Preserving Minority Structures in Graph Sampling

Ying Zhao, Haojin Jiang, Qi'an Chen, Yaqi Qin, Huixuan Xie, Yitao Wu
Shixia Liu, Zhiguang Zhou, Jiazhi Xia and Fangfang Zhou

Abstract—Sampling is a widely used graph reduction technique to accelerate graph computations and simplify graph visualizations. By comprehensively analyzing the literature on graph sampling, we assume that existing algorithms cannot effectively preserve minority structures that are rare and small in a graph but are very important in graph analysis. In this work, we initially conduct a pilot user study to investigate representative minority structures that are most appealing to human viewers. We then perform an experimental study to evaluate the performance of existing graph sampling algorithms regarding minority structure preservation. Results confirm our assumption and suggest key points for designing a new graph sampling approach named mino-centric graph sampling (MCGS). In this approach, a triangle-based algorithm and a cut-point-based algorithm are proposed to efficiently identify minority structures. A set of importance assessment criteria are designed to guide the preservation of important minority structures. Three optimization objectives are introduced into a greedy strategy to balance the preservation between minority and majority structures and suppress the generation of new minority structures. A series of experiments and case studies are conducted to evaluate the effectiveness of the proposed MCGS.

Index Terms—Graph sampling, graph visualization, node-link diagram

1 Introduction

Graphs contain plentiful structures that can be categorized from diverse perspectives, such as normal or abnormal [55] and central or periphery [8]. In this work, we categorize graph structures into majority and minority depending on their occurrence frequencies and sizes in a graph. Majority structures refer to those frequently occurring (e.g., frequent subgraphs) or large (e.g., communities). Minority structures are those rarely occurring and containing only a few nodes (e.g., extremely high degree nodes and bridges between communities). Both categories are important in graph analysis and are of great concern in various fields, such as community detection [79], frequent subgraph mining [89], spammers identification [55], and bridge vulnerability estimation [68].

Sampling is an efficient graph reduction technique [34,82,85,88]. Many graph sampling algorithms have been proposed to reduce graph sizes while preserving structures [27,46,72]. They are particularly useful in accelerating graph computations [28] and simplifying graph visualizations [58]. However, we assume that the existing algorithms tend to preserve majority structures but overlook minority structures, because the influence of majority structures on the representativeness of the original graph is considerable for measurable metrics and visual perception, whereas that of minority ones is negligible. For example, the algorithms tend to select the nodes with common degrees far more than the nodes with rare degrees to maintain the power law of degree distribution [66]. Human viewers prefer to observe large structures in advance but may ignore small structures when judging whether a sample is visually similar to the original graph [43,75].

To verify this hypothesis, we conducted a pilot user study and an experimental study. In the user study (Section 3), we recruited 20 participants and asked them to freely select structures of interest (SOIs) in 34 real-world graph data sets. The results showed that four representative types of minority structures, namely, super pivot, huge star, rim, and tie, elicited strong interest from the participants. Super pivots and huge stars are a small proportion of nodes with extremely high degrees. Rims are parachute- or chain-like structures attached to community margins. Ties are sparely distributed bridges at community boundaries. Figure 1(a) shows a toy case graph containing three super pivots, one huge star, three rims, and one tie.

In the experimental study (Section 4), we selected 12 real-world graph data sets and 20 reference graph sampling algorithms and designed three new quantitative indicators to evaluate their performance on preserving the four types of minority structures. The results showed that most of the algorithms cannot effectively preserve minority structures and may generate new minority structures that did not exist in the original graphs, especially for huge stars, rims, and ties. Figure 1(b−g) show the samples obtained by popular graph sampling algorithms. Most of the samples fail to preserve the huge stars, rims, and ties in Figure 1(a). New huge stars or new rims are found in Figure 1(c, d, e, and g). This situation is detrimental to graph analyses that focus on minority structures.

A new graph sampling algorithm oriented to minority structure analysis is needed, but its design is difficult. On the basis of results and experience in the experimental study, five key points should be seriously considered in the design: (1) identifying minority structures quickly and accurately; (2) avoiding losing minority structures in samples caused by the undersampling of their neighbors [75], such as the loss of the parachute-like rim in Figure 1(b, d, and e); (3) minimizing the inconsistency of preserved minority structures in random sampling; (4) balancing the preservation between minority and majority structures; and (5) suppressing the generation of new minority structures.

We propose a new graph sampling method called mino-centric graph sampling (MCGS) by considering the above points. This work stipulates graphs are simple, unattributed, undirected, and connected to simplify representations. First, we design a fast triangle-based algorithm to identify super pivots and huge stars and a fast cut-point-based algorithm to identify rims and ties. Second, we propose a set of importance assessment criteria to guide the preservation of minority structures and their neighbors and minimize the influence of random sampling. Finally, we introduce three optimization objectives into a greedy strategy to strike a balance between the preservation of minority and majority structures and suppress the generation of new minority structures. A series of experiments and case studies are conducted to evaluate the effectiveness of the proposed MCGS. The results reveal that MCGS performs the best among the 20 reference algorithms on the preservation of minority structures and suppression of new minority structure generation. MCGS also achieves highly satisfactory performance on the majority structure preservation.

In summary, this work presents the first attempt to investigate minority structure preservation in graph sampling. This work contributes: (1) four representative types of minority structures that are summarized through a controlled user study, (2) an experimental study that evaluates the

- *Ying Zhao, Haojin Jiang, Qi'an Chen, Yaqi Qin, Huixuan Xie, Yitao Wu, Jiazhi Xia, and Fangfang Zhou are with School of Computer Sciences and Engineering, Central South University, China. E-mail: {zhaoying, gallows, 204712126, 0921160110, 204711057}@csu.edu.cn, {wuyitao51, xiajiazhi, zff}@csu.edu.cn.*
- *Shixia Liu is with School of Software, Tsinghua University, China. E-mail: shixia@tsinghua.edu.cn*
- *Zhiguang Zhou is with School of Information, Zhejiang University of Finance and Economics, China. E-mail: zhgzhou1983@zufe.edu.cn.*
- *Jiazhi Xia and Fangfang Zhou are corresponding authors.*

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

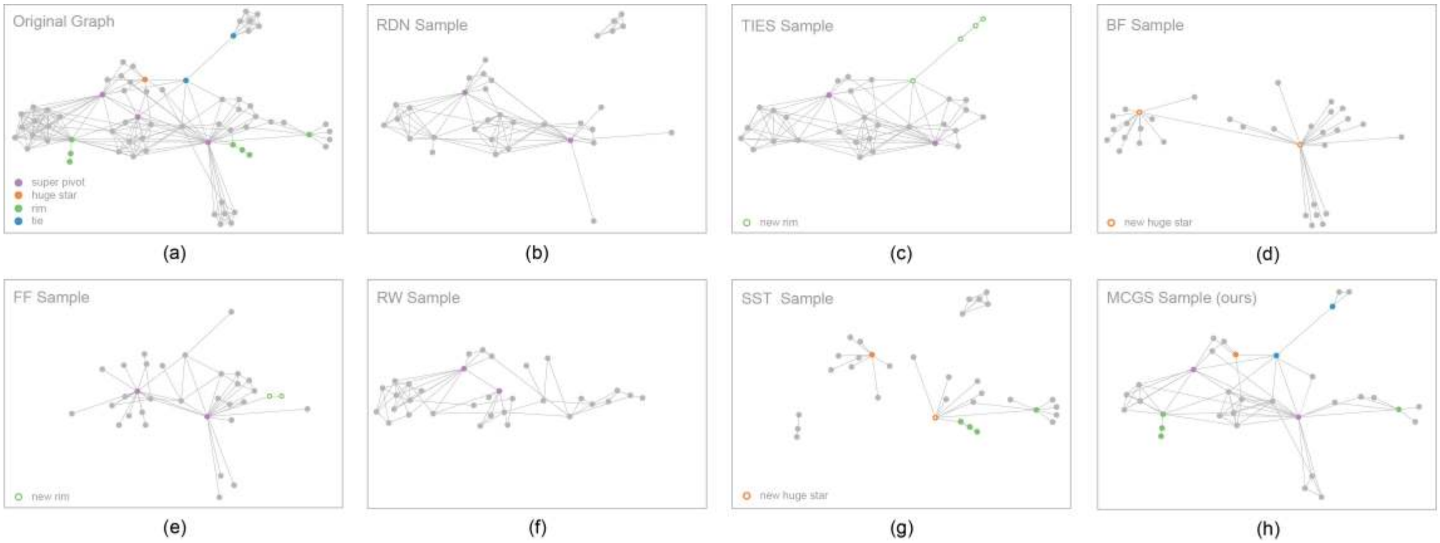

Figure 1. Illustration of four representative types of minority structures in a toy case graph (a) and seven graph samples obtained by RDN (b), TIES (c), BF (d), FF (e), RW (f), SST (g), and our proposed MCGS (h), respectively, with a sampling rate of 50%. The graph is slightly modified from the character relationship network of the novel *Les Misérables* [31]. The relative locations of nodes in a sample are consistent with those of the corresponding nodes in the original graph. Solid-color dots represent original minority structures. Hollow-color dots represent newly generated minority structures.

performance of existing graph sampling algorithms on minority structure preservation, and (3) a new graph sampling algorithm that is oriented to minority structure preservation.

2 Related Work

2.1 Graph Structure Analysis

Graph structures have been categorized and defined from diverse analytical perspectives. In this work, we categorize graph structures into minority and majority and define four types of minority structures (i.e., super pivot, huge star, rim, and tie). Our categorization and definition have certain connections with those in graph anomaly detection, community detection [37], structural role discovery [32], and frequent subgraph mining [89].

Graph structures are categorized into normal and abnormal in graph anomaly detection. Anomalies are nodes, edges, or subgraphs that differ from most ones [21, 55]. In general, node and edge anomalies are minority structures, such as spammers in email networks with extremely high degrees, because of low occurring probability and small size. However, subgraph anomalies with relatively large sizes are not minority structures.

Graph structures can be categorized into communities/clusters, hubs, and outliers to facilitate community detection [63]. Communities are majority structures [76]. Hubs correspond to ties in minority structures. Outliers refer to small structures attaching at margins of communities, such as secret leaders controlling a criminal gang through intermediaries [78], corresponding to chain-like rims in minority structures.

Structural role discovery assigns behavior roles, such as clique- or periphery-members, to structures [32, 33]. Cliques are not minority structures because of containing many densely connected nodes, but extremely high degree nodes in cliques are minority structures. Some periphery members at clique marginal areas, such as parachute- and chain-like nodes, are rims in minority structures.

In the fields of frequent subgraph mining [89], motifs discovery [18], and graphlet-based characterization [42, 69], graph structures are categorized depending on occurrence frequency. Most frequent subgraphs, motifs, and graphlets are majority structures. However, their variants could be minority structures. For example, large star-shaped motifs can be regarded as huge stars in minority structures but only the central high degree nodes of such motifs are included in our definition of huge stars. Moreover, many studies of attributed data analysis devoted to rare category detection [56] and classified data entities into majority and minority classes [86], which supports this work.

2.2 Graph Sampling Algorithms

Existing graph sampling algorithms can be classified into three groups, namely, node-, edge-, and traversal-based [34, 75]. In the node-based and edge-based groups, Random Node (RN), Random PageRank Node (RPN), and Random Degree Node (RDN) [46] are typical node-based algorithms. Random Edge (RE) [46] and Random Node Edge (RNE) [40] are classic edge-based algorithms. These algorithms are lightweight and provide theoretical references and building blocks for advanced algorithms. However, these algorithms have a common defect that randomly selected nodes are uncorrelated, thus causing unsatisfying preservation of graph connectivity [46]. Totally-Induced Edge Sampling (TIES) [3] is an edge-based approach that introduces a graph induction step into sampling. Additional edges in this step are retained to restore graph connectivity. We learn from this idea to reduce the generation of new minority structures.

The traversal-based group includes many algorithms. Their common merit is that connected graphs remain connected after sampling [15]. Breadth-First (BF) and Depth-First (DF) samplings are two basic approaches that select nodes in a breadth-first and depth-first traversal order, respectively [14, 15]. Snowball (SB) and Forest Fire (FF) are variants of BF [42] that expand limited neighbors instead of exhaustive expansion to ensure the picking of depth nodes [34, 46]. Random Walk (RW) and Random Jump (RJ) are variants of DF [30, 48] that allow walking to neighbors or random nodes during the traversal for preserving the global topology. Variants of RW include Multi-Dimensional Random Walk (MDRW) [61], Metropolis–Hastings Random Walk (MHRW) [67], Rejection-Controlled Metropolis-Hastings (RCMH) [26], and Generalized Maximum-Degree Random Walk (GMD) [49]. In our approach, a depth-first traversal and random-pick processing are adopted.

Some graph sampling approaches cannot be directly sorted into the above three groups. These approaches provide alternative solutions for specific goals. Random Areas Selection Sampling (RAS) selects an area of nodes each time to fully preserve their neighborhood structures [89]. We use this idea to preserve neighbors of minority structures. Distributed Learning Automata Sampling (DLAS) uses multiple automata for sampling [60]. Sampling with Shortest Paths (SSP) and Sampling with Spanning Trees (SST) identify important edges to guide sampling [36, 59]. Multiple Snowball with Cohen (RMSC) combines the advantages of RN and SB. Sampling based on Graph Partition (SGP) and Sampling based on Densification Power Law (DPL) partition a graph before sampling [16, 81]. We adopt this partition idea to deal with irregularly shaped graphs. Moreover, all the aforementioned algorithms perform graph sampling in the data space rather than the visual space to simultaneously support graph computation acceleration and graph visualization simplification. Our MCGS is also a data space approach.

2.3 Evaluation of Graph Sampling

Graph sampling algorithms have been thoroughly evaluated by two groups of metrics. The first group quantifies how well structural properties of the original graph are preserved [46, 61, 72]. Popular metrics are the Degree Distribution (DD) and Clustering Coefficient Distribution (CCD) [7, 45].

The second group measures the similarity between the original and sampled graphs. Two popular metrics in this group are the Jaccard Index (JI) that measures the similarity by the size of intersections [20] and the number of connected components (NCC) that measures the similarity of graph connectivity [64]. Recently, visual perception factors are considered in graph sampling evaluation. Wu et al. [75] found that three factors, namely, cluster quality, high degree nodes, and coverage area, influence the visual perception of sampled node-link diagrams. Quan et al. [54] studied proxy graphs to measure the shape-based faithfulness of sampled graphs.

At present, no metrics are tailored for evaluating the preservation of majority and minority structures. In this work, we use traditional metrics, including DD, JI, and NCC, to evaluate majority structure preservation because inherent connections exist between these metrics and structural properties or overall shapes of majority structures. For example, DD can measure the structural properties of frequent structures. JI can compare the overall shapes formed by large structures. Moreover, we design three new indicators to evaluate minority structure preservation. We also consider perception factors in our evaluation experiments [73, 77, 84, 87].

3 Pilot User Study

We assumed that existing graph sampling algorithms cannot effectively preserve minority structures. Before verifying this hypothesis, we conducted a pilot user study to answer two basic guiding questions: whether and which minority structures are important in graph analysis.

3.1 User Study Design

We recruited 20 participants (8 females and 12 males, all were graduate students aged 20-26 years) and selected 34 real-world graph data sets. The task was to select any SOIs in the graphs. The graphs were visualized in node-link diagrams as plain graphs. Graph layouts used the ForceAtlas2 algorithm [35]. The participants were asked to perform the task on the 34 graphs in a random order. For a graph, 60 s was allotted, and an interval of 2 s was set before proceeding to the next graph. After completing all graphs, the participants reviewed their selections and stated their thoughts. The study was conducted on a desktop with a 23.8-inch 1920 × 1080 LCD display, a standard keyboard, and a mouse. Descriptions of data sets and the experimental interface are provided in the supplementary material.

3.2 Result Analysis

The selected SOIs and selection sequence of each participant on each graph were recorded as the results. We manually categorized all SOIs into eight types and counted the entries for each type. If multiple SOIs of the same type were sequentially selected by a participant in a graph, then the count of the type was only 1 to avoid overcounting. We also counted the entries for each type in orders of 1st, 2nd, 3rd, and others in all sequences. Table 1 shows the statistical results of the eight SOI types.

Table 1. Statistics of eight SOI types for the participants in the user study

SOI Type	Number of Entries by Order				
	1st	2nd	3rd	Others	Total
HD-global	184	146	72	24	426
MS	103	124	103	51	381
BS	87	81	128	49	345
CS	22	65	58	84	229
FC	41	18	14	22	95
CS-overlapping	15	15	14	17	61
HD-local	4	4	20	4	32
IS	10	1	3	4	18

The result showed that four SOI types, namely, global high degree structure (HD-global), margin structure (MS), boundary structure (BS), and community structure (CS), had entries more than the average of 198 for all types. The remaining four SOI types, namely, small cliques far away from graph main bodies (FC), community overlapping structure (CS-overlapping), local high degree structure (HD-local), and isolated structure (IS), had entries far fewer than the average. Therefore, HD-global, MS, BS, and CS were considered as the most appealing structures in interactive graph explorations. Among them, HD-global, MS, and BS belonged to minority structures, whereas CS belonged to majority structures.

HD-global, MS, and BS must be further discussed. (1) HD-global ranked first. Most of the participants confirmed that the nodes with extremely high degrees had a strong visual saliency, which was in line with the previous research that stated that visually salient high degree nodes should not be lower than the global top 10% [57]. In general, high degree nodes have two subtypes [71], namely, pivot and star. A pivot is a high degree node whose neighbors have at least one interconnection. A star is a high degree node whose neighbors do not interconnect. We stipulated that our concerned pivots had degrees within the global top 5%, named as super pivots; our concerned stars had degrees above the global mean [62], named as huge stars. We used a strict threshold for pivots but a lax one for stars because stars were lower in quantity than pivots. (2) MS structures ranked second. They were appendage nodes that occasionally occurred in the marginal areas of communities and formed specific visual shapes, such as shapes like parachute, chain, balloon, or tree. A large proportion of the participants reported that parachute- and chain-like structures at community rims were especially eye-catching. Thus, we used rims as a concrete representative of MS structures with the two visual shapes. (3) BS structures ranked third and were a sequence of nodes bridging any two communities. Many of the participants commented that the structures that tied communities were sometimes more attractive than communities. Thus, we used ties as a concrete representative of BS structures in this work.

As a result, we obtained four representative types of minority structures (i.e., super pivot, huge star, rim, and tie). These structures were visually salient in node-link diagrams and elicited strong interest from the participants in the pilot user study. A literature review (Section 2.1) confirmed that these structures were also important in various research and application branches.

4 Experimental Study

We conducted a controlled experiment to examine the performance of existing graph sampling algorithms in preserving the four representative types of minority structures.

4.1 Hypotheses

To guide the experiment, we formulated three specific hypotheses:

H1: Existing algorithms pursue graph similarity and have a low ability to preserve minority structures. At present, graph similarity is largely measured by the structural properties and overall shapes of majority structures (Section 2.3). Thus, we assume that existing algorithms naturally have a relatively low ability for minority structure preservation.

H2: Existing algorithms may produce new minority structures. Many nodes in samples inevitably have incomplete original neighbors, thereby causing that some structures may degenerate to minority structures that do not exist in the original graph.

H3: Existing algorithms cannot guarantee the preservation of important minority structures. Only a part of minority structures in a graph are crucially important according to certain criteria (Section 5.4.2). The randomness of graph sampling has a fatal influence on minority structure preservation. Slight differences in selecting nodes may lead to the disappearance of minority structures. Thus, we assume that important minority structures will disappear or be no longer important in samples.

4.2 Experimental Study Design

Data preparation. We selected 12 real-world graph data sets as the experiment data, two of which were for pilot tests to determine the design of the formal experiment. The graphs were mainly social, web, and communication networks popular in graph studies and included seven small, three medium, and two large scales. Data processing was conducted to detect and label the type and importance of each minority structure in each graph by using the methods introduced in Sections 5.4.1 and 5.4.2.

Reference algorithms and parameter settings. We selected 20 graph sampling algorithms as references. These algorithms had two common parameters, namely, initial seed and sampling rate. To reduce the influence of parameter settings, we prepared four types of seeds [62] (i.e., random, high degree, high betweenness, and peripheral nodes) and four sampling rates (i.e., 10%, 20%, 30%, and 40%). Other distinctive parameters were set with defaults. Detailed data descriptions and parameter setting considerations are provided in the supplementary material.

Experimental Procedure. Each algorithm ran 800 trials (10 graphs × 4 types of seeds × 4 different sampling rates × 5 runs). 16,000 samples in total were obtained as the raw results. The experiment was conducted on a desktop with a 3.4 GHz Intel i7 CPU and 16 GB of RAM.

4.3 Indicator Design

Given the lack of established indicators, we designed three quantitative indicators to verify the three hypotheses (Section 4.1) and measure the performance of minority structure preservation.

Minority structure preservation rate (MSPR). This indicator is the ratio of the preservation rate of minority structures to a sampling rate (H1), in which the preservation rate refers to the proportion of original minority structures preserved in a sample. For example, given a sample with a sampling rate of 30%, the MSPR is 1 when 3 out of 10 minority structures in the original graph are preserved. Generally, MSPR is related to a certain type of minority structure and is defined as

$$MSPR = \frac{\mid MS_{S_x} \cap MS_x \mid / \mid MS_x \mid}{\Phi},$$

where MS_x represents the set of minority structures of x type in a graph G; MS_{S_x} represents the set of minority structures of x type in a sample G_s; Φ is the sampling rate; and $\mid MS_x \mid$ and $\mid MS_{S_x} \mid$ are the cardinalities of MS_x and MS_{S_x} respectively. An MSPR approaching, equal to, or even greater than 1 means a "perfect" result.

New minority structure generation rate (MSGR). This indicator represents the probability that new minority structures of a certain type occur in a sample (H2). An MSGR approaching or equaling 0 is a "perfect" result. MSGR is formulated as

$$MSGR = \mid (MS_{S_x} - MS_x) \mid / \mid MS_{S_x} \mid .$$

Mean importance precision (MIP). This indicator evaluates the mean preservation precision of top K minority structures of a certain type before and after sampling (H3). MIP is a variant of average precision in recommendation system ranking [74]. We define MIP as

$$MIP = \frac{\sum_{i=1}^{K} \mid Top_i(MS_x) \cap Top_i(MS_{S_x}) \mid / i}{K},$$

where $Top_i(.)$ denotes the top i important minority structures. For example, the top five super pivots in a graph are nodes ID-11, 12, 26, 9, and 15; and those in a sample are nodes ID-11, 18, 26, 12, and 15. Four super pivots at the 1st, 3rd, 4th, and 5th positions in the sample are matched. Therefore, MIP is 0.743 [(1/1+1/2+2/3+3/4+4/5)/5]. An MIP approaching or equal to 1 is a "perfect" result.

4.4 Result Analysis

Result processing. The result processing consisted of three parts. (1) We calculated the values of the three indicators for the four types of minority structures on each sample. (2) We calculated the medians and standard deviations of each indicator on each minority structure type and algorithm. (3) We selected empirical thresholds based on two criteria: indicating good sampling results and differentiating the performance of the reference algorithms. We stipulated that MSPR $\geq$ 0.9 was good results, indicating that the preservation rate of minority structures was approximately equal to the sampling rate. Empirically, this threshold represents an ideal balance for the preservation of minority and majority structures. MSGR $\leq$ 0.5 denoted good results, representing that new minority structures were no more in quantity than the preserved original ones. Thus, original minority structures can still be dominant in the sample and analyzed without distinct interference. MIP $\geq$ 0.5 was good, implying that more than half of the top K important minority structures were preserved.

We verified the hypotheses based on the processed results (Table 2). Additional results are provided in the supplementary material.

Hypothesis verification. H1 was partially confirmed. The MSPR results reflected that these algorithms generally had low ability in preserving minority structures but a few algorithms can effectively preserve a certain minority structure type. For the four types of minority structures, the MSPR results of super pivots (μ = 0.6733) and huge stars (μ = 0.4387) were better than those of rims (μ = 0.2835) and ties (μ = 0.1199). The reason was that the former two were commonly embedded in communities with relatively stable neighborhood structures, whereas the latter two were located at the margin or boundary areas of communities with sparse and unstable neighborhoods. From a single-algorithm perspective, RDN, RPN, DPL, and RAS performed well (MSPR $\geq$ 0.9) in super pivots because they were in favor of high degree nodes [26,46,59]. TIES also performed well in super pivots because of its graph induction step [3]. SST performed well in rims because the use of spanning trees maintained peripheral nodes [36].

H2 was partially confirmed. The MSGR results reflected that these algorithms can effectively suppress the generation of new super pivots (μ = 0.3212) but hardly inhibited the generation of new huge stars (μ = 0.6304), rims (μ = 0.8231), and ties (μ = 0.6947). Many algorithms performed well (MSGR $\leq$ 0.5) in super pivots because other structures rarely degenerated to pivots after sampling. However, pivots may degenerate to stars when all edges between neighbors were lost. Thus, only seven algorithms performed well in huge stars. Furthermore, peripheral nodes and CS-overlapping structures had chances to degenerate to rims and ties, respectively. RMSC was the only algorithm that effectively suppressed the generation of new ties. Its breadth-first and multi-snowball strategy effectively preserved the connections between communities [26].

H3 was fully confirmed. Most of the MIP medians in Table 2 were poor (lower than 0.5), indicating that no algorithms can guarantee the preservation of important minority structures for two reasons. First, originally important minority structures were not well preserved. Second, newly generated minority structures became important in samples.

Other findings. The graph data sets, sampling rates, and initial seeds affected minority structure preservation to some extent. (1) Data sets. A graph with a single cluster or multiple clusters in approximate sizes is called a balanced graph. A graph with multiple clusters that present a wide difference in size is called an unbalanced graph. We found that the results on unbalanced graphs were worse than those on balanced graphs, because small clusters that contained important ties or chain-like rims in unbalanced graphs were not effectively maintained. (2) Sampling rates. The scores of the three indicators generally improved when the sampling rate increased because high sampling rates resulted in highly completed structures. (3) Initial seeds. The high-degree type of initial seeds generally performed better than the other three types of initial seeds because the former provided numerous available paths for sampling.

5 New Algorithm Proposal

The results of the experimental study confirm that the existing algorithms can hardly preserve minority structures. In this section, we introduce a new algorithm called MCGS.

5.1 Definitions and Notations

A *graph* is notated with $G = (V,E)$, where $V = \{v_1, v_2, ..., v_n\}$ represents nodes, $E = \{e_1, e_2, ..., e_m\}$ represents edges, and an edge $e = (v_i, v_j)$ connects nodes v_i and v_j. This work focuses on scale-free graphs [2] and stipulates that graphs are simple, unattributed, undirected, and connected to simplify the representations.

A *graph sample* is notated with $G_s = (V_s, E_s)$, where V_s is a subset of nodes ($V_s \subset V$) and $E_s = (V_s \times V_s) \cap E$. Considering a node-based sampling strategy, a *sampling rate* is defined as $\Phi = \mid V_s \mid / \mid V \mid$, where $\mid V_s \mid$ is the cardinality of V_s and $\mid V \mid$ is the cardinality of V. E_s is obtained from the induced subgraph of G based on V_s.

We use $\Omega = \{P,S,R,T\}$ to represent the four types of minority structures in G, where P represents the set of all super pivots, notated by $P = \{p_1, \ldots, p_l\}$, whereas S, R, and T represent the sets of all huge stars, rims, and ties, respectively. The four types are defined as follows:

A super pivot is a node whose one-step neighbor nodes have at least one interconnection, with its degree within the global top 5% as represented by μ. Super pivot is notated as $\forall p_i \in P$: (1) $\exists! v \in V_{p_i}$: $\mid \Gamma(v) \mid \geq \mu$, where V_{p_i} is the node set of p_i, $\Gamma(v)$ is the set of one-step neighbors of v, $\mid \Gamma(v) \mid$ is the cardinality of $\Gamma(v)$; and (2) $\exists x \in \Gamma(v)$: $\mid \Gamma(x) \cap \Gamma(v) \mid \geq 1$.

A huge star is a node whose one-step neighbor nodes are not connected to one another, with its degree above the global mean as represented by ε. Huge star is notated as $\forall s_i \in S$: (1) $\exists! v \in V_{s_i}$: $\mid \Gamma(v) \mid \geq \varepsilon$; (2) $\forall x \in \Gamma(v)$: $\Gamma(x) \cap \Gamma(v) = \emptyset$.

A rim is a node appearing at community margins and forming a parachute-like visual shape with its one-step neighbors or a sequence of

Table 2. Results of the experiments in Sections 4, 6.1.1, and 6.1.2 in terms of the medians of indicators (columns) and algorithms (rows). *P*, *S*, *R*, and *T* represent super pivot, huge star, rim, and tie, respectively. Blue indicates the winners in significance tests among algorithms. Bold indicates that the value is better than the empirical "good" indicator threshold (only for MSPR, MSGR, and MIP). Using the first column as an example, significant differences are found among algorithms on MSPR and super pivot. RDN, RPN, TIES, and MCGS are the winners. Six algorithms obtain "good" results of MSPR on super pivot.

Sampling Strategy	Algorithm	Indicators for Minority Structure Preservation												Indicators for Majority Structure Preservation			
		MSPR				MSGR				MIP				KSD	SDD	RCC	JI
		P	*S*	*R*	*T*	*P*	*S*	*R*	*T*	*P*	*S*	*R*	*T*				
Node-based	RN	0.796	0.453	0.098	0.000	**0.238**	0.723	0.924	1.000	0.000	0.000	0.000	0.000	0.495	0.080	0.006	0.066
	RDN	**0.997**	0.015	0.000	0.000	**0.000**	**0.000**	1.000	**0.479**	0.457	0.000	0.000	0.000	0.224	0.003	0.167	0.152
	RPN	**0.994**	0.256	0.017	0.000	**0.000**	**0.000**	0.948	0.854	0.457	0.000	0.000	0.000	0.222	0.003	0.067	0.128
Edge-based	RNE	0.704	0.346	0.135	0.000	**0.295**	1.000	0.851	1.000	0.000	0.000	0.000	0.000	0.353	0.006	1.000	0.057
	TIES	**0.994**	0.172	0.083	0.000	**0.000**	**0.000**	0.873	0.889	0.457	0.000	0.000	0.000	0.182	0.003	0.333	0.144
Traversal-based	BF	0.418	0.000	0.007	0.000	0.580	1.000	0.998	1.000	0.000	0.000	0.000	0.000	0.713	0.012	1.000	0.036
	FF	0.691	0.199	0.044	0.000	**0.308**	0.938	0.985	0.729	0.000	0.000	0.000	0.000	0.424	0.003	1.000	0.050
	GMD	0.705	0.350	0.131	0.000	**0.293**	1.000	0.851	1.000	0.000	0.000	0.000	0.000	0.354	0.006	1.000	0.056
	MDRW	0.700	0.350	0.113	0.000	**0.303**	1.000	0.875	1.000	0.000	0.000	0.000	0.000	0.373	0.006	1.000	0.051
	MHRW	0.264	0.035	0.092	0.000	0.750	0.948	0.945	1.000	0.000	0.000	0.000	0.000	0.662	0.014	0.015	0.046
	RCMH	0.496	0.603	0.174	0.000	0.526	0.991	0.818	0.963	0.000	0.000	0.000	0.000	0.391	0.006	1.000	0.059
	RJ	0.677	0.377	0.119	0.000	**0.316**	1.000	0.903	1.000	0.000	0.000	0.000	0.000	0.351	0.006	1.000	0.057
	RW	0.713	0.355	0.125	0.000	**0.286**	1.000	0.857	1.000	0.000	0.000	0.000	0.000	0.363	0.006	1.000	0.056
Others	DLAS	0.734	0.421	0.093	0.000	**0.265**	**0.337**	0.861	0.952	0.000	0.000	0.000	0.000	0.359	0.004	0.017	0.114
	DPL	**0.946**	0.297	0.000	0.000	**0.019**	**0.250**	1.000	1.000	0.000	0.000	0.000	0.000	0.254	0.004	0.083	0.098
	RAS	**0.962**	0.113	0.131	0.000	**0.000**	**0.000**	0.611	0.633	0.457	0.000	0.000	0.000	0.181	0.003	0.500	0.157
	RMSC	0.432	0.000	0.000	0.000	0.548	0.989	0.945	**0.000**	0.000	0.000	0.000	0.000	0.989	0.017	0.003	0.000
	SGP	0.877	0.000	0.034	0.000	**0.125**	**0.425**	0.891	1.000	0.000	0.000	0.000	0.000	0.305	0.004	0.019	0.098
	SSP	0.697	0.580	0.253	0.000	**0.293**	1.000	0.949	1.000	0.000	0.000	0.000	0.000	0.481	0.007	0.069	0.050
	SST	0.588	0.696	**0.918**	0.167	**0.400**	0.777	0.624	0.875	0.000	0.000	0.000	0.000	0.419	0.005	0.333	0.144
	MCGS (Ours)	**1.000**	**1.003**	**1.013**	**1.837**	**0.000**	**0.000**	**0.206**	**0.341**	**0.910**	**0.803**	0.497	**0.870**	0.170	0.003	1.000	0.209

nodes forming a chain-like visual shape, notated as $\forall r_i \in R$: (1) $V_{r_i} \subset V_{c_i}$, where $C = \{c_1, \ldots, c_n\}$ is the set of all disjoint communities created from G, and V_{c_i} is the node set of c_i; (2) $\exists v \in V_{r_i}$: $v \in V_{cut} \wedge \mid \Gamma(v) \mid \geq 2$, where V_{cut} is the set of cut points of G; and (3) $\forall x \in V_{r_i} - V_{cut}$: $\mid \Gamma(x) \mid = 1 \wedge \Gamma(x) \subset V_{r_i}$.

A tie is a sequence of nodes that bridge any two communities and form a chain-like visual shape, notated as $\forall t_i \in T$: (1) $V_{t_i} \subset V_{cut}$; (2) $\exists! u, v \in V_{t_i}$: $u \in V_{c_j} \wedge v \in V_{c_k} \wedge j \neq k$; and (3) $\forall x \in V_{t_i} - \{u, v\}$: $\mid \Gamma(x) \mid = 2 \wedge \Gamma(x) \subset V_{t_i}$.

5.2 Design Considerations

On the basis of the experience and results of the experimental study, we formulate six key points to be considered in the algorithm design.

C1: Identifying minority structures. Effectively preserving minority structures through global random sampling is difficult because minority structures only involve a small proportion of nodes. A feasible way is to identify and maintain them in advance.

C2: Preserving important minority structures. Prioritizing important minority structures is necessary for two reasons, that is, reserving space in a sample for subsequent majority structure sampling and minimizing the influence of random sampling on minority structure preservation.

C3: Preserving neighbors of minority structures. Simply selecting the self-nodes of minority structures is insufficient because they are no longer minority structures if losing neighbors.

C4: Balancing minority and majority structures. The absence of majority structures invalidates the sample because minority structures coexist with majority structures.

C5: Suppressing the generation of new minority structures. Existing algorithms produce new minority structures, possibly leading to a misjudgment on the graph by sample analysis.

C6: Improving robustness and scalability. For robustness, the influence of parameter settings and graph data sets on sampling should be minimized. For scalability, sampling should be completed within an acceptable time on large-scale graphs.

5.3 Algorithm Pipeline

The proposed MCGS algorithm consists of four steps, as shown in Figure 2.

STEP1. Minority structure identification. Given a graph G, a sample G_s, and a sampling rate Φ, we identify minority structures in G by using two newly designed algorithms (Section 5.4.1). This step outputs the four sets of minority structures $\{P, S, R, T\}$ that contain all super pivots, huge stars, rims, and ties in G, respectively.

STEP2. Minority structure ranking. We initially rank the four sets of minority structures separately in descending order of importance by using our proposed importance assessment criteria (Section 5.4.2) and quick sort. Then, we select the most important ones in each of the four sets based on Φ/α, where α is a constant that controls the quantity of preserved minority structures and is set to 1 by default. Specifically, a smaller α indicates more minority structures to be preserved. For example, given a graph with 3 ordered rims and $\Phi = 50\%$, we pick the top two $\lceil 3 \times (0.5/1) \rceil$ important ones. This step outputs $\{P_{im}, S_{im}, R_{im}, T_{im}\}$.

STEP3. Minority structure sampling. We directly put all nodes in $\{P_{im}, S_{im}, R_{im}, T_{im}\}$ into G_s and then randomly select a proportion of their one-step neighbor nodes into G_s by using an improved RAS sampling (Section 5.4.3). This step outputs an incomplete G_s that contain nodes of all important minority structures and parts of their neighbors.

STEP4. Majority structure sampling. We propose a greedy strategy to select the nodes in G to G_s to maximize the similarity between G_s and G. After reaching the sampling rate, we preserve all edges in the induced subgraph from G based on G_s to suppress the generation of new minority structures (see Section 5.4.4). This step outputs the completed G_s.

We also provide an optional additional step, that is, unbalanced graph processing. If G is examined as an unbalanced graph, then G will be divided into several subgraphs, and the above sampling process will be conducted on each of them. This step can reduce the influence of unbalanced graphs on the minority structure preservation (C6). This step is optional because a majority of graphs are balanced. We use a method based on the gradient boosting decision tree [24, 50] and a multilevel partitioning method [6] in this step. Supporting information for this step is provided in the supplementary material.

5.4 Algorithm Components

5.4.1 Minority Structure Identification

STEP 1 is to identify minority structures in G (C1). We propose two fast identification algorithms because straightforward methods are generally time-consuming (C6).

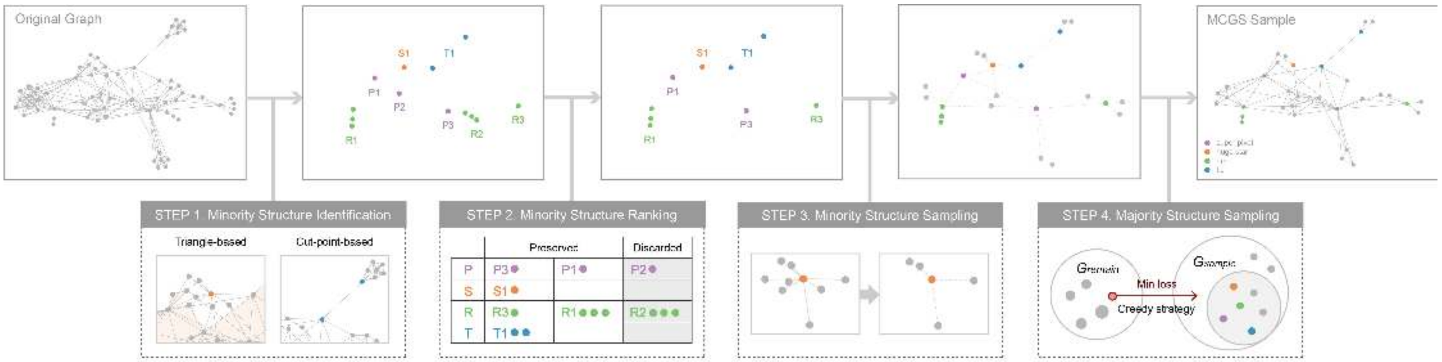

Figure 2. Four-step pipeline of MCGS: (1) identifying minority structures by using a triangle-based algorithm and a cut-point-based algorithm; (2) ranking minority structures based on importance assessment criteria; (3) preserving important minority structures and randomly selecting their neighbors into the sample according to the sampling rate (50% in this case); (4) selecting other nodes into the sample based on a greedy optimization strategy and outputting the induced subgraph as the completed sample.

A straightforward method of identifying pivots and stars is to seek out high degree nodes and check whether edges exist between its neighbors. If no edge exists, then it is a star; otherwise, it is a pivot. This method is time-consuming with a time complexity of $O(n^2)$, where n is the number of nodes in G. We design a triangle-based algorithm inspired by two ideas: a high degree node is either a pivot or a star, facilitating a simultaneous detection of pivot and star; a node whose relationships with any two of its neighbors form a triangle cannot be a star, accelerating the process.

This algorithm consists of five steps. (1) Given a graph, a DF traversal starts from any node. (2) For a visiting node, we check whether it forms a triangle with its predecessor and the node preceding the predecessor. If so, then we mark all the three nodes, such as the nodes marked with hollow dots in Figure 3(a-1) and 3(a-2). (3) After the traversal, unmarked high degree nodes are identified as stars, such as the nodes e and f in Figure 3(a-3). (4) We identify high degree nodes in G but not in the set of unmarked nodes as pivots. (5) We extract pivots with degrees within the global top 5% as super pivots and stars with degrees above the global mean as huge stars, such as the huge star e in Figure 3(a-4). The time consumption of this algorithm mainly arises from the DF traversal with a complexity of $O(n+m)$, where n and m are the numbers of nodes and edges in G, respectively.

A straightforward method to identify rims and ties is to use community information. However, community detection methods are time-consuming and complicated in parameter tuning. For example, it is difficult to determine the number of communities, which directly influences the identification of rims and ties (C6). We propose to use cut-point information because both a rim and tie have at least one cut point. A cut point is a node whose removal will cause the relevant connected subgraph to be disconnected. Figure 3(b-1) highlights all cut points in a graph.

The cut-point-based algorithm has four steps. (1) Given a graph, we obtain all cut points by DF traversal [23] to generate an induced subgraph denoted as $G_{cut} = (V_{cut}, E_{cut})$. (2) We merge each connected component of G_{cut} into a hyper node, as shown in Figure 3(b-2) and 3(b-3). (3) We identify a hyper node that contains only one original node as a parachute-like rim. (4) A hyper node point that contains multiple original nodes is a chain structure. If any end node of the chain has one and only one neighbor with degree 1 in G, then all nodes in the chain together with the neighbor are identified as a chain-like rim; otherwise, all nodes in the chain are regarded as a tie. The time complexity of this algorithm is $O(n+m)$. In addition, large parachute-like rims may be identified as super pivots or huge stars. This case rarely appears because the degree of rim is generally not high. In our work, such large rims are counted as not only rims but also super pivots or huge stars.

5.4.2 Minority Structure Ranking

Essential to STEP 2 is to determine importance assessment criteria for each minority structure type (C2). The results of the pilot user study reflect that the degree or size of a minority structure is directly related to its visual importance. We adopt this empirical result, which is simple and efficient. For a super pivot or a huge star, we stipulate its importance proportional to its degree. The importance of a parachute-like rim is proportional to the number of its neighbors with a degree of 1. For chain-like rims, a long chain is important. For a tie, we consider two factors, namely, the chain length and number of neighbors connecting to both ends of the chain.

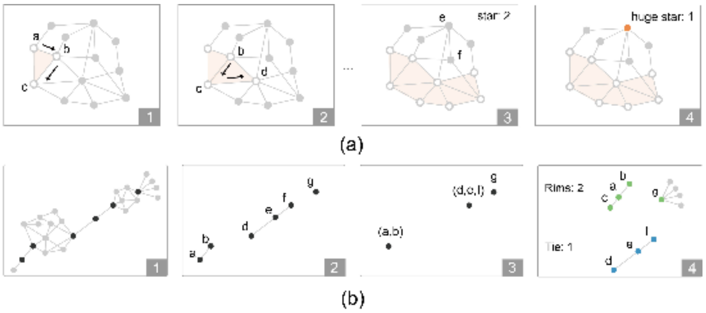

Figure 3. Illustration of triangle-based super pivot and huge star identification (a) and cut-point-based rim and tie identification (b).

5.4.3 Minority Structure Sampling

STEP 3 is to preserve the neighborhood structures of important minority structures (C3). RAS provides an efficient and effective way that directly maintains all one-step neighbors [89]. However, this way may include extensive nodes in a sample, thereby causing an early reaching of the sampling rate (C4). We propose to add a specific condition to RAS sampling. The amount of preserved neighbors for a minority structure should satisfy the condition $\mid \Gamma_s(ms_i) \mid = \mid \Gamma(ms_i) \mid * \Phi / \beta$, where ms_i denotes the key nodes of the minority structure i, $\Gamma(ms_i)$ is the set of neighbors of ms_i, $\Gamma_s(ms_i)$ is the set of preserved neighbors of ms_i. For a super pivot, huge star, or parachute-like rim, ms_i includes only one node; whereas for a chain-like rim or tie, ms_i includes the end nodes of the chain. β is a constant that controls the quantity of preserved neighbors; it is set to 2 by default to reserve half of the sample space for majority structure preservation (STEP4). Using α and β altogether can tune the preserving ratio of minority structures versus majority structures.

5.4.4 Majority Structure Sampling

An incomplete sample G_s that contains all important minority structures and parts of their neighbor nodes is obtained. STEP 4 aims to further add nodes and edges from G to G_s, making the completed sample G_s as similar to G as possible (C4). This situation can be described as an optimization problem. Given a $G = (V, E)$, an incomplete sample $G_s = (V_s, E_s)$, and a sampling rate Φ, an optimal node set V_{op} and an edge set E_{op} are added to G_s to ensure that the completed sample G_s can effectively represent G, where $V_{op} \subset \{V - V_s\}$, $E_{op} \subset \{E - E_s\}$, and $\mid V_{op} \mid = \mid V \mid * \Phi - \mid V_s \mid$. The objective function is as follows:

$$G_s^* \leftarrow \arg\min_{G_s \subset G} Deviance(G, G_s),$$

$$\text{s.t.} \mid V_s \mid = \mid V \mid \times \Phi,$$

This problem is NP-hard. Classical optimization algorithms, such as genetic algorithm [53] and simulated annealing [39], are candidates for problem solving but commonly have high computational consumptions and complicated parameter settings (C6). We adopt a greedy strategy that has no parameters, a high speed, and desired effects. The strategy consists of four steps. (1) For each node in $\{V - V_s\}$, we suppose to add it to V_s and then calculate the deviance of G and G_s (the induced subgraph of G based on V_s), called *loss*. (2) We find the minimum loss obtained in (1) and add the corresponding node into V_s formally. (3) We repeat (1) and (2) until $\mid V_s \mid$ reaches $\mid V \mid \times \Phi$. (4) We output the induced subgraph of G based on V_s as the completed sample G_s. The complexity of the strategy is $O(\mid V - V_s \mid \times (n+m))$.

The induction step is important. It can repair the neighborhood structures of sampled nodes and make the incomplete structures close to that of the original graph [3]. As a result, the neighborhoods of potential new stars, rims and ties can be repaired, thereby effectively suppressing the generation of new minority structures (C5). We are inspired by RDN, RPN, and TIES that obtained the distinguished performance on the MSGR indicator in the experimental study because they adopted an induction step.

The definition of the loss function is critical in the greedy strategy. The ultimate goal of majority structure sampling is to make G_s as similar to G as possible. Such similarities are commonly measured by the metrics mentioned in Section 2.3. We reference three popular metrics, namely, DD [34], NCC [51], and JI [20], to propose three objectives as follows.

(1) We use the mean square error (MSE) of degrees between G_s and G to depict the similarity of DD, notated as:

$$MSE = \frac{1}{\mid V_s \mid}\sum_{i=1}^{|V_s|}(\mid \Gamma(G,v_i) \mid - \mid \Gamma(G_s,v_i) \mid)^2,$$

where $\Gamma(G,v_i)$ is the set of neighbors of v_i in G

(2) We use NCC(G_s) to represent the number of connected components of G_s, which can measure the similarity of connectivity between G_s and G because G is supposed to be connected in this work, thus NCC(G)=1, notated as:

$$NCC(G_s) = \mid V_s \mid - \textstyle\sum_{i\in[\ |V_s|]} \mathrm{II}[\ \sigma_i(L) > 0],$$

where L is the Laplacian matrix of G, $L \in R^{n\times n}$, $\sigma_i(L)'s$ are the singular values of L.

(3) We use the Jaccard Index (JI) to measure the structural similarity between G_s and G, notated as:

$$JI(G,G_s) = \frac{1}{\mid V_s \mid}\sum_{v_i \in V_s}\frac{\mid \Gamma(G,v_i) \cap \Gamma(G_s,v_i) \mid}{\mid \Gamma(G,v_i) \cup \Gamma(G_s,v_i) \mid}.$$

As a result, our loss function is defined as:

$$loss = \omega_1 * Scaler(MSE) + \omega_2 * Scaler(NCC(G_s)) + \omega_3 * Scaler(JI(G,G_s)),$$

where ω_i is a weight coefficient,$\omega_i \in [\,0,1]$, $\sum_{i=1}^{3}\omega_i = 1$; and *Scaler* is a normalization processing to reduce the influence of the magnitude difference among the three objective functions.

The computation of the loss function could be accelerated (C6). We could only consider the incremental information of adding a new node into G_s each time when calculating MSE. We could use the union-find algorithm [25] to immediately obtain the number of disjoint sets of a graph for NCC. We could adjust the three weight coefficients on demand to involve only one or two objectives in computation. We set them with [1:0:0] by default. Empirically, no single sampling method can simultaneously fulfill the optimal effects on the three objectives [13, 17].

6 Evaluation

We evaluated the proposed MCGS algorithm through an objective performance analysis, a subjective assessment, and case studies.

6.1 Objective Performance Analysis

The performance analysis had three experiments with different indicators. The data, reference algorithms, execution conditions, result processing, and apparatus of the experiments were consistent with those of the experiment introduced in Section 4.

6.1.1 Minority Structure Preservation Performance

Indicators in this experiment were MSPR, MSGR, and MIP, which can evaluate the performance of minority structure preservation (Section 4.3). The results of MCGS are shown in the last row of Table 2. We conducted 12 groups of significance tests (3 indicators × 4 minority structure types) for MCGS and the 20 references algorithms. We initially used Shapiro-Wilk tests for each group to examine the normality of the experimental results of each algorithm on the 10 graphs and four sampling rates. The examination results did not follow the normal distribution. Then, we used a non-parametric Friedman test for each group. Significant differences ($p < 0.05$) were found in all the 12 groups. Finally, we used a Dunn–Bonferroni test to identify the winners in each group.

The results show that MCGS won 12 times in the significance tests and performed best 11 times in terms of indicator medians. MCGS underperformed RMSC on the MSGR medians of ties because of the superiority of RMSC in suppressing new ties. MCGS did not obtain a “good” MIP median on rims because MCGS could not completely avoid the generation of new minority structures, and new rims were most apt to be produced among the four types. In summary, MCGS overall performed best among the 20 references. It could effectively preserve the four types of minority structures, suppress the generation of new minority structures, and prevent the loss of important minority structures.

6.1.2 Majority Structure Preservation Performance

Indicators in this experiment were Kolmogorov–Smirnov distance (KSD) [52], skew divergence distance (SDD) [44], reciprocal of NCC (RCC) [64], and JI [20]. They are commonly used in graph sampling evaluations (Section 2.3). KSD and SDD measure the difference of degree distributions between a graph and sample, RCC measures the connectivity differences before and after sampling, and JI measures the similarity of graphs. The four indicators range from 0 to 1. Small values for KSD and SDD and large values for RCC and JI are good. The experimental results are shown in the four rightmost columns of Table 2. Notably, we tested significant differences but did not provide empirical “good” thresholds in the results.

Significant differences were found in the four indicators among the 21 algorithms. MCGS became the winners of KSD, RCC, and JI, indicating that MCGS did not pursue minority structure preservation at the expense of majority structure preservation. For SDD, MCGS obtained a satisfying median. Thus, the performance of MCGS in preserving majority structures was fairly satisfactory.

6.1.3 Time Performance

The indicator in this experiment was the mean time consumption of 20 samplings (4 seed types × 5 runs) performed by an algorithm on a graph under a sampling rate. We tested 21 algorithms, 10 graphs, and four sampling rates. Due to page limit, we only show the results of MCGS and seven reference algorithms under a sampling rate of 30% on four graphs (Table 3). More results are provided in the supplementary material.

Table 3. Time consumptions of MCGS and seven reference algorithms on four graph data sets.

Algorithm	**Time Consumptions on Graph Data Sets (Sec.)**			
	Facebook1912 *node:747* *edge:30025*	**Facebook107** *node:1034* *edge:26749*	**AS-733** *node:6474* *edge:13895*	**PGP** *node:10680* *edge:24316*
TIES	0.0066	0.0071	0.0075	0.0146
RMSC	0.0006	0.0012	0.0058	0.0340
FF	0.0021	0.0030	0.0121	0.0355
RDN	0.0019	0.0033	1.5038	0.4520
DPL	1.0846	1.5198	3.4403	25.3200
MCGS (Ours)	0.7712	0.9078	9.4609	34.7743
DLAS	17.9142	14.6273	10.7340	21.3550
SST	0.4130	0.8868	28.7397	83.3247

We found that the eight algorithms can be divided into two groups. The first group included TIES, RMSC, FF, and RDN. Their time consumptions were lower than those of algorithms in the second group, which included DPL, MCGS, DLAS, and SST. The main reason was that the algorithms of the second group commonly had additional computation steps during random sampling. For example, DPL detected communities and SST generated spanning trees. Such steps in MCGS (i.e., minority structure identification, importance ranking, and loss function computation) were not very time-consuming. Therefore, MCGS presented relatively good time consumptions in the second group. Moreover, MCGS was very fast on the Facebook1912 and Facebook107 due to its insensitivity to the scale of edges. In summary, the time performance of MCGS was at the low-medium level in the experiment. The time performance can be further improved by adopting parallel computations or simplifying the greedy strategy by selecting the optimal node out of random nodes rather than all remained nodes.

6.2 Subjective Assessment

We recruited the 20 participants in the pilot user study again to conduct a subjective assessment experiment. They were asked to assess similarities between a graph and samples by perceiving node-link diagrams [80] and rating on six metrics. Three of the metrics were similarities of the overall shape, community, and connectivity for assessing majority structure preservation. The other three metrics were similarities of high degree, margin, and boundary structures related to the preservation of minority structures. We selected eight popular graphs from the 34 graphs used in the user study. We selected the proposed MCGS and five reference algorithms, namely, RDN, TIES, FF, RW, and SST, most of which performed relatively well in the previous experiments. We used the same high degree nodes as initial seeds. The sampling rate was set at 30%, which is empirically suitable for visual perception [40,41]. A graph and six randomly arranged samples were presented at a time (Figure 4). The participants rated on each sample from the six metrics with a five-point Likert scale ranging from 1 (the lowest similarity) to 5 (the highest similarity).

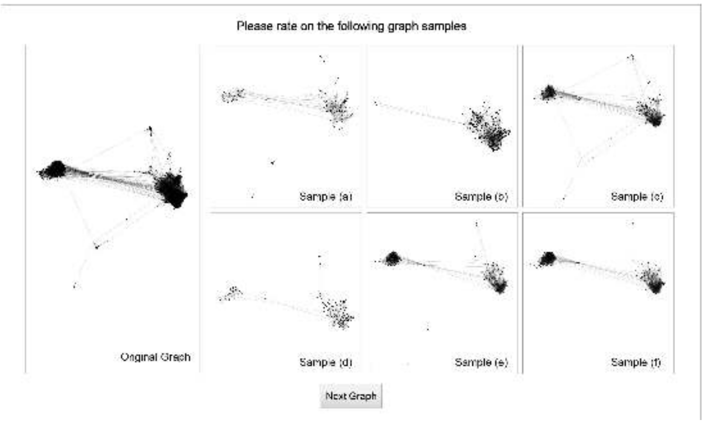

Figure 4. Interface for subjective assessment experiment with the Facebook1684 data set (775 nodes and 14,006 edges). The leftmost graph is the original graph. The six randomly arranged samples are generated by SST (a), FF (b), MCGS (c), RW (d), TIES (e), and RDN (f) with a sampling rate of 30%.

The results of the top three algorithms on each metric in terms of average rating are as follows. (1) Overall shape: MCGS (μ = 4.3), TIES (μ = 3.7), and RDN (μ = 3.5); (2) Community: MCGS (μ = 4.3), TIES (μ = 3.8), and RDN (μ = 3.7); (3) Connectivity: MCGS (μ = 4.3), TIES (μ = 3.6), and RDN (μ = 3.2); (4) High degree structure: MCGS (μ = 4.1), TIES (μ = 3.7), and RDN (μ = 3.6); (5) Margin structure: MCGS (μ = 4.0), TIES (μ = 3.0), and SST (μ = 2.9); (6) Boundary structure: MCGS (μ = 4.1), TIES (μ = 3.1), and SST (μ = 2.9). MCGS obtained the highest average rating six times, followed by TIES and RDN with relatively high average ratings. The results reflected that the minority structure preservation affected perceived similarities to a certain extent and the MCGS samples achieved considerable perceived similarities. In the interview, we focused on similarity judgment principles. Most of the participants stated that they initially observed the overall shape and connectivity and then considered visually prominent minority structures.

6.3 Case Studies

We used three popular graph data sets to demonstrate the features of MCGS. The reference algorithms, initial seeds, sampling rate, and layout method were consistent with the subjective assessment experiment. In node-link diagrams, the relative locations of nodes in a sample are consistent with those of the corresponding nodes in the original graph. Additional cases are provided in the supplementary material.

6.3.1 AS-733 Graph Data Set

The AS-733 graph data set [48] is an autonomous systems network on the Internet with 6,474 nodes and 13,895 edges. The original graph and samples obtained by RDN, SST, MCGS are shown in Figure 5.

We marked four SOIs popular in the pilot user study, as shown in Figure 5(a). SOI-1 and SOI-2 were the first and second largest communities in the graph, respectively. SOI-3 was a visually prominent super pivot in a relatively sparse area. SOI-4 was a huge star far way the two communities. In SOI-1, the top five important super pivots were ranked and named as (a [degree = 1,460] > b [degree = 752] > c [degree = 693] > d [degree = 401] > e [degree = 378]) in a descending order of degree.

For the RDN sample in Figure 5(b), the overall shape and density distribution of the original graph were considerably preserved. The five super pivots in SOI-1 were maintained, but only the top two super pivots (a [586] > b [367] > d [350] > c [205] > the other [145]) were consistent with those in the original graph in order. The super pivot in SOI-3 was well preserved. The huge star in SOI-4 was retained but lost many neighbors.

The SST sample in Figure 5(c) was of low similarity with the original graph in the overall shape and density distribution. The second largest community in SOI-2 and the huge star in SOI-4 disappeared. The five super pivots in SOI-1 were maintained. The top three important super pivots (a [499] > b [228] > c [196] > the other [156] > d [118]) were preserved in order; however, most of their preserved neighbors were the nodes with the degree of 1 and many inter-connections in the community lost. This situation was also reflected in SOI-3.

For the MCGS sample in Figure 5(d), the overall shape and the density distribution were well preserved. The five super pivots in SOI-1 were maintained, and the top four important ones (a [589] > b [338] > c [301] > d [196] > the other [179]) were consistent with those in the original graph in order. The super pivot in SOI-3 and the huge star in SOI-4 were well preserved.

Among the three algorithms, RDN performed best on majority structure preservation, followed by MCGS. MCGS performed best on preserving super pivots and huge stars, especially for the maintenance of the importance order of super pivots. SST only performed well in preserving one-degree neighbors of super pivots.

6.3.2 Cpan Graph Data Set

The Cpan data set is a collaboration network with 839 nodes and 2,127 edges [1]. It depicts the relationships between the developers using the same Perl modules. The original graph and samples obtained by FF, TIES, and MCGS are shown in Figure 6. This case focused on the preservation of parachute-like rims at marginal areas. The top four important rims in the original graph were marked as a > b > c > d in descending order in Figure 6(a). The FF sample in Figure 6(b) presented an overall shape dissimilar to the original graph, and only one of the four important rims and another rim were maintained. The TIES sample in Figure 6(c) presented an overall shape greatly similar to the original graph, and the four important rims were preserved with a changed importance order (a > d > b = c) and unclear parachute shapes. For the MCGS sample in Figure 6(d), the preservation of the overall shape was slightly worse than that of the TIES sample. The four rims were preserved with clear parachute shapes, and their importance order was completely maintained (a > b > c > d).

6.3.3 Facebook1684 Graph Data Set

The Facebook1684 graph data set is an online social network with 775 nodes and 14,006 edges. The original graph and six samples are shown in Figure 4. This data set is an unbalanced graph in which two large communities include 705 nodes and two small communities/cliques contain only 70 nodes. This case focused on the preservation of ties between communities in an unbalanced graph. The SST (a), TIES (e), and RDN (f) samples maintained a few nodes in the two small communities and lost the connections between small and large communities. The FF (b) and RW (d) samples lost the two small communities and generated new chain-like rims at the margins of the largest communities. The MCGS sample (c) was the only one that effectively preserved the two small communities and the connections between small and large communities.

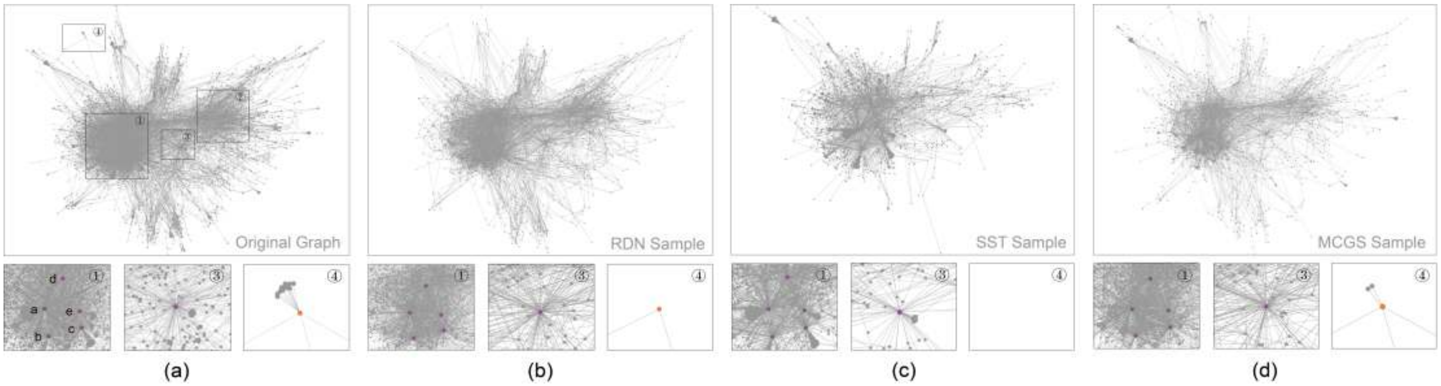

Figure 5. Case analysis of the AS-733 graph data set (a) and three samples generated by RDN (b), SST (c), and MCGS (d) with a sampling rate of 30%.

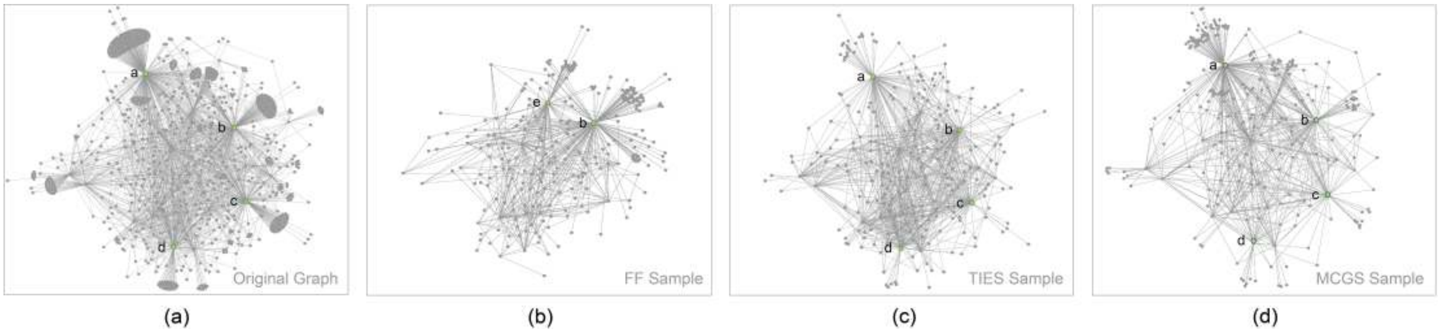

Figure 6. Visual illustration of the Cpan graph data set (a) and three samples generated by FF (b), TIES (c), and MCGS (d) with a sampling rate of 30%.

7 Discussion

In this section, we discuss the limitations of this work and suggest directions for further work.

7.1 Limitations

We mainly used scale-free graphs in this work, such as social, communication, and web networks. Whether our MCGS is applicable for other types of graphs, such as biological networks [4], bipartite graphs [5], and signed networks [47], needs to be deliberated. Our definitions of the four minority structures could be different in other graph types.

The experimental study demonstrated that the existing algorithms had a low ability of preserving minority structures. However, two points should be noted. (1) Existing algorithms did not perform well just because they are not originally designed for minority structure preservation. They still had distinguished competences in diverse application scenarios [29], such as RW in large graph estimation and RE in computational cost reduction. (2) Our MCGS is mainly suitable for graph analyses oriented to minority structures, such as graph visualization [12, 65] and anomaly detection. Its ability for other scenarios remains unknown.

In the pilot user study, we ranked the eight SOI types only depending on the number of entries. In practice, the importance of SOI types should be considered. For example, BS type should rank first in network vulnerability analysis. Likewise, the definitions of minority structure types could be adjusted on demands [70, 83]. For example, HD-global structures are not needed to be subdivided into super pivots and huge stars in some cases. A tie was strictly defined as a single chain bridging two communities in this work, but communities may be connected by multiple chains.

In the evaluation, the experiment of majority structure preservation was not fully comprehensive. Some common metrics, such as clustering coefficient distribution and connected component size distribution [46, 64], were not included. In the subjective assessment experiment, we presented six samples at a time to facilitate a convenient comparative perception, but this manner may cause differentiated ratings. An iterative manner is to show one sample at a time. Moreover, we tested the proposed MCGS on large-scale graphs (see the supplementary material). The experimental results showed that the graph visualizations after sampling still presented severe visual clutters, even when the sampling rates were very low, which reflects that sampling in the data space may not be adequately suitable for visualizing large-scale graphs. It is worth to explore if sampling in the visual space can solve this challenge.

7.2 Further work

In the pilot user study, advanced techniques could be adopted in the future, such as using crowdsourcing approaches [9] to involve extensive participants and using eye tracking [10] to improve our manual SOI classification.

In the experimental study, a ranking preservation indicator can be designed to evaluate the matching accuracy of the rankings of important minority structures before and after sampling, referring to the normalized discounted cumulative gain in the recommendation system community [11]. Sampling rates with a short interval and a wide range should be examined to find an appropriate sampling rate for minority structure preservation.

In the algorithm study, we plan to properly modify MCGS to extend its applied scope from undirected graphs to directed graphs or from scale-free graphs to other types of graphs [38]. We plan to add new types of minority structures and new objectives of loss function into the sampling process. Moreover, comprehensive methods for unbalanced graph identification and partition should be further studied. A layout that can present minority structures distinctly is worth further exploration [19, 22].

8 Conclusion

This work investigated the preservation of minority structures in graph sampling. We conducted a pilot user study and identified four representative types of minority structures. We conducted an experimental study and found that existing algorithms cannot effectively preserve the four types of minority structures. We designed a new graph sampling algorithm named MCGS that presented great performance of minority structure preservation in a series of experiments. This work is the first investigation of minority structure preservation in graph sampling. We hope this work will be conducive to the research and application of graph analyses oriented to minority structures. We also expect that this work will inspire other researchers to further study minority structure classification, identification, sampling, and visualization.

Acknowledgments

We wish to thank Xin Sun from Adobe and Qiaomu Shi and Xinming Chen from Alibaba Cloud Intelligent Security Lab for the fruitful discussions. The work is supported in part by the National Key Research and Development Program of China (No. 2018YFB1700403) and the National Natural Science Foundation of China (No. 61872388, 61872389, 61872314, 61672538, and 61672308). MCGS at Github: `https://github.com/csuvis/MCGS`.

References

[1] Cpan-explorer, an interactive exploration of the perl ecosystem. http://www.cpan-explorer.org/, 2009. Produced by RTGI Labs and Gephi.

[2] A. Ahmed, T. Dywer, S.-H. Hong, C. Murray, L. Song, and Y. X. Wu. Visualisation and analysis of large and complex scale-free networks. In *Proceedings of the 7th Joint Eurographics / IEEE VGTC Conference on Visualization*, pp. 239–246, 2005. doi: 10.2312/VisSym/EuroVis05/239-246

[3] N. Ahmed, J. Neville, and R. R. Kompella. Network sampling via edge-based node selection with graph induction. Technical Report 11-016, Department of Computer Science, Purdue University, 2011.

[4] M. Albrecht, A. Kerren, K. Klein, O. Kohlbacher, P. Mutzel, W. Paul, F. Schreiber, and M. Wybrow. On open problems in biological network visualization. In *Proceedings of the International Symposium on Graph Drawing*, pp. 256–267, 2010.

[5] A. S. Asratian, T. M. Denley, and R. Häggkvist. *Bipartite graphs and their applications*, vol. 131. Cambridge university press, 1998.

[6] S. T. Barnard and H. D. Simon. Fast multilevel implementation of recursive spectral bisection for partitioning unstructured problems. *Concurrency: Practice and Experience*, 6(2):101–117, 1994. doi: 10.1002/cpe.4330060203

[7] N. Blagus, L. Šubelj, and M. Bajec. Assessing the effectiveness of real-world network simplification. *Physica A: Statistical Mechanics and its Applications*, 413:134–146, 2014. doi: 10.1016/j.physa.2014.06.065

[8] S. P. Borgatti and M. G. Everett. Models of core/periphery structures. *Social Networks*, 21(4):375–395, 2000.

[9] R. Borgo, L. Micallef, B. Bach, F. McGee, and B. Lee. Information visualization evaluation using crowdsourcing. *Computer Graphics Forum*, 37(3):573–595, 2018. doi: 10.1111/cgf.13444

[10] M. Burch, F. Beck, M. Raschke, T. Blascheck, and D. Weiskopf. A dynamic graph visualization perspective on eye movement data. In *Proceedings of the Symposium on Eye Tracking Research and Applications*, pp. 151–158, 2014. doi: 10.1145/2578153.2578175

[11] R. Busa-Fekete, G. Szarvas, T. Elteto, and B. Kégl. An apple-to-apple comparison of learning-to-rank algorithms in terms of normalized discounted cumulative gain. In *Proceedings of the 20th European Conference on Artificial Intelligence: Preference Learning: Problems and Applications in AI Workshop*, vol. 242, 2012.

[12] W. Chen, F. Guo, D. Han, J. Pan, X. Nie, J. Xia, and X. Zhang. Structure-based suggestive exploration: A new approach for effective exploration of large networks. *IEEE Transactions on Visualization and Computer Graphics*, 25(1):555–565, 2019. doi: 10.1109/TVCG.2018.2865139

[13] G. Chiandussi, M. Codegone, S. Ferrero, and F. E. Varesio. Comparison of multi-objective optimization methodologies for engineering applications. *Computers & Mathematics with Applications*, 63(5):912–942, 2012. doi: 10.1016/j.camwa.2011.11.057

[14] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein. *Introduction to algorithms*. MIT press, 2009.

[15] C. Doerr and N. Blenn. Metric convergence in social network sampling. In *Proceedings of the 5th ACM workshop on HotPlanet*, pp. 45–50, 2013. doi: 10.1145/2491159.2491168

[16] X. Du, D. Wang, Y. Ye, Y. Li, Y. Li, et al. Sgp: a social network sampling method based on graph partition. *International Journal of Information Technology and Management*, 18(2/3):227–242, 2019. doi: 10.1504/IJITM.2019.099809

[17] K. Duh, K. Sudoh, X. Wu, H. Tsukada, and M. Nagata. Learning to translate with multiple objectives. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics*, vol. 1, pp. 1–10, 2012.

[18] C. Dunne and B. Shneiderman. Motif simplification: Improving network visualization readability with fan, connector, and clique glyphs. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pp. 3247–3256, 2013.

[19] T. Dwyer, K. Marriott, F. Schreiber, P. Stuckey, M. Woodward, and M. Wybrow. Exploration of networks using overview+ detail with constraint-based cooperative layout. *IEEE Transactions on Visualization and Computer Graphics*, 14(6):1293–1300, 2008. doi: 10.1109/TVCG.2008.130

[20] P. Eades, S.-H. Hong, K. Klein, and A. Nguyen. Shape-based quality metrics for large graph visualization. In *Proceedings of the International Symposium on Graph Drawing*, pp. 502–514, 2015. doi: 10.1007/978-3-319-27261-0_41

[21] W. Eberle and L. Holder. Discovering structural anomalies in graph-based data. In *Proceedings of the 7th IEEE International Conference on Data Mining Workshops*, pp. 393–398, 2007. doi: 10.1109/ICDMW.2007.91

[22] O. Ersoy, C. Hurter, F. Paulovich, G. Cantareiro, and A. Telea. Skeleton-based edge bundling for graph visualization. *IEEE Transactions on Visualization and Computer Graphics*, 17(12):2364–2373, 2011. doi: 10.1109/TVCG.2011.233

[23] S. Even. *Graph algorithms*. Cambridge University Press, 2011.

[24] J. H. Friedman. Greedy function approximation: a gradient boosting machine. *The Annals of Statistics*, 29(5):1189–1232, 2001. doi: 10.2307/2699986

[25] B. A. Galler and M. J. Fisher. An improved equivalence algorithm. *Communications of the ACM*, 7(5):301–303, 1964. doi: 10.1145/364099.364331

[26] Q. Gao, X. Ding, F. Pan, and W. Li. An improved sampling method of complex network. *International Journal of Modern Physics C*, 25(05):1440007, 2014. doi: 10.1142/S0129183114400075

[27] R. Gao, P. Hu, and W. C. Lau. Graph property preservation under community-based sampling. In *Proceedings of the 2015 IEEE Global Communications Conference*, pp. 1–7, 2015. doi: 10.1109/GLOCOM.2015.7417471

[28] R. Gao, H. Xu, P. Hu, and W. C. Lau. Accelerating graph mining algorithms via uniform random edge sampling. In *Proceedings of the 2016 IEEE International Conference on Communications*, pp. 1–6, 2016. doi: 10.1109/ICC.2016.7511156

[29] S. Ghani, B. C. Kwon, S. Lee, J. S. Yi, and N. Elmqvist. Visual analytics for multimodal social network analysis: A design study with social scientists. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2032–2041, 2013. doi: 10.1109/TVCG.2013.223

[30] L. A. Goodman. Snowball sampling. *The Annals of Mathematical Statistics*, 32(1):148–170, 1961. doi: 10.2307/2237615

[31] E. Griechisch and A. Pluhár. Community detection by using the extended modularity. *Acta Cybernetica*, 20(1):69–85, 2011. doi: 10.14232/actacyb.20.1.2011.6

[32] K. Henderson, B. Gallagher, T. Eliassi-Rad, H. Tong, S. Basu, L. Akoglu, D. Koutra, C. Faloutsos, and L. Li. Rolx: Structural role extraction & mining in large graphs. In *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 1231–1239, 2012. doi: 10.1145/2339530.2339723

[33] K. Henderson, B. Gallagher, L. Li, L. Akoglu, T. Eliassi-Rad, H. Tong, and C. Faloutsos. It's who you know: Graph mining using recursive structural features. In *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 663–671, 2011. doi: 10.1145/2020408.2020512

[34] P. Hu and W. C. Lau. A survey and taxonomy of graph sampling. *arXiv preprint arXiv:1308.5865*, abs/1308.5865, 2013.

[35] M. Jacomy, T. Venturini, S. Heymann, and M. Bastian. Forceatlas2, a continuous graph layout algorithm for handy network visualization designed for the gephi software. *PloS One*, 9(6):e98679, 2014. doi: 10.1371/journal.pone.0098679

[36] Z. S. Jalali, A. Rezvanian, and M. R. Meybodi. Social network sampling using spanning trees. *International Journal of Modern Physics C*, 27(05):1650052, 2016. doi: 10.1142/S0129183116500522

[37] A. Kaplan, H. Hofmann, and D. Nordman. An interactive graphical method for community detection in network data. *Computational Statistics*, 32(2):535–557, 2017.

[38] A. Kerren, H. Purchase, and M. Ward. Introduction to multivariate network visualization. In *Multivariate Network Visualization*, pp. 1–9. Springer International Publishing, 2014. doi: 10.1007/978-3-319-06793-3_1

[39] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi. Optimization by simulated annealing. *Science*, 220(4598):671–680, 1983. doi: 10.1126/science.220.4598.671

[40] V. Krishnamurthy, M. Faloutsos, M. Chrobak, L. Lao, J.-H. Cui, and A. G. Percus. Reducing large internet topologies for faster simulations. In *International Conference on Research in Networking*, pp. 328–341, 2005. doi: 10.1007/11422778_27

[41] V. Krishnamurthy, J. Sun, M. Faloutsos, and S. L. Tauro. Sampling internet topologies: How small can we go? In *Proceedings of the International Conference on Internet Computing*, pp. 577–580, 2003. doi: 10.1.1.131.1420

[42] M. Kurant, A. Markopoulou, and P. Thiran. Towards unbiased bfs sampling. *IEEE Journal on Selected Areas in Communications*, 29(9):1799–1809, 2011.

[43] B. C. Kwon, J. Verma, P. J. Haas, and C. Demiralp. Sampling for scalable visual analytics. *IEEE Computer Graphics and Applications*, 37(1):100–108, 2017. doi: 10.1109/MCG.2017.6

[44] L. Lee. On the effectiveness of the skew divergence for statistical language analysis. In *Proceedings of the 8th International Workshop on Artificial Intelligence and Statistics*, 2001. doi: 10.1.1.591.1155

[45] S. H. Lee, P. J. Kim, and H. Jeong. Statistical properties of sampled networks. *Physical review E*, 73:016102, 2006. doi: 10.1103/PhysRevE.73.016102

[46] J. Leskovec and C. Faloutsos. Sampling from large graphs. In *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 631–636, 2006. doi: 10.1145/1150402.1150479

[47] J. Leskovec, D. Huttenlocher, and J. Kleinberg. Signed networks in social media. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pp. 1361–1370, 2010. doi: 10.1145/1753326.1753532

[48] J. Leskovec, J. Kleinberg, and C. Faloutsos. Graphs over time: Densification laws, shrinking diameters and possible explanations. In *Proceedings of the 11th ACM SIGKDD International Conference on Knowledge Discovery in Data Mining*, pp. 177–187, 2005. doi: 10.1145/1081870.1081893

[49] R. Li, J. X. Yu, L. Qin, R. Mao, and T. Jin. On random walk based graph sampling. In *Proceedings of the 31st IEEE International Conference on Data Engineering*, pp. 927–938, 2015. doi: 10.1109/ICDE.2015.7113345

[50] M. Liu, J. Shi, Z. Li, C. Li, J. Zhu, and S. Liu. Towards better analysis of deep convolutional neural networks. *IEEE Transactions on Visualization and Computer Graphics*, 23(1):91–100, 2016. doi: 10.1109/TVCG.2016.2598831

[51] A. Marsden. Eigenvalues of the laplacian and their relationship to the connectedness of a graph. *University of Chicago REU Program*, 2013.

[52] F. J. Massey Jr. The kolmogorov-smirnov test for goodness of fit. *Journal of the American statistical Association*, 46(253):68–78, 1951. doi: 10.1080/01621459.1951.10500769

[53] M. Mitchell. *An introduction to genetic algorithms*. MIT press, 1998.

[54] Q. H. Nguyen, S. H. Hong, P. Eades, and A. Meidiana. Proxy graph: Visual quality metrics of big graph sampling. *IEEE Transactions on Visualization and Computer Graphics*, 23(6):1600–1611, 2017. doi: 10.1109/TVCG.2017.2674999

[55] C. C. Noble and D. J. Cook. Graph-based anomaly detection. In *Proceedings of the ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 631–636, 2003. doi: 10.1145/956750.956831

[56] D. Pelleg and A. W. Moore. Active learning for anomaly and rare-category detection. In *Proceedings of the 17th International Conference on Neural Information Processing Systems*, pp. 1073–1080, 2004.

[57] P. Polatsek, M. Waldner, I. Viola, P. Kapec, and W. Benesova. Exploring visual attention and saliency modeling for task-based visual analysis. *Computers & Graphics*, 72:26–38, 2018. doi: 10.1016/j.cag.2018.01.010

[58] D. Rafiei. Effectively visualizing large networks through sampling. In *16th IEEE Visualization*, pp. 375–382, 2005. doi: 10.1109/VISUAL.2005.1532819

[59] A. Rezvanian and M. R. Meybodi. Sampling social networks using shortest paths. *Physica A: Statistical Mechanics and its Applications*, 424:254–268, 2015. doi: 10.1016/j.physa.2015.01.030

[60] A. Rezvanian, M. Rahmati, and M. R. Meybodi. Sampling from complex networks using distributed learning automata. *Physica A: Statistical Mechanics and its Applications*, 396:224–234, 2014. doi: 10.1016/j.physa.2013.11.015

[61] B. Ribeiro and D. Towsley. Estimating and sampling graphs with multidimensional random walks. In *Proceedings of the 10th ACM SIGCOMM Conference on Internet Measurement*, pp. 390–403, 2010. doi: 10.1145/1879141.1879192

[62] A. Sheikhahmadi, M. A. Nematbakhsh, and A. Shokrollahi. Improving detection of influential nodes in complex networks. *Physica A: Statistical Mechanics and its Applications*, 436:833–845, 2015. doi: 10.1016/j.physa.2015.04.035

[63] H. Shiokawa, Y. Fujiwara, and M. Onizuka. Scan++: Efficient algorithm for finding clusters, hubs and outliers on large-scale graphs. *Proceedings of the VLDB Endowment*, 8(11):1178–1189, 2015. doi: 10.14778/2809974.2809980

[64] U. Soni, Y. Lu, B. Hansen, H. C. Purchase, S. Kobourov, and R. Maciejewski. The perception of graph properties in graph layouts. *Computer Graphics Forum*, 37(3):169–181, 2018. doi: 10.1111/cgf.13410

[65] M. Streit and M. Kalkusch. Interactive visualization of complex graphs. In *Proceedings of the Central European Seminar on Computer Graphics*, pp. 1–8, 2007.

[66] M. P. Stumpf and C. Wiuf. Sampling properties of random graphs: The degree distribution. *Physical Review E*, 72:036118, 2005. doi: 10.1103/PhysRevE.72.036118

[67] D. Stutzbach, R. Rejaie, N. Duffield, S. Sen, and W. Willinger. On unbiased sampling for unstructured peer-to-peer networks. *IEEE/ACM Transactions on Networking*, 17(2):377–390, 2009. doi: 10.1109/TNET.2008.2001730

[68] E. Sullivan, M. Sondag, I. Rutter, W. Meulemans, S. Cunningham, B. Speckmann, and M. Alfano. Vulnerability in social epistemic networks. *International Journal of Philosophical Studies*, forthcoming.

[69] I. Trpevski, T. Dimitrova, T. Boshkovski, N. Stikov, and L. Kocarev. Graphlet characteristics in directed networks. *Scientific Reports*, 6(1):1–8, 2016. doi: 10.1038/srep37057

[70] F. Van Ham, H. J. Schulz, and J. M. Dimicco. Honeycomb: Visual analysis of large scale social networks. In *Proceedings of the IFIP Conference on Human-Computer Interaction*, pp. 429–442, 2009.

[71] J. L. Walteros, A. Veremyev, P. M. Pardalos, and E. L. Pasiliao. Detecting critical node structures on graphs: A mathematical programming approach. *Networks*, 73(1):48–88, 2019.

[72] T. Wang, Y. Chen, Z. Zhang, T. Xu, L. Jin, P. Hui, B. Deng, and X. Li. Understanding graph sampling algorithms for social network analysis. In *2011 31st International Conference on Distributed Computing Systems Workshops*, pp. 123–128, 2011. doi: 10.1109/ICDCSW.2011.34

[73] Y. Wei, H. Mei, Y. Zhao, S. Zhou, B. Lin, H. Jiang, and W. Chen. Evaluating perceptual bias during geometric scaling of scatterplots. *IEEE Transactions on Visualization and Computer Graphics*, 26(1):321–331, 2019. doi: 10.1109/TVCG.2019.2934208

[74] S. Wu. Evaluation of retrieval results. In *Data Fusion in Information Retrieval*, pp. 7–18. Springer Berlin Heidelberg, 2012. doi: 10.1007/978-3-642-28866-1_2

[75] Y. Wu, N. Cao, D. Archambault, Q. Shen, H. Qu, and W. Cui. Evaluation of graph sampling: A visualization perspective. *IEEE Transactions on Visualization and Computer Graphics*, 23(1):401–410, 2016. doi: 10.1109/TVCG.2016.2598867

[76] Y. Wu, W. Wu, S. Yang, Y. Yan, and H. Qu. Interactive visual summary of major communities in a large network. In *Proceedings of the 2015 IEEE Pacific Visualization Symposium*, pp. 47–54, 2015. doi: 10.1109/PACIFICVIS.2015.7156355

[77] J. Xia, F. Ye, W. Chen, Y. Wang, W. Chen, Y. Ma, and A. K. Tung. Ldsscanner: Exploratory analysis of low-dimensional structures in high-dimensional datasets. *IEEE Transactions on Visualization and Computer Graphics*, 24(1):236–245, 2017. doi: 10.1109/TVCG.2017.2744098

[78] J. Xu and H. Chen. Untangling criminal networks: A case study. In *International Conference on Intelligence and Security Informatics*, pp. 232–248, 2003.

[79] J. Yang, Y. Liu, X. Zhang, X. Yuan, Y. Zhao, S. Barlowe, and S. Liu. Piwi: visually exploring graphs based on their community structure. *IEEE Transactions on Visualization and Computer Graphics*, 19(6):1034–1047, 2012. doi: 10.1109/TVCG.2012.172

[80] V. Yoghourdjian, D. Archambault, S. Diehl, T. Dwyer, K. Klein, H. C. Purchase, and H.-Y. Wu. Exploring the limits of complexity: A survey of empirical studies on graph visualisation. *Visual Informatics*, 2(4):264–282, 2018. doi: 10.1016/j.visinf.2018.12.006

[81] S. H. Yoon, K. N. Kim, J. Hong, S. W. Kim, and S. Park. A community-based sampling method using dpl for online social networks. *Information Sciences*, 306:53–69, 2015. doi: 10.1016/j.ins.2015.02.014

[82] F. Zhang, S. Zhang, P. Chung Wong, H. Medal, L. Bian, I. Swan, J. Edward, and T. Jankun-Kelly. A visual evaluation study of graph sampling techniques. *Electronic Imaging*, 2017(1):110–117, 2017. doi: 10.2352/ISSN.2470-1173.2017.1.VDA-394

[83] J. Zhang, J. Chae, S. Afzal, A. Malik, D. Thom, Y. Jang, T. Ertl, S. A. Matei, and D. S. Ebert. Visual analytics of user influence and location-based social networks. In *Transparency in Social Media: Tools, Methods and Algorithms for Mediating Online Interactions*, pp. 223–237. Springer International Publishing, 2015. doi: 10.1007/978-3-319-18552-1_12

[84] Y. Zhao, F. Luo, M. Chen, Y. Wang, J. Xia, F. Zhou, Y. Wang, Y. Chen, and W. Chen. Evaluating multi-dimensional visualizations for understanding fuzzy clusters. *IEEE Transactions on Visualization and Computer Graphics*, 25(1):12–21, 2018. doi: 10.1109/TVCG.2018.2865020

[85] Y. Zhao, Y. She, W. Chen, Y. Lu, J. Xia, W. Chen, J. Liu, and F. Zhou. Eod edge sampling for visualizing dynamic network via massive sequence view. *IEEE Access*, 6:53006–53018, 2018. doi: 10.1109/ACCESS.2018.2870684

[86] D. Zhou, A. Karthikeyan, K. Wang, N. Cao, and J. He. Discovering rare categories from graph streams. *Data Mining and Knowledge Discovery*, 31(2):400–423, 2017.

[87] F. Zhou, X. Lin, C. Liu, Y. Zhao, P. Xu, L. Ren, T. Xue, and L. Ren. A survey of visualization for smart manufacturing. *Journal of Visualization*, 22(2):419–435, 2019. doi: 10.1007/s12650-018-0530-2

[88] Z. Zhou, L. Meng, C. Tang, Y. Zhao, Z. Guo, M. Hu, and W. Chen. Visual abstraction of large scale geospatial origin-destination movement data. *IEEE Transactions on Visualization and Computer Graphics*, 25(1):43–53, 2019. doi: 10.1109/TVCG.2018.2864503

[89] R. Zou and L. B. Holder. Frequent subgraph mining on a single large graph using sampling techniques. In *Proceedings of the 8th Workshop on Mining and Learning with Graphs*, pp. 171–178, 2010. doi: 10.1145/1830252.1830274