DOI: 10.1111/cgf.15077

Eurographics Conference on Visualization (EuroVis) 2024
W. Aigner, D. Archambault, and R. Bujack
(Guest Editors)

COMPUTER GRAPHICS *forum*
Volume 43 (2024), Number 3

An Experimental Evaluation of Viewpoint-Based 3D Graph Drawing

S. van Wageningen[1] and T. Mchedlidze[1] and A. Telea[1]

[1]Utrecht University, Department of Information and Computing Sciences, The Netherlands

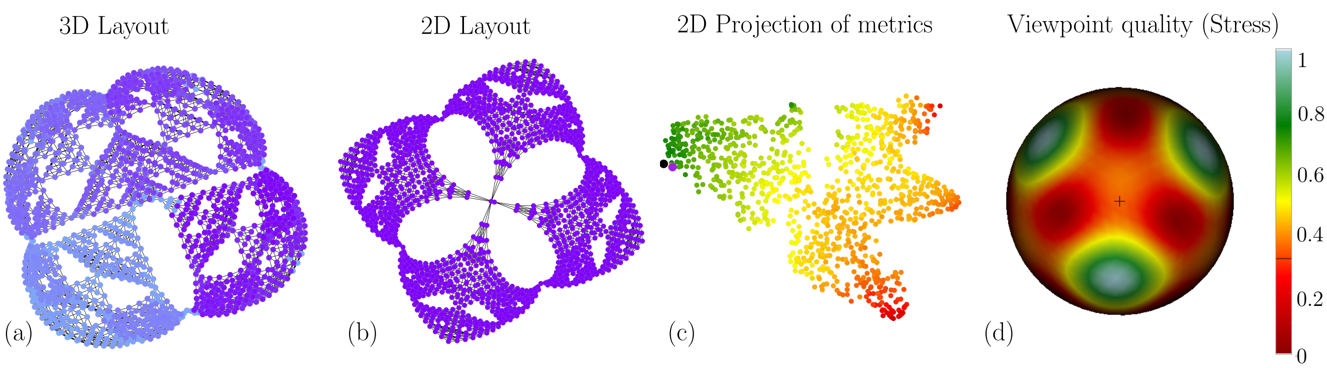

Figure 1: *Four views of our tool used to analyze differences between 3D graph viewpoint layouts and their 2D graph layout counterpart for the same graph. (a) and (b) depict the 3D layout and 2D layout, respectively, (c) portrays the projection scatter plot of the viewpoint metric space, (d) shows the quality metric sphere.*

Abstract

Node-link diagrams are a widely used metaphor for creating visualizations of relational data. Most frequently, such techniques address creating 2D graph drawings, which are easy to use on computer screens and in print. In contrast, 3D node-link graph visualizations are far less used, as they have many known limitations and comparatively few well-understood advantages. A key issue here is that such 3D visualizations require users to select suitable viewpoints. We address this limitation by studying the ability of layout techniques to produce high-quality views of 3D graph drawings. For this, we perform a thorough experimental evaluation, comparing 3D graph drawings, rendered from a covering sampling of all viewpoints, with their 2D counterparts across various state-of-the-art node-link drawing algorithms, graph families, and quality metrics. Our results show that, depending on the graph family, 3D node-link diagrams can contain a many viewpoints that yield 2D visualizations that are of higher quality than those created by directly using 2D node-link diagrams. This not only sheds light on the potential of 3D node-link diagrams but also gives a simple approach to produce high-quality 2D node-link diagrams.

CCS Concepts

• ***Human-centered computing*** $\rightarrow$ ***Graph drawings;***

1. Introduction

Node-link diagrams are one of the most popular methods for visualizing relational data or so-called *graphs*. These graph drawings can aid users in the understanding and interpretation of data. Yet, the quality of the drawings can heavily impact how well users can perform certain tasks and understand the visualized data [PCJ96]. As such, much research has been done for measuring the quality of such drawings via several *quality metrics* that can capture various aspects of goodness of a node-link diagram [Pur02].

Conventionally, node-link diagrams are constructed as static two-dimensional images, due to their ease of use on computer screens and in print. However, our daily experience is rooted in a 3-dimensional world, and this familiarity with 3D spaces, combined with advancements in Virtual Reality (VR) technology, makes exploring 3D data visualizations more intuitive. This naturally leads to an interest in examining how node-link diagrams could benefit from a 3D representation. Although 3D graph drawing has been around for a couple of decades, with the popular Fruchterman-Reingold drawing technique [FR91] being among the first to also provide 3D drawings, it was mostly cast aside for 2D graph drawing. Three-dimensional node-link diagrams are used far less than 2D ones as they have many known limitations and few well-understood advantages. A key issue of 3D visualizations is that they require interactive exploration. More specifically, users need to interact with the visualization to search for suitable viewpoints that *e.g.* provide a clear view of the graph's structure. As such, the availability of high-quality viewpoints is crucial to the effectiveness of 3D node-link diagrams.

The importance of good viewpoints of 3D graph drawings has been acknowledged by previous studies [KK88, PMER97]. There are also a few attempts to find such good viewpoints efficiently [PMER97, HW98]. However, the question whether the state-of-the-art 3D graph layouts contain *many* such good viewpoints has not been addressed in the literature. Furthermore, up to our knowledge, it remains uncertain whether such viewpoints yield visualizations that are perhaps of even better quality than the 2D layouts constructed by 2D versions of the same algorithms. To address the above, we examine the following research questions

RQ1: Do viewpoints acquired from 3D node-link diagrams created by state-of-the-art graph layout algorithms lead to higher quality layouts than their 2D counterpart?

RQ2: In case such viewpoints exists, how much better are they than the 2D counterparts?

In order to answer these questions we rely on a set of nine quality metrics that allow us to measure and compare the goodness of 2D viewpoints of 3D layouts with that of 2D layouts. Separately, we collect quality metrics for a large sampling of viewpoints, graphs, and layout techniques to additionally explore how the space of the metrics varies across the graph classes and layout algorithms. This addresses the following question:

RQ3: Are changes in quality metrics across viewpoints consistent for different graphs and layout techniques?

The paper is structured as follows. Section 2 covers related work. Section 3 describes the overall structure of our experiments in terms of used quality metrics, datasets, and the interactive tool we constructed to analyze our data and build our hypotheses. Section 4 discusses the obtained data and our answers to the research questions. We conclude with future work in Section 6.

2. Related Work

2.1. 3D drawing algorithms

One of the first algorithms capable of drawing straight-line 3D node-link diagrams is the Fruchterman-Reingold algorithm [FR91], although it was primarily developed for 2D drawings. While fast, this algorithm would often converge at a local minimum of its cost function. Davidson and Harel [DH89] pioneered the use of Simulated Annealing to achieve often higher-quality results than force-directed methods. The Simulated Annealing technique, however, had longer computing times. To tackle this limitation, Kosak et al. [KMS94] looked at parallel computing, which was later picked up by Monien et al. [MRS96] who made use of a parallel simulated annealing algorithm to construct 3D graph layouts. Similarly, Cruz et al. [CT96] also developed a Simulated Annealing drawing algorithm for 3D graph layouts.

Bruss and Frick [BF96] developed a fast and interactive 3D layout algorithm, one of the first methods where the user could interactively navigate a 3D layout. Later on, Gajer et al. [GGK04] designed an algorithm that could draw large graphs in any multidimensional space. Additionally, their algorithm was capable of generating graph layouts much faster compared to previous techniques. Similarly, Hu [Hu05] developed an efficient force-directed algorithm that could also draw 3D layouts. More recently, ForceAtlas2 (`FA2`) [JVHB14] has become a widely used state-of-the-art layout algorithm that has also been adapted to work in 3D.

Gansner et al. [GKN05] were one of the first to consider Multidimensional Scaling techniques. These techniques are commonly used to reduce the dimensionality of datasets, creating so called low-dimensional *projections* of the data. Their proposed method named Stress Majorization (`SM`) built upon the work of the stress cost functions of Kamada and Kawai [KK89] and can draw in any number of dimensions. Similarly, PivotMDS [BP07] aimed to draw large graphs by approximating multidimensional scaling. More recently, `tsNET` [KRM*17] adapted the `t-SNE` [MH08] dimensionality reduction technique for graph drawing.

Graphs can also be drawn in other styles in 3D. Clustered graphs for instance can be drawn using a fish-eye viewpoint [AKY05]. Miller et al [MHK22] created 3D graph layouts by plotting coordinates on a sphere rather than a 2D plane, whereas Carriere and Kazman. [CK95] show that hierarchical graph data can be aesthetically visualized using cone trees in 3D.

2.2. Viewpoints in 3D

A key issue with 3D graph layouts is that interaction is required in order to benefit from the information encoded in the third dimension. User studies that focus on the comparison of 2D and 3D graph drawings find only small differences regarding user response times and error rates. However, when stereoscopic depth cues and/or motion are used in 3D, users tend to perform better on path detection [WM08] and community detection tasks [GPK11]. Similarly, McGee et al. [MGM*19] state in their literature survey the importance of these cues, as 3D graph visualizations without stereoscopic/motion cues offer no benefits over 2D. Other studies [FPK*23, JJHS*22] have looked towards using Virtual Reality (VR) to explore the differences between 2D, 2.5D, and 3D network visualizations and found that different tasks were done more effectively using different number of dimensions. Research in other fields show mixed results in the comparison of 2D versus 3D. Aygar et al. [AWR18] highlight that stereoscopic depth cues and/or motion also improve target detection in 3D point cloud visualizations, mirroring the findings of Ware and Mitchell [WM08], and Greffard et al. [GPK11]. Conversely, Borkin et al. [BGP*11] show that 2D tree diagrams are more effective and efficient than 3D tree diagrams in highlighting regions of interest. Marriot et al. [MCH*18] summarize research done on 3D visualizations in the entire field of information visualization. They find that user studies advocate that 3D representations are more favorable for portraying the structure of high-dimensional data, whereas 2D representations are more favorable for accurate data measurements, comparisons and manipulations.

The takeaway from such user studies is the need for *interaction* with a 3D graph drawing in order to find suitable viewpoints. Finding a good viewpoint in a 3D graph drawing is a difficult problem on its own. It is debatable what constitutes a 'good' viewpoint and which quality metrics should be considered to gauge that. Kamada and Kawai [KK88] state that, for a 3D object, a viewpoint is good if it preserves shape information. Eades et al. [PMER97] propose to take occlusion of vertices/edges as an important factor of what constitutes a good viewpoint. Additionally, Eades et al. deliver an algorithm for finding such a viewpoint with a running time of $O((n+m)^4 log(n+m))$, where n and m are the number of nodes and edges, respectively. Given such large running times, Houle and Webber [HW98] propose a few heuristics that approximate a viewpoint in a faster way.

Analogously, Castelein et al. [CTMT23] propose a method of computing and evaluating viewpoints of a 3D projection scatterplot. The set of viewpoints are acquired by taking a uniform covering sampling of viewpoints of the 3D projection. These viewpoint projections are then quantitatively and qualitatively evaluated in order to determine how much better (or worse) they are compared to their direct 2D projection counterparts. They conclude that 3D projections can offer viewpoints with higher quality than their 2D counterparts but the positive findings depend on the metric and dataset. In this paper, we adapt Castelein et al.'s approach to apply it to node-link diagrams. We also provide an interactive tool to allow for in-depth analysis, ease-of-use, and viewpoint quality metric exploration.

2.3. Visual quality evaluation

Since we define a *good viewpoint* as a one which creates a high-quality graph visualization, we next discuss ways to measure the quality of graph drawings. Since graph drawing and projection techniques share many common aspects (see Sec. 2.1), we next also highlight quality measurement techniques in the latter area.

Graph layout quality is commonly assessed by a set of metrics that consider node positions (stress, node resolution), edge lengths (edge length deviation), how edges cross each other (crossing number, crossing resolution), how edges 'spread' around a node they are adjacent to (angular resolution), and the overlap of nodes and/or edges. Several studies discuss how such metrics must be combined to capture well the quality of graph layouts [HEHL13] and how these metrics relate to human perception [PCJ96, Hua07]. We discuss such metrics in detail in Sec. 3.3.

Projections can be measured by a range of quality metrics of which several are similar to graph drawing (*e.g.*, stress, node resolution, and overlap). Espadoto et al. [EMK*19] propose to date the most detailed quantitative survey of projection algorithms. Vernier et al. [VGdS*20, VCT21] propose similar studies for dynamic projections; we follow much of their methodology in dataset, technique, and metric selection and result aggregation and interpretation in our work, while using graph layout algorithms instead of projections. Other recent evaluations considering a different set of quality metrics for projections include [CAS*23, ACS*24].

3. Method

Our first two research questions are concerned with the quality of the viewpoints that state-of-the-art 3D graph drawing layouts produce. To address these questions and evaluate the potential usefulness of the viewpoint-driven approach, we use five popular layout techniques to graphs with different characteristics, and produce 2D and 3D graph layouts, as described in Sections 3.2 and 3.4, respectively. We use nine quality metrics (Section 3.3) to measure the quality of the 2D layouts and the quality of a sample (1000) of viewpoints of the 3D graph layouts. We then compare the obtained quality values to gain insight in whether we can attain higher quality viewpoints of 3D drawings than 2D drawings, and how much better these can be. Additionally, given the amount of collected experimental data, we use the dimensionality reduction technique `t-SNE` [MH08] to shrink the metric space and explore viewpoint quality differences between graphs and techniques.

3.1. Preliminaries

We next introduce a few notations and terms. An *undirected graph* $G(V,E)$ consists of a set of node-set $V = \{v_1, v_2, \ldots, v_n\}$ and an undirected unweighted edge-set $E \subseteq V \times V = \{e_1, e_2, \ldots, e_m\}$. A *graph drawing*, or *layout*, algorithm GL_k assigns k-dimensional coordinates to all nodes in V, where we next consider $k \in \{2,3\}$. The edges are drawn as straight-lines connecting the adjacent nodes. Hence, $GL_k(G) = X$, where $X \in \mathbb{R}^{n \times k}$ is a matrix of $n \times k$ node coordinates. With the above notation, vector $X_i \in \mathbb{R}^{1 \times k}$ contains the coordinates of node v_i. Let $D \in \mathbb{R}^{n \times n}$ denote the *shortest path matrix* of graph-theoretic distances between all node-pairs $\{v_i, v_j\}$ in V. The Euclidean distance $\|X_i - X_j\|$ between two nodes v_i and v_j is referred next as simply *distance*. Let $deg(v)$ be the *degree* of node v, *i.e.*, the number of edges incident to v; and $L(e) = \|X_i - X_j\|$ be the length of the edge $e = (v_i, v_j)$ in the drawing X. Finally, a *quality metric* is a function $Q(X) \in [0,1]$ which assigns a value to the layout X of G, with low (resp. high) values denoting poorer (resp. better) layouts.

From any $k = 3$ dimensional layout $GL_3(G) = X$ of a graph G, we can generate 2D layouts by orthographically projecting X on a plane given by a 3D viewing direction or viewpoint. In our work, we construct a set of viewpoints by uniformly sampling points on a sphere enclosing the 3D layout $GL_3(G)$ using the spherical Fibonacci lattice method [Gon10]. We refer next to these 2D 'projected layouts' as *viewpoint layouts*. In contrast, we call the 2D layout $GL_2(G)$ directly computed from G using the same technique GL the *viewpoint's 2D counterpart*.

3.2. Layout Techniques

In our experiments, we use five techniques to produce 2D and 3D layouts. Our choice is based on the popularity of the techniques and their ability to produce layouts in both 2D and 3D. All techniques are iterative, set to run for 300 iterations. We use default parameter values, as described in the supplementary material, unless otherwise specified. The five used graph drawing techniques are as follows: ForceAtlas2 `(FA2)` [JVHB14], Stress Majorization `(SM)` [GKN05], `PivotMDS` [BP07], `tsNET` [KRM*17] and `tsNET*` [KRM*17]. For `FA2`, `SM`, `tsNET` and `tsNET*` we use their existing Python implementations [Chi19], [VGO*20] and [Kru17], respectively. For `FA2`, we adapt the code to allow for 3D graph drawing. We implement `PivotMDS` from scratch based on the original paper. The number of pivots chosen in `Pivot-MDS` is set to $max(5, n/10)$; the perplexities for `tsNET`/`tsNET*` are handpicked based on multiple experiments, as provided in the supplementary material.

3.3. Quality Metrics

We choose several quality metrics to gauge the quality of 2D and viewpoint layouts. Our choice follows the metrics' popularity in previous work, their usefulness in assessing the quality of viewpoints of 3D layouts [PMER97], and user studies confirming their ability to gauge the actual quality of layouts as perceived by humans [PCJ96, Hua07]. Each metric ranges from 0 to 1, with 0 and 1 being the worst and best acquirable result, respectively.

Stress This metric [KK89] measures how much all node-pair Euclidean distances in a drawing deviate from their shortest path distances as

$$\mathtt{ST} = 1 - \frac{1}{n(n-1)/2} \sum_{i<j}^{n} \frac{(\|Z_i - Z_j\| - d_{ij})^2}{d_{ij}^2}, \tag{1}$$

where Z_i scales the coordinate X_i by the shortest path distances as

$$Z_i = \frac{\sum_{i \neq j} \|X_i - X_j\| / d_{ij}}{\sum_{i \neq j} \|X_i - X_j\|^2 / d_{ij}^2} X_i. \tag{2}$$

Edge length deviation It is known that users favor consistent edge lengths in a drawing [CP96]. We thus compute an *edge length deviation* to gauge the average deviation of edge lengths to the mean μ of all edge lengths in a layout as

$$\mathtt{ELD} = 1 - \sqrt{\frac{1}{m} \sum_{i=1}^{m} (L(e_i) - \mu)^2}. \tag{3}$$

Node resolution Ideally, nodes should be placed far enough apart to distinguish them. The *node resolution* metric [ALD*22] measures this by taking the smallest node-pair distance over the largest node-pair distance

$$\mathtt{NR} = \frac{\min_{1 \leq i,j \leq n} \|X_i - X_j\|}{\max_{1 \leq i,j \leq n} \|X_i - X_j\|}. \tag{4}$$

Angular resolution Small angles between consecutive adjacent edges of a node in a layout tend to make it harder to identify edges [CP96, Hua07]. To measure how often small angles occur in a graph drawing, we compute the average deviation of angles of adjacent edges *vs* the best possible angle as

$$\mathtt{AR} = 1 - \frac{1}{n'} \sum_{i=1}^{n'} \left| \frac{\Theta_i - \theta_i}{\Theta_i} \right|, \tag{5}$$

where n' is the number of nodes with 2 or more incident edges; θ_i is the smallest measured angle between consecutive edges of node i; and $\Theta_i = 2\pi / deg(v_i)$ is the best possible angle between consecutive edges for the node i.

Crossing number Purchase et al. [PCJ96] show that the number of crossings affects how well a drawing can be understood. We measure the number of crossings c_{cnt} normalized by its maximum possible value c_{poss} by

$$\mathtt{CN} = 1 - \frac{C_{\text{cnt}}}{C_{\text{poss}}}, \tag{6}$$

where the maximum possible number of crossings C_{poss} is given by

$$C_{\text{poss}} = \frac{m(m-1)}{2} - \frac{1}{2} \sum_{i=1}^{n} (deg(v_i)(deg(v_i) - 1)). \tag{7}$$

Crossing resolution Next to minimizing the crossing number, the angles of edge crossings should also be maximized [HHE08], in order to ease the perception of edges. The *crossing resolution* metric computes the average deviation of crossing angles to the right-angle optimum

$$\mathtt{CR} = \frac{1}{C_{\text{cnt}}} \sum_{i=1}^{C_{\text{cnt}}} \frac{\alpha_i}{\pi/2}. \tag{8}$$

Here, $\alpha_i, i = 1, \ldots, C_{\text{cnt}}$ are all the crossing angles in the layout.

Node-node occlusion Eades et al. [EHW97] propose various occlusion metrics to capture how good or bad a viewpoint can be. The *node-node occlusion* concerns itself with how often two nodes perfectly overlap. We slightly modify this metric, to make it act in a continuous fashion, by measuring the area of the node-node occlusion as

$$\mathtt{NN} = 1 - \frac{1}{n(n-1)/2} \sum_{i<j}^{n} A(v_i, v_j), \tag{9}$$

In this and the following two metrics we use notation $A(\cdot,\cdot)$ to depict the intersection between the geometric objects that the arguments of the function represent. These are either a circle of a radius r in case of a node, or a thin rectangle of width w in case of an edge. Based on initial experiments, we set $r = min(1/\sqrt{n}, 1/150)$ and $w = r/5$.

Node-edge occlusion We use the node-edge occlusion metric [EHW97] to measure the amount of node-edge overlaps as

$$\mathtt{NE} = 1 - \frac{1}{nm} \sum_{i}^{n} \sum_{j}^{m} A(v_i, e_j). \tag{10}$$

Edge-edge occlusion While the crossing number measures how many crossings are in a layout, *edge-edge occlusion* [EHW97] measures the amount of overlap of edge crossings as

$$\mathtt{EE} = 1 - \frac{1}{m(m-1)/2} \sum_{i<j}^{m} A(e_i, e_j). \tag{11}$$

Combinations Often the optimization of multiple quality metrics is key to producing good layouts [HEHL13]. Therefore to measure the quality of a layout in a global manner, we define the `ALL` metric as the linear combination of all nine aforementioned metrics with equal weights. However, not every metric is equally important. Since

Graph	*n*	*m*	$deg(V)_{avg}$	*source*
mesh1em6	48	129	5.38	[DH11]
grid	54	93	3.44	generated [vWMT23]
gridaug	54	98	3.63	generated [vWMT23]
GD96_c	65	125	3.85	[DH11]
grid1_dual	224	420	3.75	[DH11]
mesh3em5	289	800	5.54	[DH11]
L	956	1820	3.81	[DH11]
stufe	1036	1868	3.61	[DH11]
Rome Graphs	[32, 105]	[35, 141]	(2.11, 2.82)	[GDT]
Graphs tsNET	[72, 4941]	[75, 13722]	(2.00, 27.70)	[KRM*17]

Table 1: *Descriptive statistics of graphs used in the experiments.*

the number of crossing has been experimentally proven to be an important metric for user preference and task performance [PCJ96], and based on the anecdotal evidence that low-stress drawings are preferred by humans, we also use in our evaluations the metric `ST+CN` equal to the linear combination of stress (`ST`) and crossing number (`CN`).

3.4. Dataset

We measure our metrics on a set $\mathcal{G}$ of 51 graphs acquired from different sources. Table 1 displays descriptive statistics about these graphs, as well as their origin. We handpick a set of graphs from the *SuiteSparse Matrix Collection* [DH11], as these graphs tend to have recognizable 3D structures. The graphs used by Kruiger et al. [KRM*17] vary in their size, density, and structure, which is why we have included these in our dataset. Lastly, we reuse the *Rome* [GDT] graphs dataset as it has widely been used as a benchmark for novel 2D drawing algorithms [WYHS21, GLA*21].

3.5. Tool

We adapt the tool from Castelein et al. [CTMT23] to work for graph data (and graph quality metrics) and also extend its functionality. The updated tool consists of seven distinct widgets, each with their own function. Figure 1 shows four of these widgets. Figures 1a and 1b depict a graph's 3D layout and 2D layout, respectively. For each viewpoint layout we compute nine quality metrics, we repeat that process for all 1000 viewpoints and their 2D counterpart ($GL_2(G)$). To visualize the metric space of this 1000×9 dataset we create a projection scatterplot, as visualized in Figure 1c. Here, points are colored according to the `ALL` quality metric; the large black and purple points top-left in the scatterplot show the quality of the 2D counterpart and the currently selected viewpoint in the 3D layout, respectively. The fourth displayed widget is the quality metric sphere, as seen in Figure 1d. Here a viewpoint on the sphere is colored according to the quality metric of that viewpoint's layout, which in case of Figure 1d is stress (`ST`). The other three widgets, not visualized here, allow users to switch between metrics and techniques, to set weighted metric combinations, as well as provide information on the metric distributions and metric comparisons. We implemented our tool in Python 3.9 with the core functionality coming from *PyQtGraph* [pyq23]. Demonstrations of the tool, along with the code can be found on GitHub.

4. Results

In this section we describe our results and how these answer our research questions stated in Section 1.

		SM	FA2	PivotMDS	tsNET	tsNET*	ALL
ST	*median*	0.0	20.9	10.0*	33.3	16.6	13.0
	max	76.1	78.4	96.2	100.0	100.0	100.0
CR	*median*	23.3	33.0	28.8	14.8	10.5	23.9
	max	99.9	100.0	99.8	100.0	100.0	100.0
AR	*median*	61.2	59.3	42.7*	2.6*	1.8*	31.1
	max	100.0	100.0	100.0	100.0	100.0	100.0
NR	*median*	8.5*	68.8	40.9*	0.0*	0.0*	8.5
	max	74.6	100.0	100.0	95.5	99.8	100.0
NN	*median*	31.5*	66.2	46.2*	28.5*	17.0*	45.6
	max	100.0	100.0	100.0	100.0	100.0	100.0
NE	*median*	11.0*	36.9	30.3	2.7*	0.7*	11.7
	max	95.8	100.0	99.9	99.5	100.0	100.0
CN	*median*	23.3	19.9	35.5	1.4*	1.0*	9.3
	max	100.0	100.0	100.0	99.6	67.1	100.0
EE	*median*	36.8*	69.1	55.7*	1.0*	2.4*	31.1
	max	100.0	100.0	100.0	100.0	100.0	100.0
ELD	*median*	82.1*	67.8*	20.4*	99.1	98.5	68.3
	max	100.0	100.0	99.2	100.0	100.0	100.0
ST+CN	*median*	11.7*	20.4	22.8	17.4*	8.8*	16.2
	max	100.0	100.0	100.0	100.0	100.0	100.0
ALL	*median*	30.9*	49.1	34.5*	20.4*	16.5*	30.3
	max	100.0	100.0	100.0	100.0	100.0	100.0

Table 2: *Percentages of viewpoints that are better than their 2D counterpart, aggregated over all graphs. Cells with an * indicate that the highlighted blue cell with the largest median has significant better results.*

4.1. RQ1: Do viewpoints lead to better layouts?

To answer RQ1, we explore the quantitative results in Table 2, Figure 2 and Figure 3. These show the percentage of viewpoints that are better than their 2D counterpart. In particular, for each graph technique GL, each quality metric Q and each graph G, we count the number $N^Q_{GL}(G)$ of viewpoints of $GL_3(G)$ that are better than $GL_2(G)$ with respect to Q. Table 2 shows the median and maximum values of the percentage of viewpoints that are better than $GL_2(G)$, aggregated over all graphs $\mathcal{G}$ used in the experiment, i.e the values $\text{median}\{N^Q_{GL}(G)/1000 : G \in \mathcal{G}\}$ and $\max\{N^Q_{GL}(G)/1000 : G \in \mathcal{G}\}$. Additionally, for each metric we highlight the largest median value of all techniques. The techniques for which the highlighted cell scores significantly better, according to a paired Wilcoxon signed rank test with $\alpha = 0.05$, are annotated with an *.

A large portion of the metrics and techniques in Table 2 has maximum values (close to) 100%, indicating that for at least one graph we can find a 3D layout for which nearly all viewpoints score better than the 2D layout. Median values provide additional insights. For instance, the median value of *stress* (`ST`) for `tsNET` is equal to 33.3, which indicates that for the graph that falls at the median value, 33.3% of the viewpoints are superior than $GL_2(G)$ w.r.t. `ST`. Additionally, the star at `PivotMDS` indicates that `tsNET` benefits significantly more from the viewpoint perspective w.r.t. `ST` than `PivotMDS`.

In general, we see that median values range between the values 0.0% for `ST` of `SM` and 99.1% for `ELD` of `tsNET`. These extremes indicate that dependent on the layout technique and quality metric, various but generally high percentages of viewpoints can have higher quality than the 2D counterparts; this answers RQ1.

To consider how much each individual algorithm benefits from the viewpoint-driven approach, we observe the columns of the table. We note that the column `FA2` contains the most blue cells, in particular for the metrics `CR, NR, NN, NE, EE, ALL`. Hence, this algorithm benefits the most from the viewpoint-approach over other algorithms.

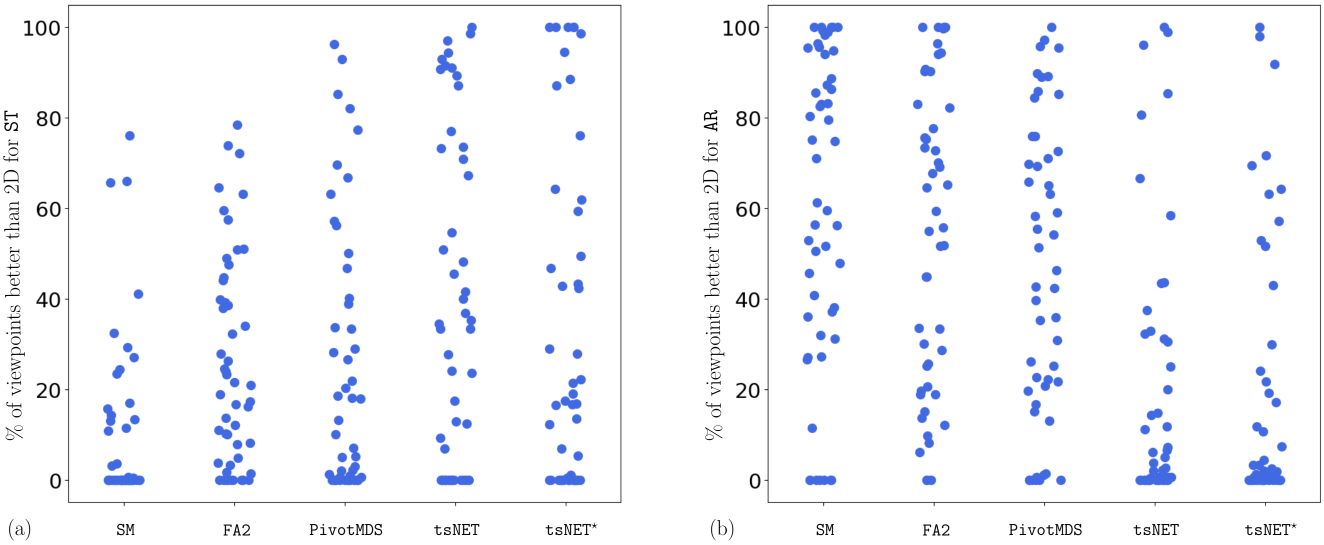

Figure 2: *Jitterplots of the percentages of viewpoints that are better than their 2D counterpart w.r.t. (a)* stress *(*`ST`*) and (b)* angular resolution *(*`AR`*), each point represents a single graph.*

Finally, when we aggregate over all techniques, we see that the *Edge Length Deviation* `ELD` metric gains the most from a viewpoint-driven approach, with a median of 68.3%. All other metrics still benefit from the approach but less compared to `ELD`, since their medians are between 20.4% and 99.1%. When considering the combination `ST+CN`, we see that `PivotMDS` and `FA2` profit the most from the viewpoint-driven approach.

Figure 2 and Figure 3 give a more detailed view on the data in Table 2. Here, each point represents a single graph and the y-axis indicates the percentage of viewpoints that score better than

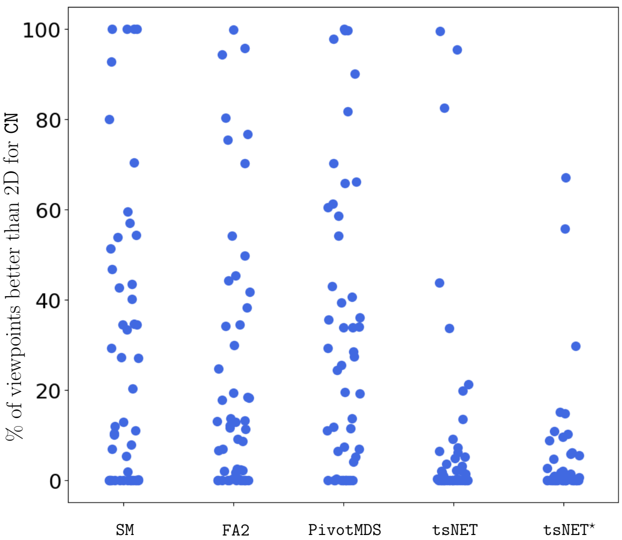

Figure 3: *Jitterplots of the percentages of viewpoints that are better than their 2D counterpart w.r.t.* crossing number *(*`CN`*), each point represents a single graph.*

$GL_2(G)$. Thus, for the quality metric Q and technique GL (shown along x-axis), we observe the raw values $\{N^Q_{GL}(G)/1000 : G \in \mathcal{G}\}$. These plots clearly show the differences between various graphs in our dataset. For instance, in Figure 2a, we see why `SM` has a median of 0 for `ST` – this is expected since `SM` directly optimizes stress, although there is still a minor sample of graphs for which viewpoints can be found with larger stress values. When it comes to the other techniques, a large portion of graphs are presented by points above 20%, with only handful of graphs in `tsNET` and `tsNET`* reaching 100%. We also observe that for every technique there is a considerable number of graphs that does not have better viewpoints with respect to `ST`, `CN`, `AR`, which also holds for all the other metrics, as can be seen in the supplementary material.

Figure 2b shows that the angular resolution metric `AR` is typically higher in viewpoints from graph drawings produced by `SM`, `FA2` and `PivotMDS` as compared to `tsNET` and `tsNET`*. Finally, Figure 3 shows similar results for the crossing number metric `CN`.

4.2. RQ2: How much better are the 3D viewpoints?

So far, we have shown that using a viewpoint-driven approach can lead to better graph layouts compared to 2D ones. We now study how *much* better these viewpoints can be. Since our dataset is comprised of graphs with very different characteristics, we first normalize the quality metric values of the viewpoints and 2D graph layouts as follows.

Let $GL_3(G,i)$ denote the viewpoint layout of 3D drawing constructed by technique GL that is seen from viewpoint i. We consider the set $\mathcal{P}(G) = \{\{GL_3(G,i)\} : GL = \texttt{SM,FA2,PivotMDS,tsNET,tsNET}^*, 1 \leq i \leq 1000\}$ of all the viewpoint layouts, and set $\mathcal{L}(G) = \{\{GL_2(G)\} : GL = \texttt{SM,FA2,PivotMDS,tsNET,tsNET}^*\}$ of all the 2D layouts. Thus $\mathcal{P}(G) \cup \mathcal{L}(G)$ is a set of 5005 layouts of graph G. For

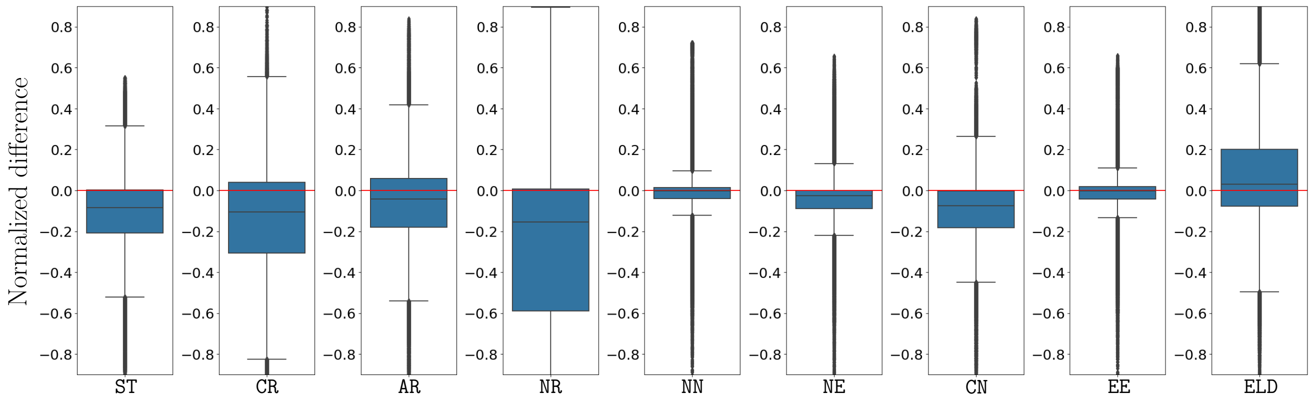

Figure 4: *Boxplots showing the normalized distributions of quality metrics of 3D viewpoints relative to their 2D counterpart, aggregated over all viewpoints, all graphs and layout techniques.*

a quality metric Q, let $Q(G) = \{Q(X) : X \in \mathcal{P}(G) \cup \mathcal{L}(G)\}$ be the set of all possible values the quality metric Q takes for the graph G. We normalize these values so that the smallest is zero and the largest is one, in order to make the values of the quality metrics comparable across graphs. Therefore, we set $Q'(X) = (Q(X) - \min Q(G))/(\max Q(G) - \min Q(G))$ for every layout $X \in \mathcal{L}(G)$. We repeat this for every quality metric Q. In order to understand how the quality metric values of the viewpoints compare to their 2D counterparts we consider the *normalized differences*: $\{Q'(X) - Q'(Y) : G \in \mathcal{C}, X \in \mathcal{P}(G), Y \text{is the 2D counterpart of } X\}$. Figure 4 shows boxplots for these normalized differences for each quality metric Q.

Figure 4 shows that for most metrics, most values and the median lines, lie below 0, telling that most viewpoint layouts score worse than the 2D layouts. The ELD metric is an exception here – for it, there are fewer viewpoints below 0 than viewpoints above 0. This tells that, on average, we can acquire viewpoints with better ELD values than for 2D graph layouts.

		SM	FA2	PivotMDS	tsNET	tsNET*	ALL
ST	*median*	-0.139*	-0.080*	-0.137*	-0.046	-0.073	-0.095
	max	0.120	0.552	0.505	0.485	0.534	0.552
CR	*median*	-0.159	-0.069	-0.171*	-0.164	-0.201	-0.153
	max	0.895	0.869	0.880	0.848	0.789	0.895
AR	*median*	0.026	0.031	-0.030*	-0.152*	-0.160*	-0.057
	max	0.840	0.611	0.456	0.332	0.357	0.840
NR	*median*	-0.267*	0.025	-0.011*	-0.573*	-0.555*	-0.276
	max	0.571	0.944	0.887	0.763	0.988	0.988
NN	*median*	-0.014	0.024	0.046	-0.005	-0.009	0.008
	max	0.342	0.402	0.722	0.434	0.370	0.722
NE	*median*	-0.046	-0.026	0.005	-0.044	-0.053	-0.033
	max	0.189	0.317	0.657	0.219	0.221	0.657
CN	*median*	-0.060	-0.029	-0.063	-0.141*	-0.148*	-0.088
	max	0.840	0.339	0.486	0.171	0.171	0.840
EE	*median*	-0.005	0.028	0.033	-0.023*	-0.028*	0.001
	max	0.395	0.327	0.659	0.161	0.165	0.659
ELD	*median*	0.079*	0.045*	-0.088*	0.175	0.154	0.073
	max	0.932	0.664	0.501	0.787	0.890	0.932
ST+ CN	*median*	-0.099*	-0.054	-0.100*	-0.093*	-0.111*	-0.091
	max	0.840	0.552	0.505	0.485	0.534	0.840
ALL	*median*	-0.065*	-0.006	-0.046*	-0.108*	-0.119*	-0.069
	max	0.932	0.944	0.887	0.848	0.988	0.988

Table 3: *Statistics of quality metrics of viewpoints relative to their 2D counterpart, scaled locally w.r.t. graphs, aggregated over all viewpoints from all graphs. Cells with an * indicate that the highlighted blue cell with the largest median has significant better results.*

Table 3 dives deeper into the specifics of the normalized differences across techniques and gives exact median and maximum values. Over all techniques and metrics, only a handful have median scores larger than 0. For instance, the -0.065 median value of all metrics ALL for SM tells us that 50% of the viewpoints, aggregated over all graphs, score 0.065 lower than $GL_2(G)$. Supporting our previous findings, we also see significant differences across the various techniques. According to the median values, FA2 benefits from the viewpoint-driven approach for five out of nice metrics: AR, NR, NN, EE and ELD, since the median values are positive. The most notable increase in quality metric score can be seen for the ELD. When using the tsNET technique, 50% of the viewpoints have values 0.175 larger compared to $GL_2(G)$.

The maximum values in Table 3 allow us to see how much better the viewpoint approach can be. Note that due to normalization the maximum values belong to $(-1, 1)$. Overall the maximum increases for all quality metrics are all rather large, between 0.552 for ST and 0.988 for NR. This means that for the graphs used in our dataset there are viewpoints that have quality values that are substantially larger than $GL_2(G)$. An intriguing finding is that the maximum seen for stress (ST) is 0.120 for the stress majorization algorithm (SM). Even though this value is small, it is still surprising that optimizing stress in 3D and then looking over many viewpoints can provide a better stress than directly optimizing stress in 2D. To conclude, the maxima values have shown that viewpoints can acquire drastically better results, depending on the metric, technique and graph.

4.3. RQ3: Are quality metric changes consistent across graphs and layout techniques?

One of the widgets in our tool, the *quality metric sphere*, as visualized in Figure 1d and Figure 5, allows us to observe how the values of a quality metric change when we view the graph from various viewpoints. Each point p on the sphere is colored according to the value of the quality metric of the viewpoint layout observed from point p. We observe that the changes in the values of all individual quality metrics for all graphs and all layout techniques are smooth.

However, we also observed that the coloring of the spheres differed over graphs and quality metrics, with the number of local minima varying. This observation brings us to our third research question of what the quality metric space looks like.

Given the amount of experimental data (51 graphs $\times$ 5 techniques $\times$ 1000 viewpoints $\times$ 9 quality metrics), we use `t-SNE` to gain more insight in the differences across graphs and techniques. In the supplementary material we highlight the differences between individual metrics. Figure 6 shows the projections of a sample of graphs for the five layout techniques. We consider each of the 1000 3D viewpoints, corresponding viewpoint layouts with their 9 quality metrics as a 9-dimensional data set with 1000 data points. Each scatterplot then shows the projection of such a dataset in 2D. In order to help the interpretation of the scatterplot, we have selected a few (boxed scatterplots) and for each of those we display the 2D counterpart, and the layouts corresponding to the best/worst viewpoints. The points here are colored according to the values of the `ALL` metric.

Figure 6 shows that the type of graph heavily influences the shape of the projections. For instance, distinct clusters are commonly spotted in the projection plots of grid graphs using the techniques `SM`, `FA2` and `PivotMDS`. The colors of these clusters further indicate that, for these types of graphs, there are massive differences between good and bad viewpoints. These differences can be explained by the fact that when viewing a flat grid in 3D, we can encounter viewpoints that have very poor quality - many overlaps and intersections. At the same time, grids such as these allow some 'perfect' quality viewpoints. We do not observe separate clusters for grids and `tsNET` and `tsNET`* and sometimes other techniques, as in these cases the methods stronger utilize the 3D space by 'curling up' the grids.

What remains relatively consistent across techniques and graphs, is the coloring of the points. The projection technique `t-SNE` accurately puts similar points together, visible by the fact that points with similar average quality metric scores stay close. In a majority of the plots, there is often a fine gradient visible, with only the occasional outliers present. The supplementary material provides some additional discussion on which metrics are the main factors for creating finer gradients and distinct clusters.

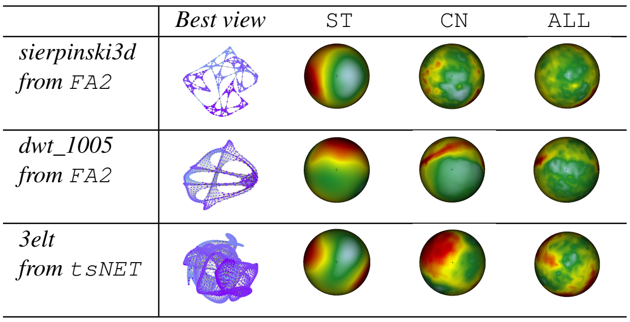

Figure 5: *The best viewpoint layout for three different graphs. The colors in the spheres correspond to the metric values of the given viewpoint layouts, see Figure 1d for the color mapping.*

5. Discussion

By analyzing the quality of the 3D viewpoints, and the extent by which they are better that their 2D counterparts, we see that there are various numbers of viewpoints that are better than their 2D counterpart (dependent on the quality metric and technique). We also observe that, even though `SM` is a method that optimizes stress, it does not always achieve optimal stress values. As it turns out, better-stress results can be found among viewpoints of 3D layouts. Similar effects were observed with stress values of 3D *vs* 2D projection scatterplots [TZvS*21]. For all evaluated layout techniques, we speculate that viewpoint layouts can score higher than their 2D counterpart as a result of the optimization process in 3D having more degrees of freedom than in 2D. As a consequence, we conjecture that by optimizing layouts in 3D one can obtain results that better reveal the overall graph structure, due to the aforementioned increase in the degrees of freedom. From a practical perspective, our results show the potential for using 3D viewpoints to create 2D layouts of high quality.

Additionally, we observe that all the studied quality metrics change *smoothly* as function of viewpoint (Sec. 4.3). This is, technically, not surprising given the fact that the 3D-to-2D projection transform is a continuous function of its viewpoint. Small viewpoint changes yield small changes in the 2D projection; if, in turn, we consider that the involved 2D quality metrics are also smooth changing functions of the 2D layout they measure, we obtain the mentioned observation. This observation has several practical implications. First, we could imagine automatic methods for finding the highest-quality (or a high-enough quality given a user-specified threshold) viewpoint of a 3D layout, *e.g.*, by using gradient ascent methods on the quality sphere. Alternatively, we could perform this search in the simpler 2D space provided by the `t-SNE` projection (see Fig. 6). Finally, we could enhance this projection view to explore additional questions, *e.g.*, how are the viewpoints of the highest, respectively lowest, quality placed with respect to each other in 3D; and how spread-out, or concentrated, over the viewpoint space (sphere) are viewpoints of similar quality. Getting such insights could help next devising automatic algorithms that efficiently find high(est) quality viewpoints without the need to exhaustively search all viewpoint samples.

5.1. Limitations

We next outline a few limitations of our study.

Parameters Even though the viewpoint-driven approach can deliver high-quality results, several limitations exist. Most notably, layout techniques depend on various so-called hyperparameters, *e.g.*, the number of iterations for certain layout techniques, such as `tsNET`$^{(\star)}$. Changing these can alter the observed results. To solve this, one can next perform a comprehensive grid search that considers the optimization of such parameters, much as done by Espadoto et al. [EMK*19] for multidimensional projections. In addition, certain quality metrics have hardcoded parameters that can influence the range of the results. Our normalization of the quality metrics should partially address this issue. We show the varying ranges of quality metrics in more detail in the supplementary material.

Samples Finding the actual highest-quality viewpoint is, in our

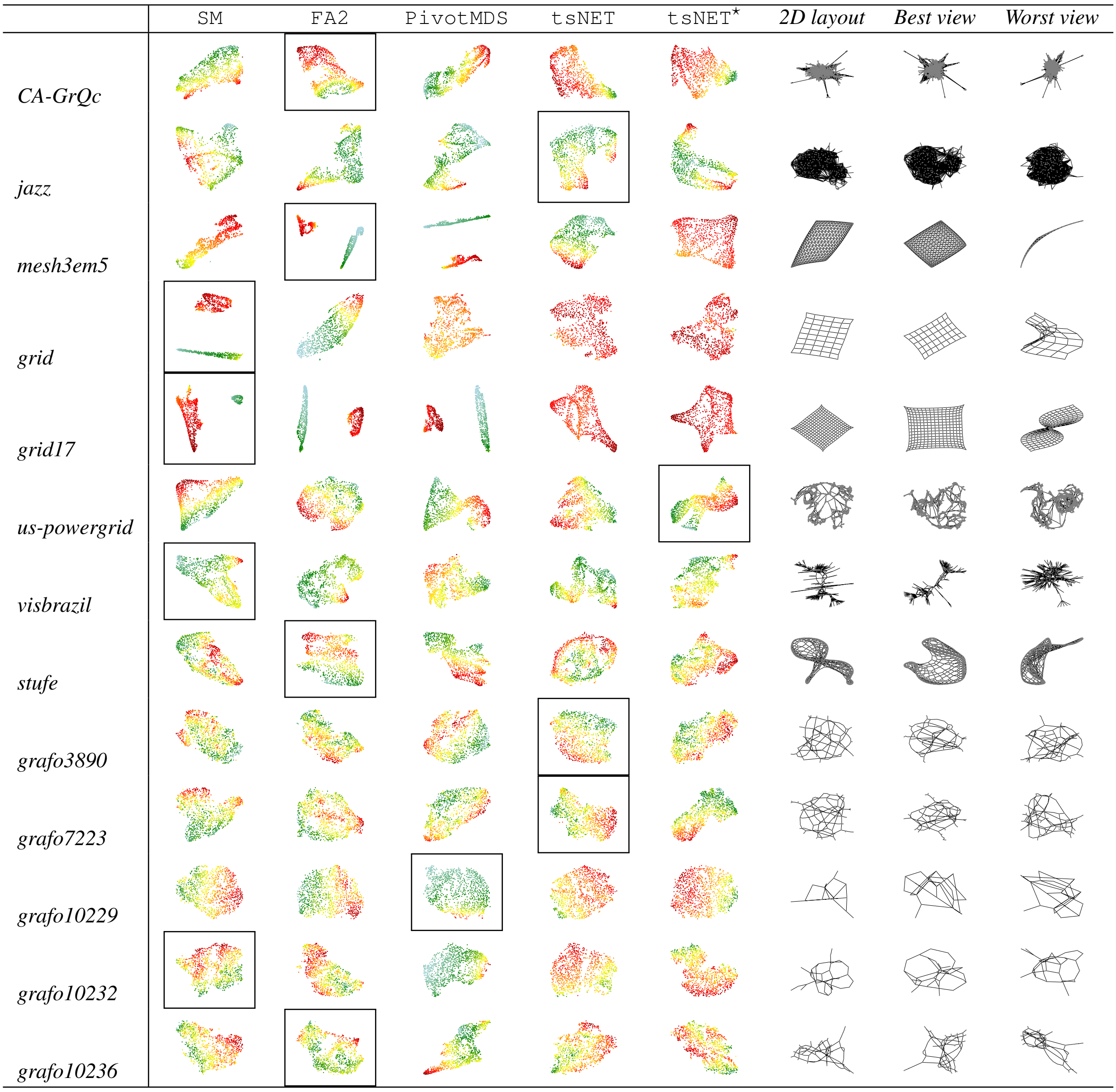

Figure 6: *Projections of quality metrics of viewpoints for selected graphs. Points are colored based on the* ALL *quality metric using the same color mapping from Figure 1d. For the highlighted projection plot the corresponding 2D graph layout is given, as well as the best and worst viewpoint layouts.*

approach, influenced by the sampling resolution of the viewpoint space. To limit computational effort, we use a set of 1000 viewpoints based on the work of Castelein et al. [CTMT23]. Given the observed smooth behavior of the quality metric (see Sec. 4.3 and discussion earlier in this section), it is improbable that significantly higher (or lower) metric values could exist for additional sample points located between the considered ones. As such, while we do not compute the *exact* maximum-quality viewpoints, our sampling approach should yield values quite close to it. Still, it is interesting – from a theoretical viewpoint – to find ways by which we can determine this exact maximum without recurring to a denser, thus computationally more expensive, sampling.

Datasets The graphs chosen in our evaluation may not be diverse enough to capture how 3D metrics compare to their 2D counterparts for *any* type of graph. For example, in our considered dataset, only the graphs inherited from tsNET work [KRM*17] differ substantially with respect to density. Expanding this dataset with more graphs with different characteristics could lead to different results.

Detailed study Our study concerned itself with finding properties that hold generally for a collection of graphs, as our key aim was to argue that specific viewpoints of 3D drawings can be better than their 2D counterparts. It is also interesting to perform a different study, namely examine in detail how these qualities (3D *vs* 2D) differ for a specific graph, by expanding the analyses presented in Sec. 4.3. This could lead to subtle quality-related insights which, in turn, can lead to the construction of better-quality layout algorithms.

5.2. Future Work

Based on the above, we see the following possible extensions of our work.

Study of the tool While using our tool we found ourselves being "in the flow" [Csi91], experiencing motivation, immersion and concentration on the process of discovery of even more readable viewpoints. The tool's widgets reinforced this state by providing high-level guidance. We conjecture that being immersed in this experience, the user is able to acquire a very good overview of the graph's structure and perhaps memorize it. We are interested in understanding whether the widgets of the tool are helpful in navigating the high-quality viewpoints. Except for the abstract setting, we plan to perform such an evaluation in an appropriate application domain.

User evaluation of 3D viewpoints We have seen that, dependent on the graph and layout technique, the 3D layouts provide viewpoints whose 2D projections yield 2D layouts of even higher quality than 'native' 2D layouts. This assessment, however, is purely based on automatically computed quality metrics. We plan to evaluate how the high-quality 3D viewpoints and their 2D counterparts compare with respect to task performance. Finally, we would also like to know whether the best viewpoints according to the metrics are also those that the humans would select themselves. Such a user-study would also provide insight in the presence or absence of correlations between high-quality layouts (w.r.t. metrics) and user-favored layouts.

Quality metrics for 3D layouts In our work we evaluate viewpoint layouts as a proxy to assess the quality of a 3D graph drawing. We speculate that this is a promising research direction with the goal to develop quality metrics for 3D graph layouts, which, to our knowledge, has not been studied extensively in the literature.

Finding good viewpoints In the future we would also like to explore the possibility of finding a graph's best viewpoint algorithmically or automatically (from the perspective of specific metrics). Specifically, we aim to find an efficient method for producing, for any arbitrary graph and its 3D drawing, the best possible viewpoint layout, without the need to iterate through a large set of metric sample values. This, coupled with our current findings, will lead to layout methods that produce 2D graph drawings which are better, quality-wise, than directly laying out the graphs in 2D, with no other algorithm modification needed. We speculate that by using Graph Neural Networks, such as DeepGD [WYHS21], we may be able to train a model to accurately predict the best viewpoint of any given graph with its corresponding 3D layout.

6. Conclusion

We have presented an experimental evaluation in which we explore the ability of state-of-the-art layout techniques to produce high-quality views of 3D graph drawings. Through the use of a covering sampling of viewpoints, we compare viewpoint layouts with their 2D counterparts, and asses their quality via nine different quality metrics. Our experiments show that acquiring high quality viewpoint layouts is possible for many graphs, with certain techniques such as `FA2` benefiting the most from this viewpoint-driven approach. The same conclusion is also made for the size of the effect, where the best viewpoint can be drastically better than its 2D counterpart. We find that the size of the increase in quality of viewpoint layouts is also dependent on the quality metric. With certain quality metrics such as `ST`, on average, acquiring the poorest viewpoint results, and the `ELD`, on average, acquiring the best viewpoint results. Moreover, we explored the metric space of the quality metrics and techniques through the use of dimensionality reduction. We found that the characteristics of the graph can immensely influence the metric space of possible viewpoints.

References

[ACS*24] ATZBERGER D., CECH T., SCHEIBEL W., TRAPP M., RICHTER R., DÖLLNER J., SCHRECK T.: Large-scale evaluation of topic models and dimensionality reduction methods for 2D text spatialization. *IEEE TVCG 30*, 1 (2024). doi:10.1109/TVCG.2023.3326569. 3

[AKY05] ABELLO J., KOBOUROV S. G., YUSUFOV R.: Visualizing large graphs with compound-fisheye views and treemaps. In *Graph Drawing* (Berlin, Heidelberg, 2005), Pach J., (Ed.), vol. 3383 of *LNCS*, Springer Berlin Heidelberg, pp. 431–441. doi:10.1007/978-3-540-31843-9_44. 2

[ALD*22] AHMED R., LUCA F. D., DEVKOTA S., KOBOUROV S., LI M.: Multicriteria scalable graph drawing via stochastic gradient descent, $(SGD)^2$. *IEEE Transactions on Visualization and Computer Graphics 28*, 06 (jun 2022), 2388–2399. doi:10.1109/TVCG.2022.3155564. 4

[AWR18] AYGAR E., WARE C., ROGERS D.: The contribution of stereoscopic and motion depth cues to the perception of structures in 3D point clouds. *ACM Trans. Appl. Percept. 15*, 2 (feb 2018). doi:10.1145/3147914. 2

[BF96] BRUSS I., FRICK A.: Fast interactive 3-D graph visualization. In *Graph Drawing* (Berlin, Heidelberg, 1996), Brandenburg F. J., (Ed.), vol. 1027 of *LNCS*, Springer Berlin Heidelberg, p. 99–110. doi:10.1007/BFB0021794. 2

[BGP*11] BORKIN M., GAJOS K., PETERS A., MITSOURAS D., MELCHIONNA S., RYBICKI F., FELDMAN C., PFISTER H.: Evaluation of artery visualizations for heart disease diagnosis. *IEEE transactions on visualization and computer graphics 17* (12 2011), 2479–88. doi:10.1109/TVCG.2011.192. 2

[BP07] BRANDES U., PICH C.: Eigensolver methods for progressive multidimensional scaling of large data. In *Graph Drawing* (Berlin, Heidelberg, 2007), vol. 4372 of *LNCS*, Springer Berlin Heidelberg, p. 42–53. doi:10.1007/978-3-540-70904-6_6. 2, 3

[CAS*23] CECH T., ATZBERGER D., SCHEIBEL W., RICHTER R., DÖLLNER J.: Evaluating architectures and hyperparameters of self-supervised network projections. In *Proc. IVAPP* (2023), SciTePress, pp. 187–194. doi:10.5220/0011699700003417. 3

[Chi19] CHIPPADA B.: ForceAtlas2 Python. URL: https://github.com/bhargavchippada/forceatlas2. 3

[CK95] CARRIERE J., KAZMAN R.: Interacting with huge hierarchies: Beyond cone trees. *Information Visualization, IEEE Symposium on 0* (01 1995), 74. doi:10.1109/INFVIS.1995.528689. 2

[CP96] COLEMAN M. K., PARKER D. S.: Aesthetics-based graph layout for human consumption. *Software Practice and Experience 26*, 12 (dec 1996), 1415–1438. doi:10.1002/(SICI)1097-024X(199612). 4

[Csi91] CSIKSZENTMIHALYI M.: *Flow: The Psychology of Optimal Experience*. Harper Perennial, New York, NY, 1991. doi:10.1080/00222216.1992.11969876. 10

[CT96] CRUZ I. F., TWAROG J. P.: 3D graph drawing with simulated annealing. In *Graph Drawing* (Berlin, Heidelberg, 1996), Brandenburg F. J., (Ed.), vol. 1027 of *LNCS*, Springer Berlin Heidelberg, p. 162–165. doi:10.1007/BFb0021800. 2

[CTMT23] CASTELEIN W., TIAN Z., MCHEDLIDZE T., TELEA A.: Viewpoint-based quality for analyzing and exploring 3D multidimensional projections. In *VISIGRAPP 2023* (Lisbon, Portugal, 2023), SCITEPRESS - Science and Technology Publications, p. 65–76. doi:10.5220/0011652800003417. 3, 5, 9

[DH89] DAVIDSON R., HAREL D.: Drawing graphs nicely using simulated annealing. *Technical Report CS89-13* (1989). doi:10.1145/234535.234538. 2

[DH11] DAVIS T. A., HU Y.: The university of Florida sparse matrix collection. *ACM Trans. Math. Softw. 38*, 1 (dec 2011). URL: https://sparse.tamu.edu/, doi:10.1145/2049662.2049663. 5

[EHW97] EADES P., HOULE M. E., WEBBER R.: Finding the best viewpoints for three-dimensional graph drawings. In *Graph Drawing* (Berlin, Heidelberg, 1997), vol. 1353 of *LNCS*, Springer Berlin Heidelberg. doi:10.1007/3-540-63938-1_53. 4

[EMK*19] ESPADOTO M., MARTINS R., KERREN A., HIRATA N., TELEA A.: Toward a quantitative survey of dimension reduction techniques. *IEEE TVCG 27*, 3 (2919), 2153–2173. doi:10.1109/TVCG.2019.2944182. 3, 8

[FPK*23] FEYER S. P., PINAUD B., KOBOUROV S., BRICH N., KRONE M., KERREN A., BEHRISCH M., SCHREIBER F., KLEIN K.: 2D, 2.5D, or 3D? An exploratory study on multilayer network visualisations in virtual reality. *IEEE Transactions on Visualization and Computer Graphics* (2023), 1–11. doi:10.1109/TVCG.2023.3327402. 2

[FR91] FRUCHTERMAN T. M. J., REINGOLD E. M.: Graph drawing by force-directed placement. *Software: Practice and Experience 21*, 11 (1991), 1129–1164. doi:10.1002/spe.4380211102. 2

[GDT] GDTOOLKIT: Rome graphs. URL: http://www.graphdrawing.org/data.html. 5

[GGK04] GAJER P., GOODRICH M., KOBOUROV S.: A fast multidimensional algorithm for drawing large graphs. *Computational Geometry 29* (2004), 3–18. 2

[GKN05] GANSNER E. R., KOREN Y., NORTH S.: Graph drawing by stress majorization. In *Graph Drawing* (Berlin, Heidelberg, 2005), vol. 3383 of *LNCS*, Springer Berlin Heidelberg, p. 239–250. doi:10.1007/978-3-540-31843-9_25. 2, 3

[GLA*21] GIOVANNANGELI L., LALANNE F., AUBER D., GIOT R., BOURQUI R.: Deep neural network for drawing networks, (DNN^2). In *Graph Drawing* (Cham, 2021), Purchase H. C., Rutter I., (Eds.), Springer International Publishing, pp. 375–390. doi:10.1007/978-3-030-92931-2_27. 5

[Gon10] GONZÁLEZ : Measurement of areas on a sphere using fibonacci and latitude–longitude lattices. *Mathematical geosciences 42* (01 2010), 49–64. doi:10.1007/s11004-009-9257-x. 3

[GPK11] GREFFARD N., PICAROUGNE F., KUNTZ P.: Visual community detection: An evaluation of 2D, 3D perspective and 3D stereoscopic displays. In *Graph Drawing* (Berlin, Heidelberg, 2011), vol. 7034 of *LNCS*, Springer Berlin Heidelberg, p. 215–225. doi:10.1007/978-3-642-25878-7_21. 2

[HEHL13] HUANG W., EADES P., HONG S.-H., LIN C.-C.: Improving multiple aesthetics produces better graph drawings. *Journal of Visual Languages and Computing 24*, 4 (2013), 262–272. doi:10.1016/j.jvlc.2011.12.002. 3, 4

[HHE08] HUANG W., HONG S.-H., EADES P.: Effects of crossing angles. In *IEEE Pacific Vis* (2008), pp. 41–46. doi:10.1109/PACIFICVIS.2008.4475457. 4

[Hu05] HU Y.: Efficient, high-quality force-directed graph drawing. *Mathematica journal 10*, 1 (2005), 37–71. 2

[Hua07] HUANG W.: Using eye tracking to investigate graph layout effects. In *APVIS 2007, Sydney, Australia, 5-7 February 2007* (2007), Hong S., Ma K., (Eds.), IEEE Computer Society, pp. 97–100. doi:10.1109/APVIS.2007.329282. 3, 4

[HW98] HOULE M. E., WEBBER R.: Approximation algorithms for finding best viewpoints. In *Graph Drawing* (Berlin, Heidelberg, 1998), vol. 1547 of *LNCS*, Springer Berlin Heidelberg, p. 210–223. doi:10.1007/3-540-37623-2_16. 2, 3

[JJHS*22] JOOS L., JAEGER-HONZ S., SCHREIBER F., KEIM D. A., KLEIN K.: Visual comparison of networks in VR. *IEEE Transactions on Visualization and Computer Graphics 28*, 11 (2022), 3651–3661. doi:10.1109/TVCG.2022.3203001. 2

[JVHB14] JACOMY M., VENTURINI T., HEYMANN S., BASTIAN M.: Forceatlas2, a continuous graph layout algorithm for handy network visualization designed for the gephi software. *PLOS ONE 9*, 6 (Jun 2014), e98679. doi:10.1371/journal.pone.0098679. 2, 3

[KK88] KAMADA T., KAWAI S.: A simple method for computing general position in displaying three-dimensional objects. *Computer Vision, Graphics, and Image Processing 41* (1988), 43–56. doi:10.1016/0734-189X(88)90116-8. 2, 3

[KK89] KAMADA T., KAWAI S.: An algorithm for drawing general undirected graphs. *Information Processing Letters 31*, 1 (1989), 7–15. doi:10.1016/0020-0190(89)90102-6. 2, 4

[KMS94] KOSAK C., MARKS J., SHIEBER S.: Automating the layout of network diagrams with specified visual organization. *IEEE Transactions on Systems, Man, and Cybernetics 24*, 3 (1994), 440–454. doi:10.1109/21.278993. 2

[KRM*17] KRUIGER J. F., RAUBER P. E., MARTINS R. M., KERREN A., KOBOUROV S., TELEA A. C.: Graph layouts by t-SNE. *Computer Graphics Forum 36*, 3 (Jun 2017), 283–294. doi:10.1111/cgf.13187. 2, 3, 5, 9

[Kru17] KRUIGER H.: tsNET Python. URL: https://github.com/HanKruiger/tsNET/. 3

[MCH*18] MARRIOTT K., CHEN J., HLAWATSCH M., ITOH T., NACENTA M. A., REINA G., STUERZLINGER W.: *Immersive Analytics: Time to Reconsider the Value of 3D for Information Visualisation*. Springer International Publishing, Cham, 2018, pp. 25–55. doi:10.1007/978-3-030-01388-2_2. 2

[MGM*19] MCGEE F., GHONIEM M., MELANÇON G., OTJACQUES B., PINAUD B.: The state of the art in multilayer network visualization. *Computer Graphics Forum 38*, 6 (2019), 125–149. doi:10.1111/cgf.13610. 2

[MH08] MAATEN L. V. D., HINTON G.: Visualizing data using t-SNE. *Journal of Machine Learning Research 9*, 86 (2008), 2579–2605. 2, 3

[MHK22] MILLER J., HUROYAN V., KOBOUROV S.: Spherical graph drawing by multi-dimensional scaling. In *Graph Drawing* (2022), vol. 13764 of *LNCS*, Springer Berlin Heidelberg. doi:10.1007/978-3-031-22203-0_7. 2

[MRS96] MONIEN B., RAMME F., SALMEN H.: A parallel simulated annealing algorithm for generating 3D layouts of undirected graphs. In *Graph Drawing* (Berlin, Heidelberg, 1996), Brandenburg F. J., (Ed.), vol. 1027 of *LNCS*, Springer Berlin Heidelberg, p. 396–408. doi:10.1007/BFb0021823. 2

[PCJ96] PURCHASE H., COHEN R., JAMES M.: Validating graph drawing aesthetics. In *Graph Drawing* (Germany, 1996), Brandenburg F., (Ed.), vol. 1027 of *LNCS*, Springer Berlin Heidelberg, pp. 435–446. doi:10.1007/bfb0021827. 1, 3, 4, 5

[PMER97] PETER E., MICHAEL E H., RICHARD W.: Finding the best viewpoints for three-dimensional graph drawings. In *Graph Drawing* (Berlin, Heidelberg, 1997), vol. 1353 of *LNCS*, Springer Berlin Heidelberg. doi:10.1007/3-540-63938-1_53. 2, 3, 4

[Pur02] PURCHASE H. C.: Metrics for graph drawing aesthetics. *Journal of Visual Languages and Computing 13*, 5 (2002), 501–516. doi:10.1006/jvlc.2002.0232. 1

[pyq23] PyQtGraph Python. URL: https://www.pyqtgraph.org/. 5

[TZvS*21] TIAN Z., ZHAI X., VAN STEENPAAL G., YU L., DIMARA E., ESPADOTO M., TELEA A.: Quantitative and qualitative comparison of 2D and 3D projection techniques for high-dimensional data. *Information 12*, 239 (2021). doi:10.3390/info12060239. 8

[VCT21] VERNIER E., COMBA J., TELEA A.: Guided stable dynamic projections. *Computer Graphics Forum 40*, 3 (2021), 87–98. doi:https://doi.org/10.1111/cgf.14291. 3

[VGdS*20] VERNIER E., GARCIA R., DA SILVA I., COMBA J., TELEA A.: Quantitative evaluation of time-dependent multidimensional projection techniques. *Computer Graphics Forum 39* (2020), 241–252. doi:10.1111/cgf.13977. 3

[VGO*20] VIRTANEN P., GOMMERS R., OLIPHANT T. E., HABERLAND M., REDDY T., COURNAPEAU D., BUROVSKI E., PETERSON P., WECKESSER W., BRIGHT J., VAN DER WALT S. J., BRETT M., WILSON J., MILLMAN K. J., MAYOROV N., NELSON A. R. J., JONES E., KERN R., LARSON E., CAREY C. J., POLAT İ., FENG Y., MOORE E. W., VANDERPLAS J., LAXALDE D., PERKTOLD J., CIMRMAN R., HENRIKSEN I., QUINTERO E. A., HARRIS C. R., ARCHIBALD A. M., RIBEIRO A. H., PEDREGOSA F., VAN MULBREGT P., SCIPY 1.0 CONTRIBUTORS: SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods 17* (2020), 261–272. doi:10.1038/s41592-019-0686-2. 3

[vWMT23] VAN WAGENINGEN S., MCHEDLIDZE T., TELEA A.: Identifying cluttering edges in near-planar graphs. In *EuroVis 2023 - Short Papers* (2023), Hoellt T., Aigner W., Wang B., (Eds.), The Eurographics Association. doi:10.2312/evs.20231048. 5

[WM08] WARE C., MITCHELL P.: Visualizing graphs in three dimensions. *ACM Trans. Appl. Percept. 5*, 1 (jan 2008). doi:10.1145/1279640.1279642. 2

[WYHS21] WANG X., YEN K., HU Y., SHEN H.: DeepGD: A deep learning framework for graph drawing using GNN. *IEEE Computer Graphics and Applications 41*, 05 (2021), 32–44. doi:10.1109/MCG.2021.3093908. 5, 10