# DSS: Drawing Dynamic Graphs with Spectral Sparsification†

Amyra Meidiana[1], Seok-Hee Hong[1], Yanyi Pu[1], Justin Lee[1], Peter Eades[1], and Jinwook Seo[2]

[1]University of Sydney, Australia [2]Seoul National University, South Korea

**Abstract**

*This paper presents DSS (Dynamic Spectral Sparsification), a sampling approach for drawing large and complex dynamic graphs which can preserve important structural properties of the original graph. Specifically, we present two variants:* DSS-I (Independent) *which performs spectral sparsification independently on each dynamic graph time slice; and* DSS-U (Union) *which performs spectral sparsification on the union graph of all time slices. Moreover, for evaluation of dynamic graph drawing using sampling approach, we introduce two new metrics: DSQ* (Dynamic Sampling Quality) *to measure how faithfully the samples represent the ground truth change in the dynamic graph, and DSDQ* (Dynamic Sampling Drawing Quality) *to measure how faithfully the drawings of the sample represent the ground truth change. Experiments demonstrate that DSS significantly outperform random sampling on quality metrics and visual comparison. On average, DSS obtains over 80% (resp., 30%) better DSQ (resp., DSDQ) than random sampling, and visually better preserves the ground truth changes in dynamic graphs.*

## 1. Introduction

*Graph sampling* is often used to address scalability issues for analysis and visualization of large graphs [HL13; LF06], however most previous work focuses on *static* graphs [HNM*18; HL20; WCA*17; ZZW*15]. *Dynamic graphs*, with structural changes over time, add significant challenges for effective analysis and visualization. Consequently, sampling dynamic graphs adds challenges: samples should faithfully represent not only the ground truth structure at each time slice, but also the changes between time slices.

*Spectral sparsification* computes a subgraph that preserve important structural properties, e.g. commute distance [ST11]. While some theoretical results are known for spectral sparsification of dynamic graph streams [AGM13; KLM*17], practical implementations and empirical validations on the effectiveness of spectral sparsification for dynamic graphs have not been studied.

Furthermore, quality metrics for graph sampling and their drawings are important for *quantitative evaluation*, however existing quality metrics for sampling quality [HL13] and drawing quality [NHEM17] focus on static graphs. Quality metrics for sampling and drawing of dynamic graphs need to measure whether samples and drawings preserve the ground truth changes in dynamic graphs.

We present *DSS* (Dynamic Spectral Sparsification), new sampling algorithms using spectral sparsification for dynamic graphs, with two variants: *I (Independent)* performs *SS* (Spectral Sparsification) independently on each time slice of a dynamic graph; and *U (Union)* performs spectral sparsification on the *union graph* of all time slices. We also introduce two quality metrics for sampling (resp., drawing) dynamic graphs, *DSQ* (Dynamic Sampling Quality) and *DSDQ* (Dynamic Sampling Drawing Quality) to measure how faithfully the change in *samples* and the *drawings* thereof preserve the ground truth changes in the original dynamic graph.

We validate the effectiveness of *DSQ* and *DSDQ* using deformation experiments, showing both metrics can effectively measure the quality of dynamic graph sampling and drawing. We then evaluate *DSS* against random sampling using *DSQ*, *DSDQ* and visual comparison. Experimental results demonstrate that *DSS* significantly outperform random sampling, on both quality metrics and visual comparison. Furthermore, *DSS-U* preserves the dynamics of the original graphs better than *DSS-I* on real-world dynamic graphs.

## 2. Related Work

### 2.1. Dynamic Graph Drawing

The most well-known evaluation criteria for dynamic graph drawing is to preserve the *mental map* [MELS95]. Similarly, *dynamic stability* is defined as the minimization of the geometric distance between subsequent drawings [TDB88; BBDW17].

To preserve the mental map and maximize the stability, the graph layout is often computed using a *union graph* approach, which is defined as $G_u = (V_u, E_u), V_u = \bigcup_{i=1}^{k} V_i, E_u = \bigcup_{i=1}^{k} E_i$. A layout for the union graph is computed, and the same vertex position are used for layouts of each time slice [DGK01; DG02].

† This paper was supported by ARC (Australian Research Council) grants DP180102553 and DP190103301, and Global Research Fellowship from Seoul National University.





DOI: 10.2312/evs.20221093

### 2.2. Static Graph Sampling and Spectral Sparsification

Sampling on *static* graphs has been extensively studied for efficient analysis and visualization of big complex graphs [LF06; HL13]. Random sampling methods are fast and easy to implement but often fail to preserve important structural properties [NHEM17; ZZW*15; WCA*17]. Graph topology has been used with sampling to improve the connectivity of the samples [HNM*18; MHH*19].

*Spectral sparsification* is a subgraph which preserves the structural properties of the original graph, such as commute distance. Every $n$-vertex graph has a spectral approximation with $O(n \log n)$ edges [ST11]. Spectral sparsification selects edges based on *effective resistance* values, defined as the voltage drop across the edge when modelling a graph as an electrical network and can be computed in near-linear time [SS11].

Spectral sparsification has been applied to graph drawing [ENH17; HHE19], producing good quality sample graph drawings preserving the original graph's structure. Spectral sparsification using graph topology [HCH*21; MHH*19] has been presented to reduce runtime and improve quality. Fast spectral sparsification using a multi-level approach has also been presented [ITW*20]. However, most existing spectral sparsification methods focus on *static* graphs, and results for dynamic graphs remain purely theoretical without practical implementations [AGM13; KLM*17].

Graph sampling methods are evaluated using *sampling quality metrics* which compare important statistical properties of the original and sample graph, such as degree correlation, closeness centrality, clustering coefficient, and average neighbor degree [HL13; ZZW*15]. However, these metrics focus on sampling *static* graphs, and not directly applicable for sampling *dynamic* graphs.

### 2.3. Faithfulness Metrics

While *readability* metrics evaluate graph drawings based on how humans understand the drawing, *faithfulness metrics* measure how faithfully a drawing displays the ground truth of a graph.

Faithfulness metrics for *static* graphs are well studied: the *shape-based metrics* [EHKN15] measures how faithfully a drawing displays the ground truth structure of a graph using *proximity graphs*, and the *proxy faithfulness metrics* [NHEM17] measures how faithfully the drawing of a sample graph displays the structure of the original graph. Other examples include the *cluster* faithfulness [MHEK19] and the *symmetry* faithfulness [MHEK20] metrics.

For *dynamic* graph drawing, *change faithfulness* metrics measure how proportionally geometric change in a dynamic graph drawing displays the ground truth change in dynamic graphs. Examples include the *cluster change* faithfulness and *distance change* faithfulness metrics [MHE20]. However, change faithfulness metrics for *sampling of dynamic graphs* have not been presented yet.

## 3. DSS: Dynamic Spectral Sparsification

We present *DSS* for sampling dynamic graphs with spectral sparsification with two variants: (1) *DSS-I (Independent)*, which samples each time slice independently to be *locally* faithful to each time slice; and (2) *DSS-U (Union)*, which computes samples based on the *union graph* of all time slices to improve *change faithfulness*.

### 3.1. DSS-I (Independent)

For a dynamic graph $G$ with time slices $G_1, \ldots, G_k$, *DSS-I* independently computes spectral sparsification $G'_i$ of size $x\%$ of each time slice $G_i$: (1) Compute the effective resistance values of edges in $G_i$; (2) Compute spectral sparsification $G'_i$ of size $x\%$ by selecting edges in decreasing order of effective resistance values in $G_i$.

With *DSS-I*, we expect to obtain samples that are locally faithful to each time slice. However, they may not be as change faithful, and local sampling alone may miss edges that are locally less important but become more important globally throughout other time slices.

### 3.2. DSS-U (Union)

To improve the issues with *DSS-I*, we design *DSS-U*, which computes effective resistance values of edges using the *union graphs*: (1) Compute the union graph $G_u = (V_u, E_u), V_u = V_1 \cup \ldots \cup V_k, E_u = E_1 \cup \ldots \cup E_k$; (2) Compute effective resistance values of edges in $G_u$; (3) For each time slice $G_i$, compute spectral sparsification $G'_i$ of size $x\%$ by selecting edges in decreasing order of effective resistance values computed on $G_u$.

We expect *DSS-U* to select more globally important edges across all time slices for more *change faithful* samples of dynamic graphs, highlighting not only locally important edges at each time slice but also edges that are globally important throughout more time slices.

## 4. DSQ and DSDQ: New Quality Metrics for Dynamic Graphs

We present new quality metrics: *DSQ* (Dynamic Sampling Quality) and *DSDQ* (Dynamic Sampling Drawing Quality), to measure the quality of *samples* and *drawings of samples* of dynamic graphs.

### 4.1. DSQ: Dynamic Sampling Quality Metric

*DSQ* measures how proportional the *combinatorial* change in the samples of a dynamic graph is to the ground truth change in the original graph. Unlike many existing sampling quality metrics for *static* graphs [HL13], it is specifically designed for evaluating sampling *dynamic* graphs. Namely, given two time slices of a dynamic graph $G_1$ and $G_2$, *DSQ* is computed as the following steps:

1. Compute sampled graphs $G'_1$ and $G'_2$ of $G_1$ and $G_2$.
2. Compare the similarity between the change in the sampled graphs $\Delta(G'_1, G'_2)$ to the ground truth change $\Delta(G_1, G_2)$.

Specifically, given two time slices of a dynamic graph $G_1, G_2$ and the sampled graphs $G'_1, G'_2$, *DSQ* is defined as follows:

$$DSQ = 1 - \frac{|JS(G_1,G_2) - JS(G'_1,G'_2)|}{max(JS(G_1,G_2), JS(G'_1,G'_2))}$$

where $JS(G_1, G_2)$ is the *Mean Jaccard Similarity* [Jac12]:

$$JS(G_1,G_2) = \frac{1}{|V|} \sum_{v \in V} \frac{|N_1(v) \cap N_2(v)|}{|N_1(v) \cup N_2(v)|}$$

where $N_1(v)$ (resp. $N_2(v)$) is the neighbor set of $v$ in $G_1$ (resp., $G_2$). *DSQ* is defined between 0 to 1, where higher value means better.

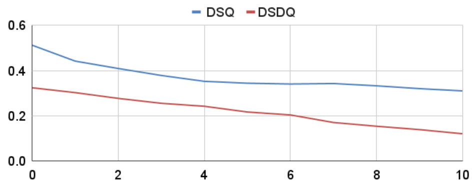



**Figure 1:** *Average DSQ and DSDQ score. Clearly, both metrics decrease along the deformation step, confirming our hypotheses.*

### 4.2. DSDQ: Dynamic Sampling Drawing Quality Metric

*DSDQ* is a *change faithfulness* metric, which measures how proportional the *geometric change* in *drawings* of *sample* graphs is to the ground truth change in a dynamic graph. The change in the drawing is quantified by the change in the *proximity graph* of the drawings, such as *GG* (Gabriel Graph) [GS69] and *RNG* (Relative Neighbourhood Graph) [Tou80]. Specifically, given two time slices of a dynamic graph $G_1$ and $G_2$, *DSDQ* is defined as follows:

1. Compute sampled graphs $G'_1$ and $G'_2$ of $G_1$ and $G_2$.
2. Compute drawings $D'_1$ and $D'_2$ of $G'_1$ and $G'_2$ using a layout.
3. Compute the proximity graphs $S'_1$ and $S'_2$ of $D'_1$ and $D'_2$.
4. Compare the similarity between the change in the proximity graphs $\Delta(S'_1, S'_2)$ to the change in dynamic graphs $\Delta(G_1, G_2)$.

More specifically, *DSDQ* is computed as:

$$DSDQ = 1 - \frac{|JS(G_1,G_2) - JS(S'_1,S'_2)|}{max(JS(G_1,G_2), JS(S'_1,S'_2))}$$

where $S'_1$ (resp., $S'_2$) is the proximity graph of of $D'_1$ (resp., $D'_2$). *DSDQ* ranges between 0 to 1, where higher is better.

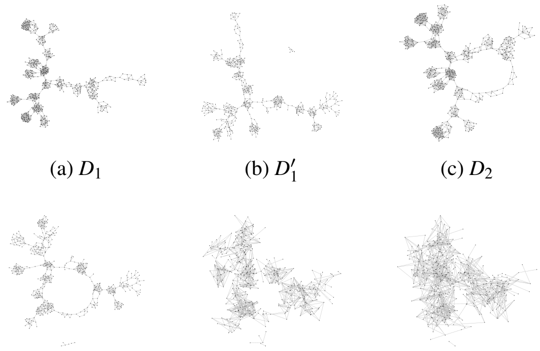
## 5. Validation Experiments: DSQ and DSDQ

To validate the effectiveness of *DSQ* and *DSDQ*, we conduct validation experiments using dynamic graphs based on three graph types: sparse *mesh-type* (M) graphs [DH11], *black-hole* (BH) graphs with global mesh structures and locally dense "blobs" [ENH17], and real-world *scale-free* (SF) graphs with globally sparse but locally dense clusters [LK14].

### 5.1. DSQ Validation Experiment

For a dynamic graph with two time slices $G_1$ and $G_2$, we compute samples $G'_1$ and $G'_2$ by performing *SS* on $G_1$ and $G_2$. We select a sample size of $0.3|E|$, as it produces samples that are similar enough to the original graph. We start $G'_2$ by selecting edges with *high* effective resistance values, and gradually reduce the quality of the sample by selecting edges with *lower* effective resistance values. At step $s$, we exclude the top $s \times k$ edges ($0.04|E| < k < 0.06|E|$), and choose the next $0.3|E|$ edges instead, to make the change in samples less similar to the ground truth change.

Figure 1 shows the *DSQ* metric along deformation steps, averaged across all data sets. Clearly, *DSQ* decreases as the samples are further deformed. *Therefore, deformation experiments show that DSQ effectively measures the quality of dynamic graph samples.*



(a) $D_1$ (b) $D'_1$ (c) $D_2$
(d) $D'_2$ (step 0) (e) $D'_2$ (step 5) (f) $D'_2$ (step 10)

**Figure 2:** *DSDQ deformation experiment: time slices $G_1$, $G_2$ and drawings $D'_1$, $D'_2$ of $G'_1$, $G'_2$, with $D'_2$ gradually deformed.*

### 5.2. DSDQ Validation Experiment

For a dynamic graph with two time slices $G_1$ and $G_2$, we compute samples $G'_1$ and $G'_2$ with sample sizes between $0.3|E|$ to $0.7|E|$, selected to produce similar drawings to the original graph. We first compute *good* drawings $D'_1$ and $D'_2$ using the Organic [WEK04] or Backbone [NLCB13] layout. We then gradually perturb the vertex positions in $D'_2$ to make it less change faithful (Fig. 2).

Figure 1 shows the *DSDQ* metric per deformation step, averaged over all data sets. Clearly, *DSDQ* decreases along the deformations steps. *Therefore, deformation experiments show that DSDQ effectively measures the quality of drawings of dynamic graphs samples.*

## 6. DSS Experiments

### 6.1. Experiment Design

We compare *DSS-I* and *DSS-U* to *RE* (Random Edge), using quality metrics (*DSQ* and *DSDQ*) and visual comparison. We first use graphs with simulated dynamics (i.e., add or delete vertices or edges) with real-world benchmark *scale-free graphs* [LK14] (drawn with the Backbone layout) and *black-hole graphs* [ENH17; HHE19] (drawn with the Organic layout). We also use *real-world dynamic* graphs [LK14; FB14] (drawn with the Backbone layout). We compute samples of size $0.3|E|$ with *DSS-I*, *DSS-U*, and *RE*.

Note that for *DSS-U*, the union graph is used for sampling only while the layout of each sample is computed independently. This is done to not overly constrain the drawings, and we expect that the increased faithfulness in the change of the samples would also lead to higher change faithfulness of the drawings.

We expect *DSS-I* and *DSS-U* to outperform *RE* on both the quality metrics and visual comparison. Furthermore, we expect *DSS-U* to perform better than *DSS-I*, due to the union graph approach.

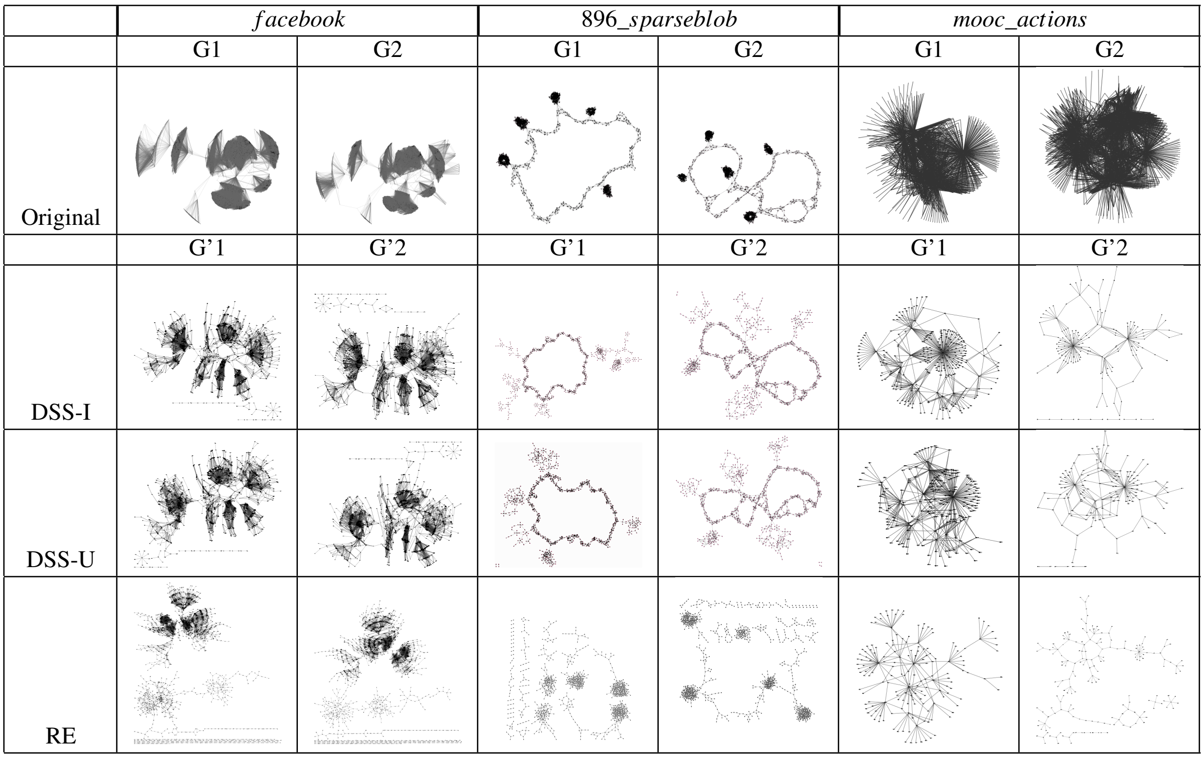
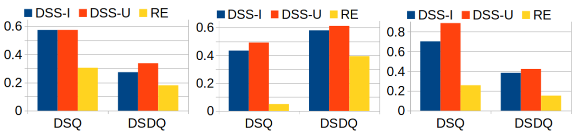



**Table 1:** *facebook (scale-free graph),* 896_*sparseblob (black-hole graph), and mooc_actions (real-world dynamic graph) samples. DSS preserves structural changes better than RE; DSS-U preserves dense areas & change in density better than DSS-I.*




(a) Scale-free (b) Black-hole (c) Real-world

**Figure 3:** *Average DSQ and DSDQ of DSS-I, DSS-U, and RE. DSS vastly outperforms RE, with DSS-U performing better than DSS-I.*

### 6.2. Quality Metrics Comparison

Figure 3 shows the *DSQ* and *DSDQ* metrics, averaged for all data sets. Clearly, *DSS* vastly outperforms *RE*: for black-hole (resp., scale-free) data sets, *DSS* obtains over 47% (resp., 50% ) higher *DSDQ* and 7 (resp., 3) times higher *DSQ* than *RE*; for real-world dynamic graphs, *DSS* perform over 1.5 times better than *RE* on both metrics. Moreover, *DSS-U* performs significantly better than *DSS-I*: for scale-free (resp., black-hole) data sets, 20% (resp., 6%) higher *DSDQ* and around 3% (resp., 13%) higher *DSQ* than *DSS-I*; for real-world dynamic graphs, *DSS-U* obtains 15% higher *DSQ* and 10% higher *DSDQ* than *DSS-I*.

### 6.3. Visual comparison

Table 1 shows visual comparisons of sample graphs, computed by *DSS-I*, *DSS-U*, and *RE*, for dynamic graphs. Clearly, *DSS* preserves the structure of the original dynamic graphs significantly better than *RE*, which fails to preserve the global structure in most time slices. Moreover, *DSS-U* better preserves the locally dense clusters than *DSS-I* as seen for 896_*sparseblob* where *DSS-U* preserves all five blobs in both time slices while one blob is disconnected in the first time slice for *DSS-I*. *DSS-U* also preserves the change in density better than *DSS-I*: for *mooc_actions* , *DSS-U* preserves the change in density in the largest connected component the best.

## 7. Conclusion

We present spectral sparsification approach for dynamic graphs, *DSS*, with two variants *I* and *U*. We also present two new faithfulness metrics for dynamic graph sampling: *DSQ* for the quality of samples, and *DSDQ* for the quality of drawings of samples. Extensive experiments demonstrate that *DSS* greatly outperforms random edge sampling *RE* on quality metrics and visual comparison: *DSS* obtains on average 30% higher *DSDQ* and over 80% higher *DSQ* than *RE*. Furthermore, *DSS-U* perform better than *DSS-I*, at on average 13% higher *DSDQ* and 8% higher *DSQ* than *DSS-I*, as well as preserving locally dense areas and changes in density better.

Future work includes incorporating topological decomposition to further improve the quality of the sampling. Other dynamic sampling quality metrics can also be defined which incorporates statistical properties used for static sampling quality metrics.

## References


[AGM13] AHN, KOOK JIN, GUHA, SUDIPTO, and MCGREGOR, ANDREW. "Spectral sparsification in dynamic graph streams". *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques*. Springer, 2013, 1–10.

[BBDW17] BECK, FABIAN, BURCH, MICHAEL, DIEHL, STEPHAN, and WEISKOPF, DANIEL. "A taxonomy and survey of dynamic graph visualization". *Computer Graphics Forum*. Vol. 36. 1. Wiley Online Library. 2017, 133–159.

[DG02] DIEHL, STEPHAN and GÖRG, CARSTEN. "Graphs, they are changing". *International Symposium on Graph Drawing*. Springer. 2002, 23–31.

[DGK01] DIEHL, STEPHAN, GÖRG, CARSTEN, and KERREN, ANDREAS. "Preserving the mental map using foresighted layout". *Data Visualization 2001*. Springer, 2001, 175–184.

[DH11] DAVIS, TIMOTHY A and HU, YIFAN. "The University of Florida sparse matrix collection". *Transactions on Mathematical Software* 38.1 (2011), 1.

[EHKN15] EADES, PETER, HONG, SEOK-HEE, KLEIN, KARSTEN, and NGUYEN, AN. "Shape-based quality metrics for large graph visualization". *Proc. of Graph Drawing*. Springer. 2015, 502–514.

[ENH17] EADES, PETER, NGUYEN, QUAN, and HONG, SEOK-HEE. "Drawing Big Graphs Using Spectral Sparsification". *Graph Drawing*. Springer. 2017, 272–286.

[FB14] FOURNET, JULIE and BARRAT, ALAIN. "Contact patterns among high school students". *PloS one* 9.9 (2014), e107878.

[GS69] GABRIEL, K RUBEN and SOKAL, ROBERT R. "A new statistical approach to geographic variation analysis". *Systematic zoology* 18.3 (1969), 259–278.

[HCH*21] HU, JINGMING, CHU, TUAN TRAN, HONG, SEOK-HEE, et al. "BC tree-based spectral sampling for big complex network visualization". *Appl. Netw. Sci.* 6.1 (2021), 60.

[HHE19] HU, JINGMING, HONG, SEOK-HEE, and EADES, PETER. "Spectral vertex sampling for big complex graphs". *International Conference on Complex Networks and Their Applications*. Springer. 2019, 216–227.

[HL13] HU, PILI and LAU, W. "A Survey and Taxonomy of Graph Sampling". *ArXiv* abs/1308.5865 (2013).

[HL20] HONG, SEOK-HEE and LU, SHAN. "Graph sampling methods for big complex networks integrating centrality, k-core, and spectral sparsification". *Proceedings of the 35th Annual ACM Symposium on Applied Computing*. 2020, 1843–1851.

[HNM*18] HONG, SEOK-HEE, NGUYEN, QUAN, MEIDIANA, AMYRA, et al. "BC Tree-Based Proxy Graphs for Visualization of Big Graphs". *PacificVis*. IEEE. 2018, 11–20.

[ITW*20] IMRE, MARTIN, TAO, JUN, WANG, YONGYU, et al. "Spectrum-preserving sparsification for visualization of big graphs". *Computers & Graphics* 87 (2020), 89–102.

[Jac12] JACCARD, PAUL. "The distribution of the flora in the alpine zone. 1". *New phytologist* 11.2 (1912), 37–50.

[KLM*17] KAPRALOV, MICHAEL, LEE, YIN TAT, MUSCO, CN, et al. "Single pass spectral sparsification in dynamic streams". *SIAM Journal on Computing* 46.1 (2017), 456–477.

[LF06] LESKOVEC, JURE and FALOUTSOS, CHRISTOS. "Sampling from large graphs". *ACM SIGKDD*. ACM. 2006, 631–636.

[LK14] LESKOVEC, JURE and KREVL, ANDREJ. *SNAP Datasets: Stanford Large Network Dataset Collection*. http://snap.stanford.edu/data. June 2014.

[MELS95] MISUE, KAZUO, EADES, PETER, LAI, WEI, and SUGIYAMA, KOZO. "Layout adjustment and the mental map". *Journal of Visual Languages & Computing* 6.2 (1995), 183–210.

[MHE20] MEIDIANA, AMYRA, HONG, SEOK-HEE, and EADES, PETER. "New Quality Metrics for Dynamic Graph Drawing". *Graph Drawing*. Ed. by AUBER, DAVID and VALTR, PAVEL. Vol. 12590. LNCS. 2020, 450–465.

[MHEK19] MEIDIANA, AMYRA, HONG, SEOK-HEE, EADES, PETER, and KEIM, DANIEL. "A Quality Metric for Visualization of Clusters in Graphs". *Graph Drawing*. Springer. 2019, 125–138.

[MHEK20] MEIDIANA, AMYRA, HONG, SEOK-HEE, EADES, PETER, and KEIM, DANIEL. "Quality Metrics for Symmetric Graph Drawings". *PacificVis*. IEEE. 2020, 11–15.

[MHH*19] MEIDIANA, AMYRA, HONG, SEOK-HEE, HUANG, JIAJUN, et al. "Topology-based spectral sparsification". *LDAV*. 2019, 73–82.

[NHEM17] NGUYEN, QUAN HOANG, HONG, SEOK-HEE, EADES, PETER, and MEIDIANA, AMYRA. "Proxy graph: Visual quality metrics of big graph sampling". *IEEE TVCG* 23.6 (2017), 1600–1611.

[NLCB13] NICK, BOBO, LEE, CONRAD, CUNNINGHAM, PÁDRAIG, and BRANDES, ULRIK. "Simmelian backbones: Amplifying hidden homophily in facebook networks". *ASONAM*. 2013, 525–532.

[SS11] SPIELMAN, DANIEL A and SRIVASTAVA, NIKHIL. "Graph sparsification by effective resistances". *SIAM Journal on Computing* 40.6 (2011), 1913–1926.

[ST11] SPIELMAN, DANIEL A and TENG, SHANG-HUA. "Spectral sparsification of graphs". *SIAM Journal on Computing* 40.4 (2011), 981–1025.

[TDB88] TAMASSIA, ROBERTO, DI BATTISTA, GIUSEPPE, and BATINI, CARLO. "Automatic graph drawing and readability of diagrams". *IEEE Transactions on Systems, Man, and Cybernetics* 18.1 (1988), 61–79. DOI: 10.1109/21.87055.

[Tou80] TOUSSAINT, GODFRIED T. "The relative neighbourhood graph of a finite planar set". *Pattern recognition* 12.4 (1980), 261–268.

[WCA*17] WU, YANHONG, CAO, NAN, ARCHAMBAULT, DANIEL, et al. "Evaluation of graph sampling: A visualization perspective". *IEEE TVCG* 23.1 (2017), 401–410.

[WEK04] WIESE, ROLAND, EIGLSPERGER, MARKUS, and KAUFMANN, MICHAEL. "yfiles - visualization and automatic layout of graphs". *Graph Drawing Software*. Ed. by JÜNGER, MICHAEL and MUTZEL, PETRA. 2004, 173–191.

[ZZW*15] ZHANG, FANGYAN, ZHANG, SONG, WONG, PAK CHUNG, et al. "A visual and statistical benchmark for graph sampling methods". *Exploring Graphs at Scale Workshop*. 2015.