be complimented by perceptual studies. Thus, we conduct an experiment to investigate the influence of sampling strategies on the perception of clusters.

Hypotheses. Sampling strategies can influence the perception of clusters in several ways. For example, *RN* may affect the identification

of clusters by blurring the boundaries between clusters. By contrast, *RW* and *FF* could get trapped inside the clusters when sampling and leave other clusters out of the sample entirely. The performance of *RW* and *FF* is likely to be affected by graph modularity, indicating how easily a random walker can escape clusters. Thus, we have the following hypotheses:

- **H4** *REN* and *RJ* will best preserve the perceived cluster quality in sampled graphs.
- **H5** *RN* will struggle in preserving the perceived cluster quality.
- **H6** The performance of *RW* and *FF* likely depends on modularity and is independent of graph size.

Testing Conditions and Data Generation. We conducted a within-subject study to compare the five sampling strategies. Three additional factors were tested: graph size, modularity, and random seed selection (Table 5).

	2	graph sizes (small=1024 nodes, large=2048 nodes)
	2	graph modularities (low, high)
	3	random seeds (3 different seeds)
×	3	repetitions
	36	trials per participant
×	20	participants
	720	**trials in total**

Table 5. Conditions of Experiment II and the number of trials.

Since this experiment focuses on cluster quality, cluster number and modularity are two important factors. Accordingly, we chose Sah et al.'s model [46], which could explicitly specify the number of clusters to embed into the graph along with an estimate of graph modularity, to generate our graphs. As a particular number of clusters are guaranteed to be present in the original graph, we can use this data to generate our stimuli. Similar to Experiment I, we conducted a pilot study with four participants to determine the other graph parameters. The same graph sizes used in Experiment I (1024 nodes and 2048 nodes) were used. The number of clusters for the large and small graph sizes were set to eight and four, respectively. High and low modularity levels for the experiment were set to 0.5 and 0.15, respectively. Thus, four graphs (2 graph sizes $\times$ 2 graph modularities) were generated and used by all participants (Fig. 8). These graphs, in turn, were sampled by our five strategies to create our stimuli.

Task and Procedure. It is difficult to judge the perceived quality of cluster preservation without a reference graph. Thus, the participants were presented with a 2×3 matrix of visualizations (similar to Fig. 2) for each trial. The upper left illustration showed the original graph. The remaining five illustrations presented the sampled graphs induced by the five sampling strategies. The participant was asked to rate each sampled graph using a five-point Likert scale, indicating how well the clusters were preserved.

Participant information was already collected in Experiment I. Thus, we directly started this study with a training session. During this session, we provided a brief description of the task and provided four practice trials. Each practice trial had a unique graph size and modularity. During this phase, we provided several candidate factors to the participant that might influence cluster quality, including cluster size, number, and density. Participants were instructed to determine the priorities of these factors on their own during the experiment.

The trials were divided into two blocks by graph size. These blocks were counterbalanced between participants. Each block contained $2 \times 2 \times 3$ trials. The trials within each block were randomized to counteract learning a fatigue effects. The stimuli were mirrored and flipped to ensure a fair comparison and to avoid memorization of previous answers. Participants could take a break between blocks. On average, the experiment took about 15 minutes to complete. After the experiment, an informal interview was conducted to investigate which factors that played important roles in the perception of cluster quality.

Participants and Apparatus. The same participants were recruited, and the same devices were used in this experiment as those in Experiment I. The node-link visualizations of the original and the five sampled graphs were displayed in a 2×3 matrix. Each window

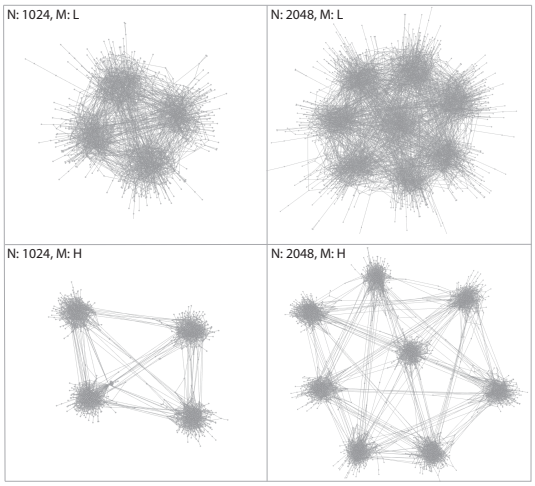

Figure 8. Four scale-free networks generated by Sah et al. [46]'s model based on graph size and modularity.

was in 500×500 pixels ($13.3cm \times 13.3$ cm). Nodes were colored gray and had a radius of two pixels. Edges were colored gray and had a width of one pixel. The participants used a mouse for interaction.

Results. The rating scores did not follow a normal distribution according to a Shapiro-Wilk test. Thus, we used Friedman tests to determine the statistical significance with $\alpha = 0.05$ for the overall graph conditions and a Bonferroni corrected value of $\alpha = 0.0125$ when dividing our results by size and modularity. A post-hoc Nemenyi-Damico-Wolfe-Dunn test was run to determine pairwise significance between the sampling strategies.

Fig. 9 shows the average rating of cluster quality with significant pairwise differences listed above each error bar. The performance of these strategies is significantly different overall and under each graph condition as indicated at the bottom of each subfigure.

Discussion. *REN* and *RJ* exhibit similar perceived cluster quality and generally perform better than the other strategies in the study, which supports **H4**. As these strategies are not restricted to local areas of the graph and have good spread, the strategies are likely to sample from all clusters. *REN* and *RJ* can collect more edges. Thus, they are likely to preserve the perceived density of the clusters as well as the number of clusters perceived by the participant. Although *RN* preserves node distributions, fewer edges are retained, which causes the cluster boundaries to become obscure and difficult to identify. In our interview, some participants could not perceive any clusters in *RN* sampled graphs with low modularity.

H5 is only partially supported as *RN* has comparable or even better performance than *FF* for high modularity graphs. Although the effect size is small, the participants reported different reasons for their ratings. For *RN*, although the graph density is low after sampling, the positions of the remaining nodes provide an important cue for estimating cluster structures. *FF* has a higher density as it iteratively explores incident edges. However, distant clusters could be left out completely.

The performance of *RW* varies with graph modularity, but is insensitive to graph size, providing support for **H6**. The random walker of *RW* can be trapped by densely connected components. These components are well preserved but the strategy can miss clusters. Participant interviews confirm that missing entire clusters is particularly severe for *RW* and influences their perception of cluster quality.

We conducted a metric evaluation of our graphs to provide a comparison (Table 6). *RW* and *FF* perform best at preserving average node number and the edge density for each cluster. However, *REN* and *RJ* perform best according to the perceived cluster quality. To investigate which metrics best match the ratings in our experiment, we compute several metrics on our stimuli (Table 6). The bold numbers in this table are the average metric values that most closely match those found in the original graph before sampling. We found that *REN* and *RJ* perform well in terms of average cluster number. These algorithms sample globally, preserving most of the clusters in the original graph.

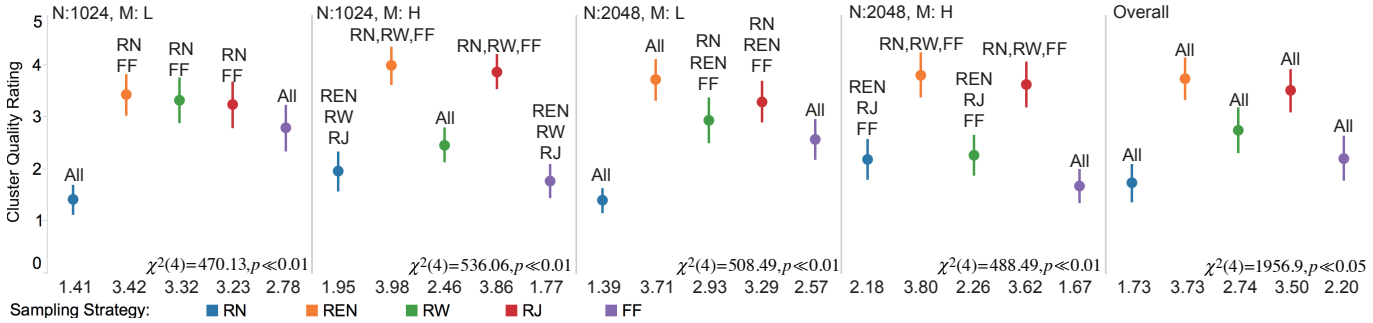

Figure 9. The average rating (bottom number of each column) of cluster quality for each graph condition. Error bars indicate 95% confidence intervals. Pairwise significant differences are indicated above each bar. Friedman test statistics appear at the bottom-right corner of each chart.

Graph	N: 1024, M: L				N: 1024, M: H				N: 2048, M: L				N: 2048, M: H				Overall			
	M	CN	CS	ER	M	CN	CS	ER	M	CN	CS	ER	M	CN	CS	ER	M	CN	CS	ER
Original	0.55	4	256	0.50	0.68	4	256	0.15	0.67	8	256	0.50	0.80	8	256	0.15	0.68	6	256	0.33
RN	0.77	4.6	14.0	0.15	0.80	4.3	15.9	0.07	0.84	2.4	21.7	0.08	0.88	4.1	26.4	0.02	0.82	3.8	19.5	0.08
REN	0.62	6	14.0	0.15	0.72	**4.0**	50.0	0.03	0.73	**8.0**	50.2	0.17	0.85	**8.0**	50.4	0.02	0.73	6.2	48.4	0.10
RW	0.59	**4.2**	**48.2**	0.20	0.57	4.4	48.0	**0.20**	0.70	**8.0**	51.5	**0.19**	0.74	6.0	68.2	**0.03**	**0.65**	5.6	**54.0**	0.16
RJ	0.60	4.9	41.5	0.22	**0.69**	**4.0**	**50.5**	0.03	0.72	**8.0**	51.0	0.16	**0.83**	**8.0**	51.0	0.02	**0.71**	6.2	48.5	0.11
FF	**0.56**	4.9	41.8	**0.27**	0.45	6.5	33.5	0.62	**0.69**	7.5	**53.9**	0.17	0.66	5.0	**80.8**	**0.03**	0.59	**6.0**	52.5	**0.27**

Table 6. Metrics computed on the experimental stimuli: average modularity (M), average number of clusters (CN), average cluster size (CS), and average external/internal edge ratio (ER). Communities were computed using Blondel et al.'s method [10]. Numbers in bold indicate the values that most closely match those of the original unsampled graph.

Cluster number is probably of higher perceptual importance as it is easier to be perceptually estimated. This finding was supported by our participant interviews. Many participants reported that they favored cluster number over cluster density or inter-cluster links when rating perceived cluster quality.

5.3 Experiment III: Perception of Coverage Area

The coverage area of a sample provides users with an overview of the data and the patterns in it. It is inherently hard to quantify using metrics. A good sampling strategy should preserve this overview of the graph. Thus, understanding how different sampling strategies influence this coverage area is crucial. It is important to study this problem from a perceptual viewpoint, complementing results of metric experiments as graphs with the same statistics can be perceived differently.

Hypotheses. *REN* and *RJ* can generate good node and edge distributions simultaneously. Thus, we conjecture that the graphs created using these two strategies will have a larger perceived coverage area. For a scale-free networks, *RN* can create sparse and disconnected samples, which reduces the coverage area. For this reason, we believe that *RN* will likely have lower perceived coverage area. *RW* and *FF* are both sensitive to graph modularity. Thus, these strategies are likely to sample a greater number of nodes from locally dense components. Thus, we have the following hypotheses:

H7 *REN* and *RJ* likely have the largest perceived coverage area.
H8 *RN* is likely to have a smallest perceived coverage area.
H9 *RW* and *FF*'s performance vary depending on graph properties.

Testing Conditions and Data Generation. We adopted a within-subject study design to compare the five sampling strategies on perceived coverage area. Four additional factors are tested: graph model, graph size, model parameters, and random seeds (Table 7):

	2	graph models (Barabási-Albert model [7] and Sah et al.'s model [46])
	2	graph sizes (small=1024 nodes, large=2048 nodes)
	2	corresponding parameters for each graph model
	3	random seeds (3 different seeds)
×	3	repetitions
	72	trials per participant
×	24	participants
	1728	**trials in total**

Table 7. Testing conditions of Experiment III and number of trials.

In contrast to our previous studies, coverage area is a global property and thus very hard to quantify. In order to avoid the bias induced from a single graph generation model, this experiment used both models present in experiment I and II. The procedures are the same ones as described in Sections 5.1 and 5.2.

Task and Procedure. We followed a similar procedure to that of Experiment II. As it is difficult to judge how well the coverage area is preserved without a reference, six visualizations (the original graph and five sampled graphs) were displayed in a 2×3 matrix similar to Fig. 2. The participants rated the sampled graphs using a five-point Likert scale, according to the coverage area of the sample.

After a pre-experiment interview, we provided the participants with a training session, which consists of four practice trials. This session helped the participants to become familiar with the task and system.

The trials were divided into four blocks based on graph models and sizes. Blocks were counterbalanced between participants. The $2 \times 3 \times 3$ trials within a block were randomized to avoid learning and fatigue effects. Visualizations were rotated and mirrored to avoid memorization effects. Each participant took approximately 30 minutes to complete this experiment. A post-study interview was conducted to understand how the strategies influenced the participants' decisions.

Participants and Apparatus. A total of 24 participants (8 females, aged 21 to 30 (mean = 24.8, SD = 1.9)) from our university were recruited for this experiment. All of them had normal or corrected-to-normal vision. The setup in Experiment II was reused here.

Results. The ratings did not follow a normal distribution according to the Shapiro-Wilk test. Thus, we also adopted a Friedman test with $\alpha = 0.05$ for the overall graph conditions and a Bonferroni corrected significance level of $\alpha = 0.00625$ when dividing the data by factor. A post-hoc analysis was conducted as in previous experiments.

Fig. 10(a) shows the average rating of the perceived coverage area under each graph condition, whereas Fig. 10(b) shows the perceived coverage area ratings overall. Friedman statistics are provided in the bottom right corner of each subfigure.

Discussion. H7 is partially supported as *REN* and *RJ* generally perform best in this experiment. *FF* surprisingly exhibits a comparable performance with *RJ* for Barabási-Albert [7] (*BA*) generated graphs. Graphs generated by *BA* have a low average node degree, which impedes the "fire" in *FF*. By frequently restarting at new random nodes, *FF* behaves similar to *RJ*, explaining their similar performance.

H8 is not supported because *RN* has significantly better performance than *RW* for six of the conditions and overall. In the post-study interview, many participants reported that although *RN* drops many edges, it preserves an overall node distribution, which helps estimate

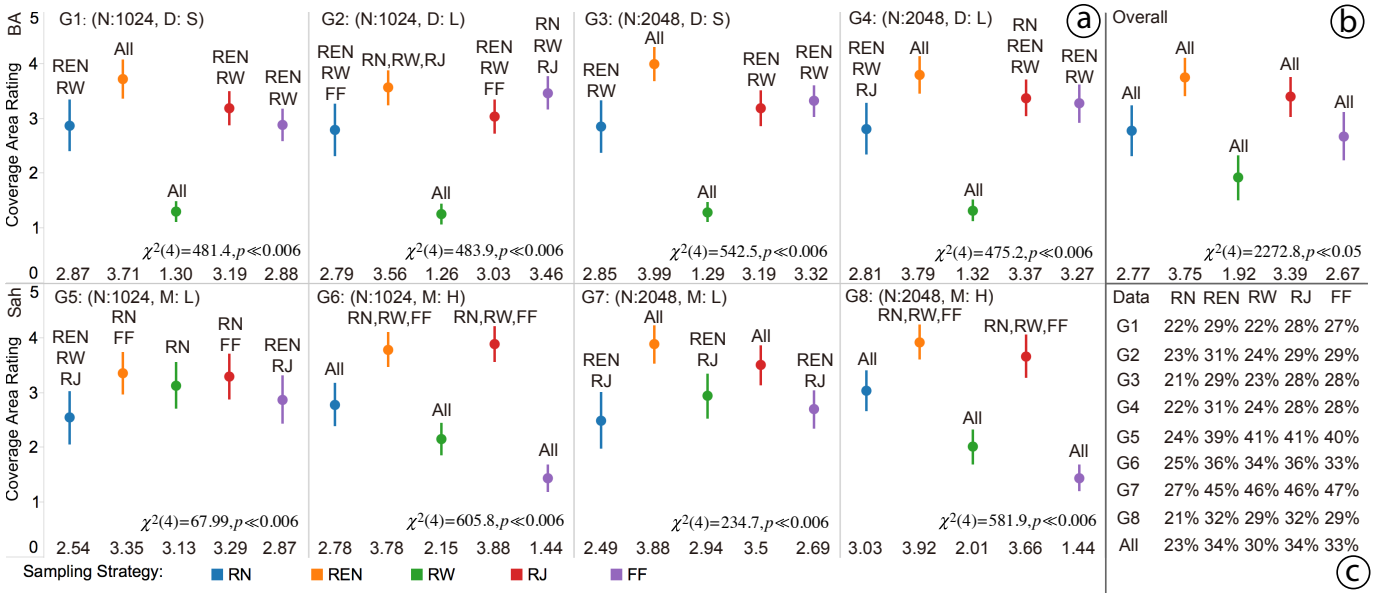

Figure 10. (a) Average rating (bottom number of each column) of perceived coverage area for each condition. The top row shows the graphs generated using Barabási-Albert model [7] while the bottom row shows the graphs generated using Sah et al.'s method [46]. (b) Average rating for the perceived coverage area under all of the eight conditions. (c) Percentage of pixels covered by the sample graph when compared with the unsampled graph. Friedman test statistics are reported in (b) and (c).

the graph shape. By contrast, *RW* misses many areas of the graph entirely, which makes *RN* have a larger perceived coverage area.

H9 is supported as *RW* and *FF* vary under different graph conditions. Notwithstanding *FF*, the performance of *RW* varies with graph modularity. From Experiment II, *RW* can sometimes get trapped in dense connected local structures, which explains why *RW* has low perceived coverage area when the graph modularity is high.

We also computed the percentage of pixels covered by the nodes and edges of each visualization compared with the unsampled graph (Fig. 10(c)). *REN* and *RJ* perform well for both models. *RW* is similar to *RN* in terms of these percentages for graphs generated with the *BA* model [7]. However, the perception rating of coverage area for *RN* is significantly higher, as reported by the participants in the study. Similarly, the perceived coverage area of *RW*, *REN*, and *RJ* are very different from the percentages for the graphs generated by Sah et al.'s model [46]. These findings show that the perceived coverage area of sampled graphs can differ from the metric results.

6 Discussion

Our experimental results show that a human-centered perspective can complement metric evaluation when evaluating graph sampling approaches. Depending on the sampling algorithm used, graph features can be perceived differently when compared to metric experiments. In Experiment I, when a high degree node was preserved in *Random Walk*, it was most likely perceived as high degree node. In the metric experiments, *Random Walk* preserved a small number of high degree nodes. In Experiment II, many cluster quality metrics were computed but only the results of *Cluster Number* matched the perceived quality. In Experiment III, the percentage of pixels covered by *Random Node* and *Random Walk* are similar, but participants felt that *Random Node* has a larger perceived coverage area. From these results, we provide the following recommendations for sampling network visualizations:

Recommend *Random Edge Node* and *Random Jump* for Global Structure and Cluster Quality. *Random Edge Node* and *Random Jump* perform well in both Experiment I and III. The perceived coverage area and cluster quality are high using these sampling approaches.

Recommend *Random Walk* for Perceived High-Degree Nodes. *Random Walk* has the best performance under all graph conditions in Experiment I. When high degree nodes are present in the sampled graph, they are perceived as high degree nodes. However, this strategy has lower perceived cluster quality and coverage area. *Random Walk* can use multiple starting nodes to alleviate this issue.

Avoid *Random Node* unless for specific requirements. The other sampling strategies outperform *Random Node* in experiment I and II. Although *Random Node* maintains a good coverage area, its level of edge filtering makes it difficult for people to identify structures. Thus, we would recommend avoiding it unless there is a clear reason to use it such as reducing the visual clutter caused by a large number of edges.

***Random Walk* and *Forest Fire* are Modularity sensitive.** We found that *Random Walk* and *Forest Fire* are more sensitive to modularity than to graph size. Thus, on graphs with these properties, we must consider what visual factors we want to preserve as some can be obfuscated by the sampling process.

7 Conclusion

In this work, we provided the first study of how graph sampling strategies can influence the perception of node-link visualizations. Our pilot study identified three important visual factors that should be preserved, namely, high degree nodes, cluster quality, and coverage area. We conducted three controlled experiments to evaluate the effects of these sampling strategies on the perception of these visual factors. Our results show that *Random Edge Node* and *Random Jump* perform well on the perceived cluster quality and coverage area. For *Random Walk* and *Forest Fire*, the visual factors can vary with graph properties such as modularity. Although *Random Walk* retains fewer high degree nodes in sampled graphs, these nodes are more likely to be perceived as high degree when compared to other strategies. These results compliment the metric evaluations conducted in the graph mining community and provide further impetus for the study of the perceptual effects of sampling on visualizations.

In future work, we would like to examine false positives. In particular, we want to investigate whether users see a high degree node or a cluster that does not exist in the original data after sampling. Secondly, we have only looked at perceived cluster quality at a high level. It would be important to investigate the perception of other cluster properties in graph samples such as cluster density or fuzziness. Finally, we need to investigate the advantages and disadvantages of sampling, from a perceptual perspective, and apply our results in real settings.

8 Acknowledgments

We gratefully thank all the anonymous reviewers for their valuable comments. This research was supported in part by the National Basic Research Program of China (973 Program) under Grant No. 2014CB340304 and a grant from Microsoft Research Asia.

REFERENCES

[1] J. Abello, F. Van Ham, and N. Krishnan. Ask-graphview: A large scale graph visualization system. *IEEE Trans. Vis. Comput. Graph.*, 12(5):669–676, 2006.

[2] L. A. Adamic and N. Glance. The political blogosphere and the 2004 u.s. election: Divided they blog. In *Proc. of the 3rd Intl. Workshop on Link Discovery*, LinkKDD '05, pages 36–43. ACM, 2005.

[3] N. K. Ahmed, J. Neville, and R. Kompella. Network sampling: From static to streaming graphs. *ACM Trans. on Knowledge Discovery from Data (TKDD)*, 8(2):7, 2014.

[4] D. Archambault, T. Munzner, and D. Auber. GrouseFlocks: Steerable exploration of graph hierarchy space. *IEEE Trans. Vis. Comput. Graph.*, 14(4):900–913, 2008.

[5] D. Archambault, T. Munzner, and D. Auber. Tugging graphs faster: Efficiently modifying path-preserving hierarchies for browsing paths. *IEEE Trans. on Visualization and Computer Graphics*, 17(3):276–289, 2011.

[6] D. Archambault, H. C. Purchase, and B. Pinaud. The Readability of Path-Preserving Clusterings of Graphs. *Computer Graphics Forum*, 29(3):1173–1182, 2010.

[7] A.-L. Barabasi and R. Albert. Emergence of scaling in random networks. *Science*, 286(5439):509–512, 1999.

[8] N. Blagus, L. Subelj, and M. Bajec. Empirical comparison of network sampling techniques. *CoRR*, abs/1506.02449, 2015.

[9] N. Blagus, L. ubelj, G. Weiss, and M. Bajec. Sampling promotes community structure in social and information networks. *Physica A: Statistical Mechanics and its Applications*, 432:206 – 215, 2015.

[10] V. Blondel, J. Guillaume, R. Lambiotte, and E. Lefebvre. Fast unfolding of communities in large networks. *J. of Statistical Mechanics*, page P10008, 2008.

[11] N. Cao, Y. Lin, L. Li, and H. Tong. g-miner: Interactive visual group mining on multivariate graphs. In *Proc. of the 33rd Annual ACM Conf. on Human Factors in Computing Systems, CHI 2015*, pages 279–288, 2015.

[12] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein. *Introduction to algorithms*, volume 6. MIT Press Cambridge, 2001.

[13] W. Cui, H. Zhou, H. Qu, P. C. Wong, and X. Li. Geometry-based edge clustering for graph visualization. *IEEE Trans. Vis. Comput. Graph.*, 14(6):1277–1284, 2008.

[14] C. Doerr and N. Blenn. Metric convergence in social network sampling. In *5th ACM Workshop on HotPlanet*, pages 45–50. ACM, 2013.

[15] T. Dwyer. Scalable, versatile and simple constrained graph layout. *Comput. Graph. Forum*, 28(3):991–998, 2009.

[16] G. Ellis and A. Dix. A taxonomy of clutter reduction for information visualisation. *IEEE Trans. Vis. Comput. Graph.*, 13(6):1216–1223, 2007.

[17] L. C. Freeman, C. M. Webster, and D. M. Kirke. Exploring social structure using dynamic three-dimensional color images. *Social Networks*, 20(2):109–118, 1998.

[18] E. R. Gansner, Y. Hu, S. C. North, and C. E. Scheidegger. Multilevel agglomerative edge bundling for visualizing large graphs. In *IEEE Pacific Visualization Symp.*, pages 187–194, 2011.

[19] M. Ghoniem, J. Fekete, and P. Castagliola. A comparison of the readability of graphs using node-link and matrix-based representations. In *10th IEEE Symp. on Information Visualization*, pages 17–24, 2004.

[20] A. Goldenberg, A. X. Zheng, S. E. Fienberg, and E. M. Airoldi. A survey of statistical network models. *Foundations and Trends in Machine Learning*, 2(2):129–233, 2009.

[21] J.-D. J. Han, D. Dupuy, N. Bertin, M. E. Cusick, and M. Vidal. Effect of sampling on topology predictions of protein-protein interaction networks. *Nature Biotechnology*, 23(7):839–844, 2005.

[22] S. J. Hardiman and L. Katzir. Estimating clustering coefficients and size of social networks via random walk. In *22nd Intl. World Wide Web Conf.*, pages 539–550, 2013.

[23] Y. He, J. Jia, B. Yu, et al. Reversible mcmc on markov equivalence classes of sparse directed acyclic graphs. *The Annals of Statistics*, 41(4):1742–1779, 2013.

[24] D. Hennessey, D. Brooks, A. Fridman, and D. E. Breen. A simplification algorithm for visualizing the structure of complex graphs. In *12th Intl. Conf. on Information Visualisation*, pages 616–625, 2008.

[25] I. Herman, G. Melançon, and M. S. Marshall. Graph visualization and navigation in information visualization: A survey. *IEEE Trans. Vis. Comput. Graph.*, 6(1):24–43, 2000.

[26] D. Holten. Hierarchical edge bundles: Visualization of adjacency relations in hierarchical data. *IEEE Trans. Vis. Comput. Graph.*, 12(5):741–748, 2006.

[27] D. Holten and J. J. van Wijk. Force-directed edge bundling for graph visualization. *Comput. Graph. Forum*, 28(3):983–990, 2009.

[28] P. Hu and W. C. Lau. A survey and taxonomy of graph sampling. *CoRR*, abs/1308.5865, 2013.

[29] C. Hurter, O. Ersoy, and A. Telea. Graph bundling by kernel density estimation. *Comput. Graph. Forum*, 31(3):865–874, 2012.

[30] Y. Jia, J. Hoberock, M. Garland, and J. C. Hart. On the visualization of social and other scale-free networks. *IEEE Trans. Vis. Comput. Graph.*, 14(6):1285–1292, 2008.

[31] M. Kurant, A. Markopoulou, and P. Thiran. On the bias of BFS (breadth first search). In *22nd Intl. Teletraffic Congress*, pages 1–8, 2010.

[32] A. Lakhina, J. W. Byers, M. Crovella, and P. Xie. Sampling biases in IP topology measurements. In *Proc. IEEE INFOCOM 2003*, 2003.

[33] B. Lee, C. Plaisant, C. S. Parr, J. Fekete, and N. Henry. Task taxonomy for graph visualization. In *Proc. of the 2006 AVI Workshop on BEyond time and errors: novel evaluation methods for information visualization, BELIV '06*, pages 1–5, 2006.

[34] S. H. Lee, P.-J. Kim, and H. Jeong. Statistical properties of sampled networks. *Physical Review E*, 73(1):016102, 2006.

[35] J. Leskovec and C. Faloutsos. Sampling from large graphs. In *Proc. of the 20th ACM SIGKDD Intl. Conf. on Knowledge Discovery and Data Mining*, pages 631–636, 2006.

[36] J. Leskovec and E. Horvitz. Planetary-scale views on a large instant-messaging network. In *17th Intl. World Wide Web Conf.*, pages 915–924, 2008.

[37] J. Leskovec, J. M. Kleinberg, and C. Faloutsos. Graphs over time: densification laws, shrinking diameters and possible explanations. In *Proc. of the 11th ACM SIGKDD Intl. Conf. on Knowledge Discovery and Data Mining*, pages 177–187, 2005.

[38] R. Li, J. X. Yu, L. Qin, R. Mao, and T. Jin. On random walk based graph sampling. In *31st IEEE ICDE*, pages 927–938, 2015.

[39] S. Liu, W. Cui, Y. Wu, and M. Liu. A survey on information visualization: recent advances and challenges. *The Visual Computer*, 30(12):1373–1393, 2014.

[40] L. Lovász. Random walks on graphs: A survey. In D. Miklós, V. T. Sós, and T. Szőnyi, editors, *Combinatorics, Paul Erdős is Eighty*, volume 2, pages 353–398. János Bolyai Mathematical Society, Budapest, 1996.

[41] A. S. Maiya and T. Y. Berger-Wolf. Benefits of bias: towards better characterization of network sampling. In *Proc. of the 17th ACM SIGKDD Intl. Conf. on Knowledge Discovery and Data Mining*, pages 105–113, 2011.

[42] J. J. McAuley and J. Leskovec. Learning to discover social circles in ego networks. In *Advances in Neural Information Processing Systems 25*, pages 548–556, 2012.

[43] J. Moody. Peer influence groups: identifying dense clusters in large networks. *Social Networks*, 23(4):261–283, 2001.

[44] C. Muelder and K. Ma. A treemap based method for rapid layout of large graphs. In *IEEE Pacific Visualization Symp. 2008*, pages 231–238, 2008.

[45] D. Rafiei and S. Curial. Effectively visualizing large networks through sampling. In *16th IEEE Visualization Conf.*, pages 375–382, 2005.

[46] P. Sah, L. O. Singh, A. Clauset, and S. Bansal. Exploring community structure in biological networks with random graphs. *BMC Bioinformatics*, 15(1):1–14, 2014.

[47] M. P. Stumpf, C. Wiuf, and R. M. May. Subnets of scale-free networks are not scale-free: sampling properties of networks. *Proc. Natl. Acad. Sci. U. S. A.*, 102(12):4221–4224, 2005.

[48] C. Vehlow, T. Reinhardt, and D. Weiskopf. Visualizing fuzzy overlapping communities in networks. *IEEE Trans. Vis. Comput. Graph.*, 19(12):2486–2495, 2013.

[49] T. von Landesberger, A. Kuijper, T. Schreck, J. Kohlhammer, J. J. van Wijk, J. Fekete, and D. W. Fellner. Visual analysis of large graphs: State-of-the-art and future research challenges. *Comput. Graph. Forum*, 30(6):1719–1749, 2011.

[50] D. J. Watts and S. H. Strogatz. Collective dynamics of small-worldnetworks. *Nature*, 393(6684):440–442, 1998.

[51] Y. Wu, N. Pitipornvivat, J. Zhao, S. Yang, G. Huang, and H. Qu. egoslider: Visual analysis of egocentric network evolution. *IEEE Trans. Vis. Comput. Graph.*, 22(1):260–269, 2016.

[52] M. Zinsmaier, U. Brandes, O. Deussen, and H. Strobelt. Interactive level-of-detail rendering of large graphs. *IEEE Trans. Vis. Comput. Graph.*, 18(12):2486–2495, 2012.