DNC: Dynamic Neighborhood Change Faithfulness Metrics †

Shijun Cai[1], Amyra Meidiana[1] and Seok-Hee Hong[1]

[1]School of Computer Science, University of Sydney, Australia

Abstract

Faithfulness *metrics measure how faithfully a visualization displays the ground truth information of the data. For example,* neighborhood *faithfulness metrics measure how faithfully the* geometric *neighbors of vertices in a graph drawing represent the* ground truth *neighbors of vertices in the graph. This paper presents a new* dynamic neighborhood change *(DNC) faithfulness metric for* dynamic *graphs to measure how proportional the* geometric neighborhood change *in dynamic graph drawings is to the* ground truth neighborhood change *in dynamic graphs. We validate the DNC metrics using deformation experiments, demonstrating that it can accurately measure neighborhood change faithfulness in dynamic graph drawings. We then present extensive comparison experiments to evaluate popular graph drawing algorithms using DNC, to recommend which layout obtains the highest neighborhood change faithfulness on a variety of dynamic graphs.*

1. Introduction

Evaluation has been established as an important research area in graph drawing. Quality metrics, called *aesthetic* criteria, such as edge crossings, area, bends, edge lengths, angular resolution, and crossing angles, have been presented for quantitative evaluation of graph drawings [BETT99]. However, most traditional metrics only measure the *readability* of graph drawings (i.e., how humans understand the graph drawing).

Recently, *faithfulness* metrics have been presented for evaluating large complex graph drawing, to measure how faithfully a drawing D represents the *ground truth* structure of a graph G [EHKN15, MHEK19, MHEK20]. In particular, *neighborhood* faithfulness metrics [NHE17] measure how faithfully the *geometric* neighborhood of vertices in a drawing displays the ground truth neighborhood of vertices of a graph.

Dynamic graphs present significant challenges for visualization and evaluation [BBDW17]. Namely, a dynamic graph drawing should be faithful to the ground truth of the graph at each time slice, but also faithfully displays the *ground truth change* in dynamic graphs. *Change faithfulness* metrics measure how proportional the geometric change in a dynamic graph drawing is to the ground truth change in a dynamic graph, such as cluster and distance change faithfulness metrics [MHE20]. However, *neighborhood change* faithfulness metric has not been presented.

To fill in this gap, we present new *Dynamic Neighborhood Change* (*DNC*) faithfulness metrics for *dynamic* graphs to measure how proportional the change in the geometric neighborhood of vertices in a dynamic graph drawing is to the ground truth neighborhood changes of vertices in a dynamic graph. We validate the *DNC* metrics using deformation experiments, demonstrating that it can effectively measure the neighborhood change faithfulness of dynamic graph drawings.

We then evaluate popular graph drawing algorithms using *DNC*, to recommend which layouts can produce better neighborhood change faithful drawings, using a variety of real-world and synthetic dynamic graphs. Overall, tsNET [KRM*17] performs the best, followed by multi-level layouts (FM^3 [HJ05] and sfdp [Hu05]), the Backbone layout [NOB14] and the Stress Majorization layout [GKN05].

2. Related Work

2.1. Faithfulness Metrics for Static Graph Drawings

Faithfulness metrics measure how faithfully graph drawings display the ground truth information of graphs. For example, *stress* [BETT99] is a *distance* faithfulness metric, measuring how proportional the Euclidean distance of vertices in a drawing D is to the graph-theoretic distance of vertices in a graph G. *Shape-based* metrics [EHKN15] measure how faithfully the "shape" (i.e., *proximity graph*) of D represents the ground truth structure of G. Similarly, the *cluster* faithfulness metrics [MHEK19] measure the similarity between the geometric clustering of a drawing D and the ground truth clustering of a graph G. The *symmetry* faithfulness metrics [MHEK20] measure the similarity between symmetries in D and the ground truth *automorphisms* of a graph G.

† This research was supported by an ARC grant (DP190103301).

DOI: 10.2312/evs.20221092

In particular, the *neighborhood* faithfulness metrics [NHE17] measure how faithfully the *geometric* neighborhood of vertices in a drawing displays the *graph-theoretic* neighborhood of vertices in a graph. Namely, the metric is defined using *dNNG* (degree-sensitive nearest neighborhood graph), a variation of the *kNNG* (*k*-nearest neighborhood graph), based on the degree of a vertex (i.e., $k = d(v)$, where $d(v)$ is the degree of a vertex v). More specifically, the neighborhood faithfulness metrics are defined using the similarity between the neighbors of vertices in a graph G and the *dNNG* of vertices in a drawing D of G.

2.2. Faithfulness Metrics for Dynamic Graph Drawings

A *dynamic* graph is defined by a sequence of static graphs $G_1, G_2, \ldots, G_l$ with l time steps, where G_i is a *time slice* of the graph at time step i [BBDW17]. The most well-known criteria for dynamic graph drawings focus on preserve the *mental map* [MELS95], which can be modeled using orthogonal ordering, clustering, topology and distances [Bra01, DG02]. Relatedly, *dynamic stability* is defined as the minimization of geometric distance between successive drawings [TBB88, BBDW17].

Change faithfulness metrics measure how faithfully the ground truth change in dynamic graphs is displayed as the geometric change in the dynamic graph drawings. For example, the *cluster change* faithfulness metrics [MHE20] measure how proportional the ground truth change in the clustering of a dynamic graph is displayed as changes in geometric clustering of the dynamic graph drawing. Similarly, the *distance change* faithfulness metrics [MHE20] measure how proportional the ground truth change in graph-theoretic distance in a dynamic graph is displayed as changes in the geometric distance in the dynamic graph drawing.

3. Dynamic Neighborhood Change Faithfulness Metrics

This section presents a new change faithfulness metric *DNC* to measure how proportionally the ground truth change in the neighborhood of vertices in dynamic graphs is represented as a change in the geometric neighborhood of vertices in the drawing.

Figure 1 shows the *DNC* metric framework, which can be computed by the following steps:

1. Compute drawings D_1 and D_2 of two time slices of a dynamic graph $G_1 = (V_1, E_1), G_2 = (V_2, E_2)$ using a graph layout.
2. Compute the degree-sensitive nearest neighborhood graphs $dNNG_1 = (V_1', E_1')$ (resp., $dNNG_2 = (V_2', E_2')$) of the point set P_1 of D_1 (resp., P_2 of D_2), corresponding to V_1 (resp., V_2).
3. Compute $\Delta(dNNG_1, dNNG_2)$, the change between $dNNG_1$ and $dNNG_2$.
4. Compare how proportional $\Delta(dNNG_1, dNNG_2)$ is to the ground truth change in dynamic graphs, $\Delta(G_1, G_2)$.

We use the Jaccard Similarity index [Jac12] to compute $\Delta(dNNG_1, dNNG_2)$ and $\Delta(G_1, G_2)$. Specifically, the normalized Jaccard similarity index $JS(G, G')$ computes the similarity between two graphs G and G' as follows:

$$JS(G,G') = \frac{1}{|V|} \sum_{v \in V} \frac{|N(v) \cap N'(v)|}{|N(v) \cup N'(v)|}$$

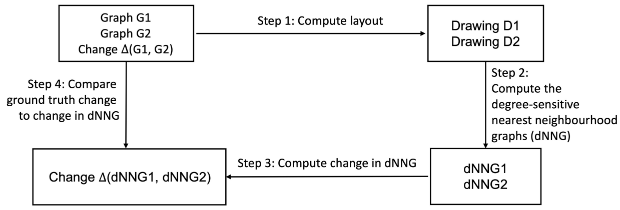

Figure 1: *The neighborhood change faithfulness metric framework.*

where $N(v)$ (resp., $N'(v)$) is the set of neighbors of a vertex v in G (resp., G'). A higher $JS(G, G') \in [0, 1]$ means higher similarity.

For Step 4, we define the *ratio of difference* (*rd*):

$$rd = \frac{|JS(G_1, G_2) - JS(dNNG_1, dNNG_2)|}{max(JS(G_1, G_2), JS(dNNG_1, dNNG_2))}$$

where $rd \in [0, 1]$ is ensured by dividing by the maximum value of $JS(G_1, G_2)$ and $JS(dNNG_1, dNNG_2)$, and a lower value is better.

Note that *rd* on its own does not consider the static neighborhood faithfulness at each time slice. However, dynamic graph drawings should not only display a change proportional to the ground truth change between the two time slices, but also the drawing at each time slice should faithfully display the structure of the time slice. Therefore, we define two variations of *DNC* with different ways to incorporate static neighborhood faithfulness.

The first metric averages the *rd* with the neighborhood faithfulness of each time slice, to give both equal weight:

$$DNC_1 = \frac{1}{2}((1 - rd) + avg(JS(G_1, dNNG_1), JS(G_2, dNNG_2)))$$

The second metric uses a multiplier instead, to be more sensitive in measuring differences in neighborhood change faithfulness:

$$DNC_2 = (1 - rd) \cdot avg(JS(G_1, dNNG_1), JS(G_2, dNNG_2))$$

DNC values range between 0 and 1, where a higher value means better neighborhood change faithfulness.

4. DNC Validation Experiment

We validate the effectiveness of *DNC* metrics, using deformation experiments on graphs with synthetic dynamics to control the extent of the ground truth change in dynamic graphs. We use different types of graphs: *mesh* graphs [DH11]; *black-hole graphs* with globally mesh-like structures and locally dense clusters [ENH17]; and real-world benchmark graphs [KAB*19, New01]. The graphs have number of vertices $|V| \in [135, 2851]$ and $density \in [2.33, 20.28]$.

We start with good neighborhood faithful drawings D_1 and D_2 of two time slices of a dynamic graph G_1 and G_2 (i.e., number of vertices and/or edges may change between time slices), such that $dNNG_1, dNNG_2$ are very similar to G_1 and G_2, and as such the change between $dNNG_1$ and $dNNG_2$ is similar to the ground truth change between G_1 and G_2.

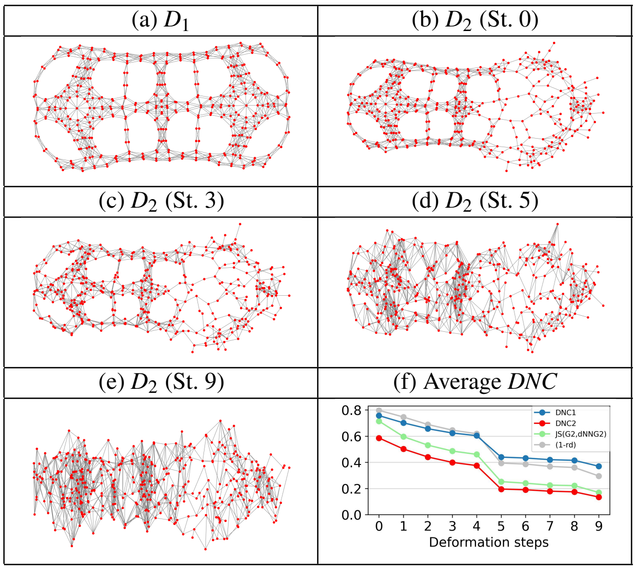

Table 1: *(a)-(e) Example of the deformation steps; (f) Average DNC over all data sets. Both metrics decrease with deformation steps, confirming H1, and DNC_2 performs better than DNC_1.*

We then gradually perturb the position of vertices in D_2 to make the change in the neighborhood graphs gradually become more disproportionate to the ground truth change in dynamic graphs. Specifically, we perform nine steps of deformation on D_2, where in each step, the coordinate of each vertex from the previous step is randomly perturbed by a value in the range $[0, \delta]$, where δ is the size of the drawing area multiplied by a value in the range $[0.02, 0.09]$. We hypothesize: **H1**: *DNC decreases as D_2 is deformed.*

Table 1(a)-(e) shows an example of the deformation steps, and Table 1(f) shows the *DNC* metrics averaged over all validation data sets. Clearly, both DNC_1 and DNC_2 decrease with the deformation steps, and the decreasing pattern is consistent throughout all validation data sets, demonstrating that they can effectively capture the neighborhood change faithfulness, supporting H1.

Furthermore, DNC_2 is more sensitive than DNC_1, as seen in Table 1(f). In addition, the drawing at step 9 is very far from the original neighborhood faithful drawing, and therefore *DNC* value should be very small. However, DNC_1 is still relatively high at step 9, while DNC_2 is much lower. Therefore, DNC_2 more effectively measures the neighborhood change faithfulness, and we use DNC_2 for layout comparison experiments.

5. Layout Comparison Experiment

Experiment Design. To evaluate the performance of various graph layouts on neighborhood change faithfulness, we conduct layout comparison experiments using DNC_2. We select nine popular graph layouts with various optimization criteria:

- Force-directed layouts: Fruchterman-Reingold (FR) [FR91] and Organic (OR) layout [WEK01].
- Multi-level graph layouts: FM^3 [HJ05] and sfdp [Hu05].

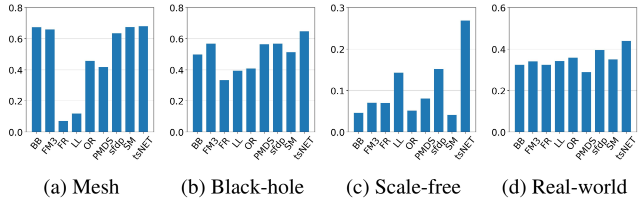

Figure 2: *Average DNC_2 for (a) mesh graphs; (b) black-hole graphs; (c) scale-free graphs; and (d) real-world dynamic graphs.*

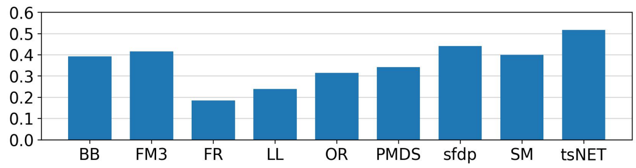

Figure 3: *Average DNC_2 across all data sets. tsNET performs the best, followed by sfdp, FM^3, SM and BB.*

- Backbone (BB) [NOB14], to untangle hairballs in a drawing.
- LinLog (LL) [Noa04], a force-directed layout showing clusters.
- Pivot MDS (PMDS) [BP07], a fast multi-dimensional scaling-based layout.
- Stress Majorization (SM) [GKN05] to minimize the *stress*.
- tsNET [KRM*17], based on t-SNE [vdMH08].

We expect neighborhood faithful layouts for static graphs would be neighborhood change faithful for dynamic graphs based on the results for previous change faithfulness metrics [MHE20]. Since tsNET is designed for neighborhood preservation, and FM^3 shows good performance on *dNNG* on static graphs, we hypothesize: **H2**: *tsNET obtains the highest DNC_2, and multi-level layouts (FM^3 and sfdp) perform well on DNC_2.*

We use graphs with simulated dynamics (i.e., add/delete vertices/edges) based on different types of static graphs: *mesh* graphs [DH11]; *black-hole graphs* [ENH17]; and real-world *scale-free* graphs [LK14]. We also use *real-world dynamic* graphs with naturally occurring dynamics [GVF*15, FB14, GBC14, SVB*11]. The graphs have $|V| \in [59, 6367]$ and $density \in [1.5, 23.28]$.

Mesh graphs. Figure 2(a) shows the average DNC_2 on mesh graphs. tsNET, BB, FM^3, SM and sfdp performs the best, supporting H2. Table 2(a) shows a layout comparison of a mesh graph *3elt*. Most layouts, except FR and LL, perform well by not only untangling the mesh to ensure neighbors are close together and non-neighbors (e.g., vertices separated by the "holes") are far apart, but also showing the change in structure faithfully. The exceptions are FR, which fails to untangle the mesh, and LL, which unnecessarily clusters vertices despite the mesh not having locally dense clusters; this is reflected by much lower DNC_2 than other layouts.

Black-hole graphs. Figure 2(b) shows the average DNC_2 on black-hole graphs. Overall, tsNET performs the best, followed by FM^3, sfdp and PMDS, supporting H2. Moreover, BB and SM also perform well. Table 2(b) shows a layout comparison of a black-hole

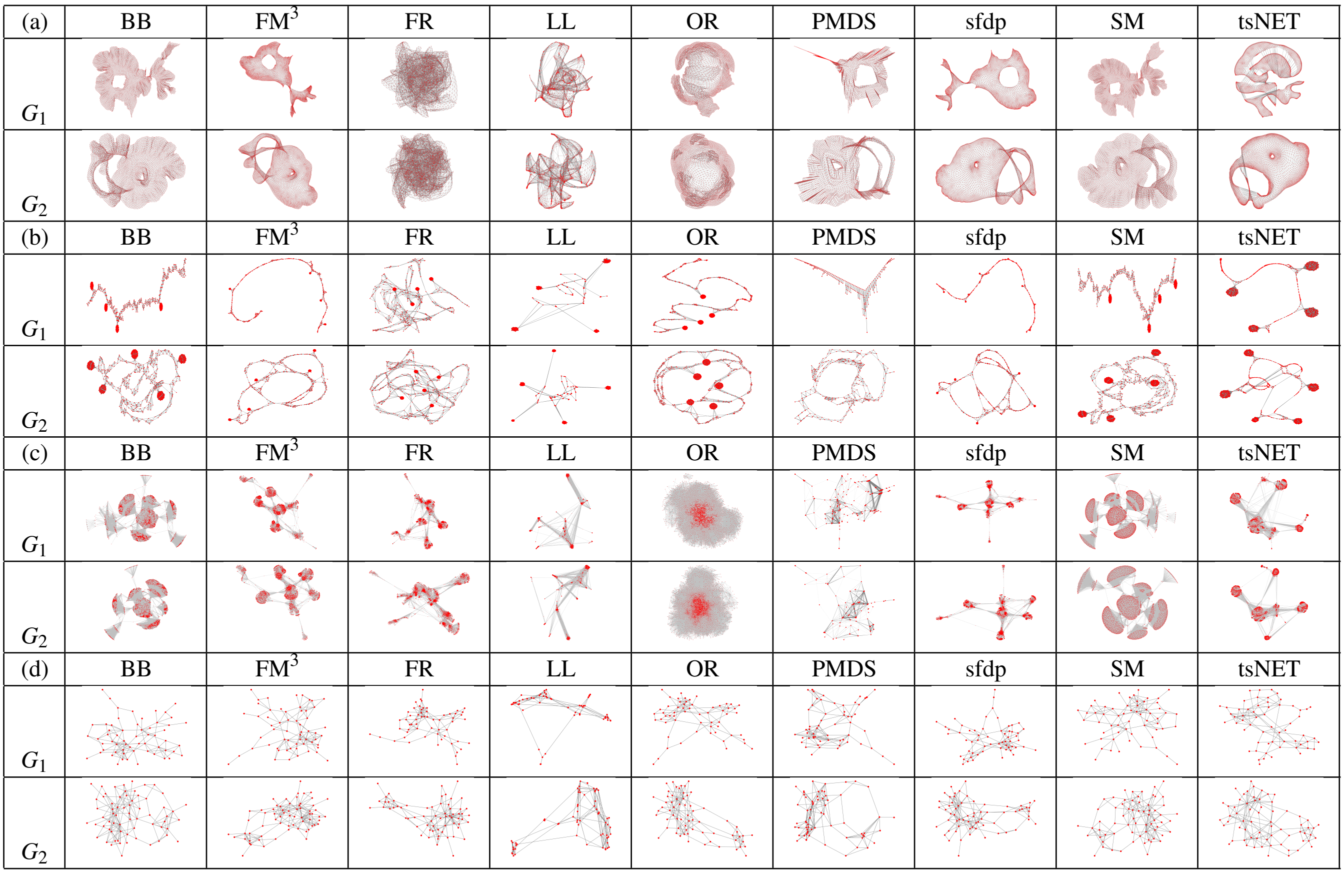

Table 2: *Layout comparison on DNC_2. (a)* 3elt *(mesh), (b)* Cycle896 *(black-hole), (c)* facebook *(scale-free) and (d)* InVS *(real-world dynamic).*

graph *Cycle896*. tsNET shows the global change in structure faithfully without excessive overlap inside the dense blobs (i.e., neighbors are close together, but non-neighbors are further apart). On the other hand, FR performs the worst, failing to untangle the global cycle-like structure.

Scale-free graphs. Figure 2(c) shows the average DNC_2 on scale-free graphs. Clearly, tsNET outperforms other layouts, and sfdp and LL also perform well, mostly supporting H2. Table 2(c) shows a layout comparison of a scale-free graph *facebook*. tsNET, sfdp, and LL show the clusters most distinctly, resulting close neighbors drawn close together, and non-neighbors drawn far apart.

Real-world dynamic graphs. Figure 2(d) shows the average DNC_2 on real-world dynamic graphs. Clearly, tsNET performs the best, followed by sfdp, partially supporting H2. The real-word dynamic graphs are relatively small, and most layouts produce drawings with similar quality, resulting in similar neighborhood change faithfulness. Table 2(d) shows a visual comparison of a real-world dynamic graph *InVS*.

Discussion and Summary. Figure 3 shows the average DNC_2 over all data sets, where tsNET performs the best, followed by FM^3 and sfdp, confirming H2. This result is consistent with our expectation, since tsNET is specifically designed to maximize neighborhood preservation, and therefore achieve high neighborhood change faithfulness. Meanwhile, multi-level graph layouts (i.e., FM^3 and sfdp) recursively use clustering to reduce the size of graphs, leading to neighboring vertices, which are most likely in the same clusters, drawn close together.

Moreover, BB and SM also perform well on average. Since BB aims to untangle hairballs, it can reduce non-neighbors drawn too close together. Meanwhile, SM aims to minimize *stress*, i.e., faithfully represents the graph-theoretic distances in the drawing, which also helps perform well on neighborhood change faithfulness.

Note that while LL obtains relatively high DNC_2 for scale-free graphs, it obtains the lowest DNC_2 on mesh graphs. Since LL focuses on showing clusters, it can perform well on scale-free graphs containing locally dense clusters. However, for mesh graphs without dense clusters, it may introduce unnecessary clusters, resulting in non-neighboring vertices drawn too close together. Furthermore, FR performs the worst on average, consistent with all data types.

In summary, DNC layout comparison experiments show that on average tsNET performs the best, followed by FM^3, sfdp, BB and SM. Specifically, we recommend tsNET, BB, FM^3, sfdp, SM for mesh graphs, and tsNET, LL, sfdp for scale-free graphs.

References

[BBDW17] Beck F., Burch M., Diehl S., Weiskopf D.: A taxonomy and survey of dynamic graph visualization. *CGF 36*, 1 (2017), 133–159. doi:10.1111/cgf.12791. 1, 2

[BETT99] Battista G. D., Eades P., Tamassia R., Tollis I. G.: *Graph Drawing: Algorithms for the Visualization of Graphs*. Prentice-Hall, 1999. 1

[BP07] Brandes U., Pich C.: Eigensolver methods for progressive multidimensional scaling of large data. In *GD* (2007), pp. 42–53. doi:10.1007/978-3-540-70904-6_6. 3

[Bra01] Branke J.: Dynamic graph drawing. In *Drawing Graphs, Methods and Models*. 2001, pp. 228–246. doi:10.1007/3-540-44969-8_9. 2

[DG02] Diehl S., Görg C.: Graphs, they are changing. In *GD* (2002), pp. 23–31. doi:10.1007/3-540-36151-0_3. 2

[DH11] Davis T. A., Hu Y.: The university of florida sparse matrix collection. *ACM TOMS 38*, 1 (2011), 1. doi:10.1145/2049662.2049663. 2, 3

[EHKN15] Eades P., Hong S., Klein K., Nguyen A.: Shape-based quality metrics for large graph visualization. In *GD* (2015), pp. 502–514. doi:10.1007/978-3-319-27261-0_41. 1

[ENH17] Eades P., Nguyen Q. H., Hong S.-H.: Drawing big graphs using spectral sparsification. In *Graph Drawing* (2017), pp. 272–286. doi:10.1007/978-3-319-73915-1_22. 2, 3

[FB14] Fournet J., Barrat A.: Contact patterns among high school students. *PLOS ONE 9*, 9 (2014), 1–17. doi:10.1371/journal.pone.0107878. 3

[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. 3

[GBC14] Gemmetto V., Barrat A., Cattuto C.: Mitigation of infectious disease at school: targeted class closure vs school closure. *BMC infectious diseases 14*, 1 (Dec. 2014), 695. doi:10.1186/PREACCEPT-6851518521414365. 3

[GKN05] Gansner E. R., Koren Y., North S.: Graph drawing by stress majorization. In *GD* (2005), pp. 239–250. doi:10.1007/978-3-540-31843-9_25. 1, 3

[GVF*15] GÉNOIS M., VESTERGAARD C. L., FOURNET J., PANISSON A., BONMARIN I., BARRAT A.: Data on face-to-face contacts in an office building suggest a low-cost vaccination strategy based on community linkers. *Network Science 3* (9 2015), 326–347. doi:10.1017/nws.2015.10. 3

[HJ05] Hachul S., Jünger M.: Drawing large graphs with a potential-field-based multilevel algorithm. In *GD* (2005), pp. 285–295. doi:10.1007/978-3-540-31843-9_29. 1, 3

[Hu05] Hu Y.: Efficient and high quality force-directed graph drawing. *Mathematica 10* (01 2005), 37–71. 1, 3

[Jac12] Jaccard P.: The distribution of the flora in the alpine zone. 1. *New Phytologist 11*, 2 (1912), 37–50. 2

[KAB*19] Kolodziej S. P., Aznaveh M., Bullock M., David J., Davis T. A., Henderson M., Hu Y., Sandström R.: The suitesparse matrix collection website interface. *J. Open Source Softw. 4*, 35 (2019), 1244. doi:10.21105/joss.01244. 2

[KRM*17] Kruiger J. F., Rauber P. E., Martins R. M., Kerren A., Kobourov S., Telea A. C.: Graph layouts by t-sne. *CGF 36*, 3 (2017), 283–294. doi:10.1111/cgf.13187. 1, 3

[LK14] Leskovec J., Krevl A.: SNAP Datasets: Stanford large network dataset collection. http://snap.stanford.edu/data, June 2014. 3

[MELS95] Misue K., Eades P., Lai W., Sugiyama K.: Layout adjustment and the mental map. *JVLC 6*, 2 (1995), 183–210. doi:10.1006/jvlc.1995.1010. 2

[MHE20] Meidiana A., Hong S., Eades P.: New quality metrics for dynamic graph drawing. In *GD* (2020), pp. 450–465. doi:10.1007/978-3-030-68766-3_35. 1, 2, 3

[MHEK19] Meidiana A., Hong S., Eades P., Keim D.: A quality metric for visualization of clusters in graphs. In *GD* (2019), pp. 125–138. doi:10.1007/978-3-030-35802-0_10. 1

[MHEK20] Meidiana A., Hong S., Eades P., Keim D.: Quality metrics for symmetric graph drawings. In *PacificVis* (2020), pp. 11–15. doi:10.1109/PacificVis48177.2020.1022. 1

[New01] Newman M. E.: The structure of scientific collaboration networks. *Proceedings of the national academy of sciences 98*, 2 (2001), 404–409. 2

[NHE17] Nguyen Q. H., Hong S., Eades P.: dNNG: Quality metrics and layout for neighbourhood faithfulness. In *PacificVis* (2017), pp. 290–294. doi:10.1109/PACIFICVIS.2017.8031607. 1, 2

[Noa04] Noack A.: An energy model for visual graph clustering. In *GD* (2004), pp. 425–436. doi:10.1007/978-3-540-24595-7_40. 3

[NOB14] Nocaj A., Ortmann M., Brandes U.: Untangling hairballs: From 3 to 14 degrees of separation. In *GD* (2014), pp. 101–112. doi:10.1007/978-3-662-45803-7_9. 1, 3

[SVB*11] Stehlé J., Voirin N., Barrat A., Cattuto C., Isella L., Pinton J., Quaggiotto M., Van den Broeck W., Régis C., Lina B., Vanhems P.: High-resolution measurements of face-to-face contact patterns in a primary school. *PLOS ONE 6*, 8 (08 2011), e23176. doi:10.1371/journal.pone.0023176. 3

[TBB88] Tamassia R., Battista G. D., Batini C.: Automatic graph drawing and readability of diagrams. *IEEE Trans. Syst. Man Cybern. 18*, 1 (1988), 61–79. doi:10.1109/21.87055. 2

[vdMH08] van der Maaten L., Hinton G.: Visualizing data using t-sne. *JMLR 9*, 86 (2008), 2579–2605. 3

[WEK01] Wiese R., Eiglsperger M., Kaufmann M.: *yFiles: Visualization and Automatic Layout of Graphs*, vol. 2265 of *Lecture Notes in Computer Science*. Springer, 2001, pp. 453–454. URL: https://doi.org/10.1007/3-540-45848-4_42, doi:10.1007/3-540-45848-4_42. 3