Comparison of Symmetry Measures and Symmetry Detection Techniques for Network Visualisations

Sebastian Röder
TU Wien

***Abstract*—Symmetry is an important quality in drawings of graphs, which motivates the need to establish objective metrics for measuring the amount of symmetry present in a a drawing. In this paper, an overview of the state of research on symmetry metrics for graph drawings is given. Two of the approaches, namely the node-based approach by Purchase, 2002, and the edge-based approach by Klapaukh et al., 2018, are discussed in more detail. The two approaches are compared both in regard to runtime and their accuracy in detecting symmetry for certain instances. Advantages and disadvantages of the metrics are presented.**

***Index Terms*—Graph Drawing, Symmetry, Aesthetics, Quality Metrics**

I. Introduction

A substantial body of work exists on aesthetic quality criteria to enhance the readability of a graph drawing and network visualisation [1]–[4]. Among the proposed criteria, the number of edge crossings has been identified as the most important criterion for evaluating the quality of a drawing [4]. However, symmetry is also a significant factor in regard to readability [4]. Symmetric drawings of graphs can improve the performance on readability tasks as shown by Purchase et al. [5]. Kieffer et al. showed that reflectively symmetric graphs are preferred by users [6]. In a user study by Marriott et al., symmetry – among other qualities – is identified to improve the memorability of graphs [7]. In a large-scale study by Ball and Geyer-Schulz, over 70 percent of analyzed graphs contain structural symmetries [8]. Efficient symmetry metrics can not only evaluate finished drawings but also guide automatic graph embedding algorithms to increase the symmetry of the output [9]. These observations motivate the need to reliably measure symmetry in graphs.

A. Previous work

Structural symmetries in graphs are also known as graph automorphisms [10]. The study of graph automorphisms goes back to the 1950s [11]. The first graph embedding algorithm using detected automorphisms was introduced by Lipton et al. in 1985 [12].

It is important to note the difference between the topological setting of graph automorphisms and the geometric setting, where symmetry is measured in the context of the positions of nodes on the plane. The latter setting is the focus of this paper. To the author's knowledge, six symmetry metrics for the geometric setting have been proposed so far; an overview is given in TABLE I. All metrics mentioned are targeted to straight-line drawings of graphs, meaning that edges are assumed to be drawn as line segments.

TABLE I
OVERVIEW OF THE IDENTIFIED SYMMETRY METRICS

Metric	**Type**	**Runtime**	**Source**
Purchase metric	Reflective	$O(n^7)$	[3]
KMP metric	Reflective, rotational, translational	$O(m^2 \cdot N)$	[13]
Stress-based symmetry	n.a.	$O(n \log n)$	[15]
Force-based symmetry	n.a.	$O(n \log n)$	[9]
Machine learning	Reflective	$O(n + m)$	[16]
Automorphism detection	Reflective, translational	$O(n \log n)$	[10]

The first symmetry metric, proposed by Purchase in 2002 [3], measures the degree of reflective symmetry. A symmetry axis is generated for each pair of nodes. Then, for each such symmetry axis α, the maximally-sized subgraph that is reflectively symmetric along α is calculated. The final metric is obtained by determining the size of the convex hull of all such maximally-sized subgraphs.

Given the asymptotic worst-case runtime of $O(n^7)$ for the Purchase metric and the fact that it mostly ignores edges, Klapaukh, Marshall and Pearce propose a different metric in an attempt to improve upon the previous approach [13]. The metric only considers axes along edges instead of each node pair and extends a method for symmetry detection of image features by Loy and Eklundh [14]. It will be referred to as the KMP metric for the remainder of this paper.

Given that stress-based graph embedding algorithms tend to yield symmetric drawings, Welch and Kobourov propose a metric based on stress [15]. The stress between a drawing of a graph $G = (V, E)$ is defined by

$$\sum_{u,v \in V} (||u - v|| - d(u, v))^2, \tag{1}$$

where $||u - v||$ refers to the Euclidean distance and $d(u, v)$ to the graph distance between u and v. The authors also show that the stress-based metric cannot outperform the previous two metrics.

Intended to be used to guide force-based graph embedding algorithms, Xu, Yang and Gou propose a local symmetry metric σ_v for each node v [9]. Let C_v be the smallest enclosing circle containing all neighbouring nodes of v. The symmetry metric σ_v is defined as the Euclidean distance between the centre of C_v and the position of v scaled by the radius of C_v. The symmetry value for the whole drawing is defined as the average over all nodes, i.e., $\frac{1}{n}\sum_{v \in V} \sigma_V$.

De Luca et al. propose a machine-learning approach to detect rotational, translational and reflective symmetry [16]. Even though the authors claim to achieve high accuracy in detecting symmetry, their approach only yields a binary value, whereas all other identified approaches can give a qualitative score between 0 and 1.

Meidiana et al. propose a symmetry metric for the degree to which a drawing represents the underlying automorphism of the graph [10], i.e., how well the drawing represents the underlying graph structure.

In this paper, the approaches proposed by Purchase as well as by Klapaukh et al. are discussed. Both algorithms return a value in the range of $[0, 1]$. In the following sections, these two metrics are presented in detail. Finally, the two metrics are compared and discussed.

B. Definitions

Let $G = (V, E)$ denote a graph consisting of a set of nodes V and a set of edges E. Let n denote the number of nodes and m the number of edges in G. A *drawing* of G maps all nodes $v \in V$ to position in the plane and all edges $e = (u, v) \in E$ to line segments starting at the position of u and ending at v.

Three types of symmetry are distinguished in literature [5]. If a pattern exhibits *reflective symmetry*, it is mirrored along an axis. If a pattern exhibits *rotational symmetry*, if it remains unchanged after rotating it around a central point for a given angle $\alpha < 360°$. A pattern exhibits *translational symmetry* if it remains unchanged after applying a shift transformation.

II. Purchase Metric

This metric is only intended for determining reflective symmetry. However, it is feasible to extend the approach to different kinds of symmetries. Here, only the original metric for reflective symmetry is considered, as it is well-defined and extending the approach necessitates additional considerations.

An overview of the approach is depicted in Fig. 1. As a pre-processing step, the approach elevates all crossings to, i.e., crossings are converted to nodes (see Fig. 1b). For each pair of nodes in this planarised graph, an axis α is generated along the perpendicular bisector as depicted in Fig. 1c.

For each axis α an induced subgraph G_α, consisting of all edges mirrored along the edges, is generated (see Fig. 1d). For an edge to be considered the mirror image of another edge, the endpoints of the edges have to be mirrored within a certain distance. If the subgraph exceeds a certain number of edges, the axis is considered further, otherwise it is ignored (see Fig. 1e). For each considered axis α, an axis-specific value s_α is calculated, which indicates the overlap in node types, i.e., original nodes versus crossings that have been elevated to nodes.

To obtain s_α, a value $s_\alpha^{e_1,e_2}$ for each mirrored pair of edge in G_α is defined as

$$s_\alpha^{e_1,e_2} = \begin{cases} f^2 & \text{if both pairs of endpoints are of different types} \\ f & \text{if one pair consists of different types} \\ 1 & \text{otherwise,} \end{cases} \tag{2}$$

where f is a fraction given as input to define the degree to which crossings should be considered different to normal nodes.

The value of s_α is the average over all edge-specific values, i.e. $\sum s_\alpha^{e_1,e_2}/|E|$.

The final metric is defined as

$$\frac{\sum_\alpha s_\alpha \cdot \text{area}(G_\alpha)}{\max(\text{area}(G), \sum_\alpha \text{area}(G_\alpha))}, \tag{3}$$

where area() refers to the area of the convex hull of a graph.

The metric can be adapted using the following parameters:

- *Threshold*: This specifies the minimum number of nodes in the symmetric subgraph G_α to consider α further.
- *Tolerance*: The maximum distance between two node positions to still be considered equal. This is used to determine whether two edges are mirror images of each other.
- *Fraction*: The fraction f used to calculate the axis-specific value s_α.

As the metric considers multiple symmetry axes, it can measure both local and global symmetries.

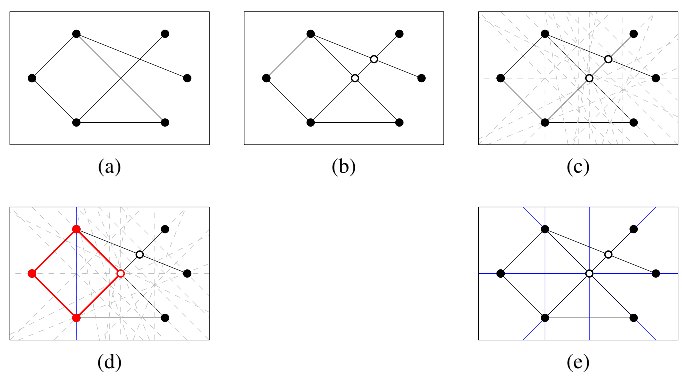


Fig. 1. Step-by-step overview of obtaining symmetry axes for the Purchase approach.

III. KMP Metric

The Purchase metric is limited to reflective symmetry. Further, it is computationally expensive – as an axis is generated for each pair of nodes – and primarily focuses on the position of nodes. Given these restrictions, Klapaugh, Marshall and Pearce propose a new symmetry metric [13]. This metric is based on an approach to detect symmetries in images from feature points proposed by Loy and Eklundh [14]. The approach can detect rotational, translational and reflective symmetry, however, it only considers points. Klapaugh, Marshall and Pearce extend the approach to be able to incorporate line segments as well. The metric obtains a separate value for each type of symmetry, with no approach to obtaining an overall metric incorporating all symmetry types.

For each pair of edges, an axis maximising the chosen symmetry type between the two edges is added. For estimating reflective symmetry, the KMP metric adds an additional symmetry axis for the perpendicular bisector of the two endpoints of each edge, as well as an axis parallel to the edge. For rotational symmetry, an axis is added at the midpoint of each edge.

For each generated axis, a score s_α between 0 and 1 is calculated. The score of an axis indicates how similar the two edges are in regard to their length and orientation. The score of an axis generated from a single edge is always 1.

Axes are then quantised, axes with equal quantised position and rotation are grouped and their individual scores are summed up.

Instead of considering all axes, as for the Purchase metric, here, only the top N axes are chosen based on the score s_α. The final metric consists of the percentage of edges that voted for the top N axes:

$$\frac{\sum_\alpha \text{number of edges that voted for } \alpha}{N \cdot \text{number of lines}} \tag{4}$$

A trade-off between global and local symmetry can be made by appropriately choosing a value for the parameter N. With lower values of N, only the stronger global symmetry axes are considered; when considering more axes, smaller symmetries are evaluated as well.

IV. Comparison

To compare the performance of the metrics, both have been implemented in Python. The implementation is published as part of the gdMetriX package on the Python package index[1]. For obtaining the presented results, version 0.0.1 was used. The implementation for the KMP metric strictly follows the Java implementation by the original authors[2]. In this section, both the runtime as well as the agreement of both approaches are compared.

A. Runtime comparison

The runtime was compared using 720 randomized Erdös-Rényi graphs with a node size in the range $[0, 80]$, and edge densities of $10\%, 20\%, 30\%, \ldots, 80\%, 90\%$. The embedding was generated by assigning random uniformly distributed node positions within $[0, 1]^2$. Both algorithms were executed on an Intel® Core™ i5-8250U with 1.6 GHz. A cutoff time of 12 seconds was chosen. The parameters of both algorithms are left to the default values of the Python package. This includes the number of axes N selected by the KMP approach, which is set to $N = 1$.

The results of the experiment can be seen in Fig. 2. The KMP metric performs better than the Purchase metric independently of the density of the instances. The Purchase metric exceeds the time limit of 12 seconds even for small instances consisting of less than 20 nodes. This is also the case for sparse instances.

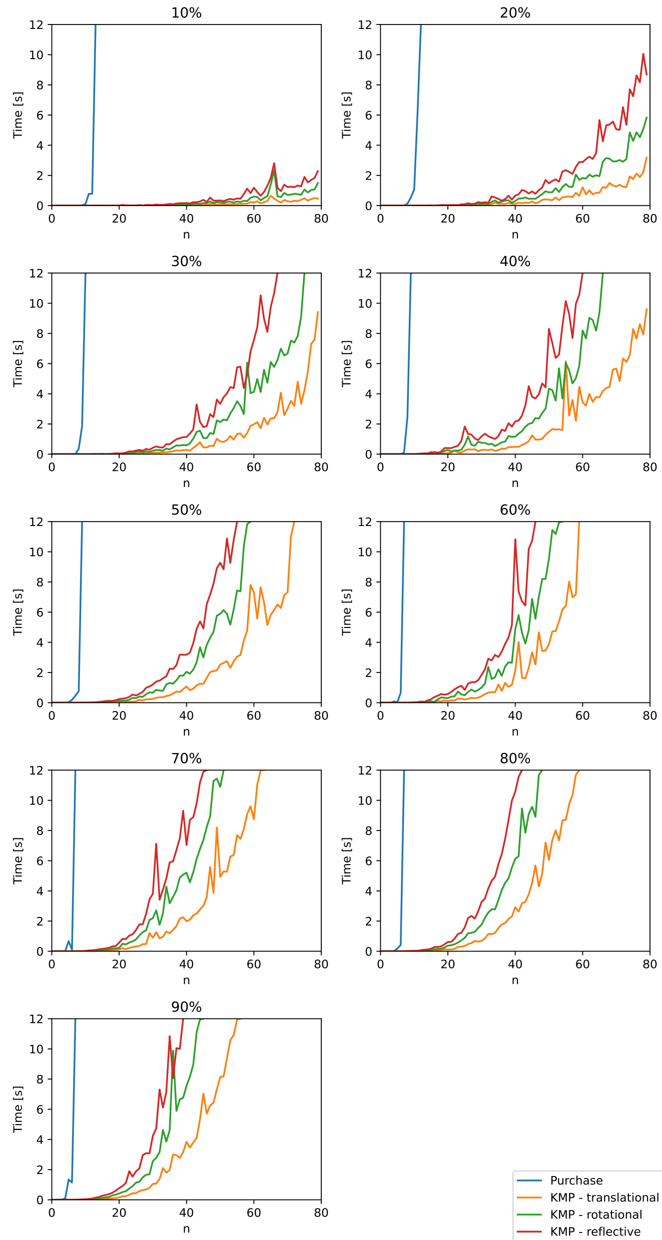


Fig. 2. A comparison in runtime between the metrics.

However, the approach by Purchase – which is primarily concerned about nodes only – still slows down with an increasing number of edges. The time limit is exceeded at $n = 16$ for a density of 10% and at $n = 8$ for a density of 90%. This aligns with the fact that the asymptotic runtime of $O(n^7)$ is only a worst-case asymptotic runtime – the best-case runtime is stated as $O(n^5)$ by Purchase [3] – and the number of considered axes depends on the number of edges in conjunction with the threshold parameter.

As expected, the KMP metric slows down with an increasing

[1] See https://pypi.org/project/gdMetriX/, the documentation can be found at https://livus.github.io/gdMetriX/

[2] See https://github.com/klapaukh/GraphAnalyser.

number of edges as well. The difference between the three types of symmetries only consists of a constant factor.

B. Agreement

No publicly available annotated dataset of graph drawings with user-evaluated symmetry values exists, which could be used to evaluate the quality and precision of the investigated metrics. To investigate the agreement on symmetry between the two metrics instead, a set of Erdös-Rényi graphs was generated. The graphs were embedded using the Spring embedder provided by the Python package networkX version 3.3[3]. The dataset contains 255 graphs with the number of nodes per graph uniformly distributed in the range of $[5, 11]$. The edge density is uniformly distributed in the range of $[0, 1]$. The values for the Purchase metric are compared to the KMP metric for reflective symmetry only, as this is the only type of symmetry supported by both metrics.

The results for both metrics vary significantly depending on the chosen parameters. Unfortunately, the authors do not give a guideline for choosing reasonable values. It is left to the user to investigate suitable values for a given use case. For the results presented in this section, a tolerance of 0.15, a threshold of 10, and a fraction of 0.5 was used for the Purchase metric. For the KMP metric, the default values specified in the original Java implementation were used.

The two metrics express a Pearson's correlation coefficient of 0.07. A scatter plot of the two metrics can be seen in Fig. 3. While the Purchase metric outputs a symmetry estimate in the range of 0.6 to 0.85 for most instances, the KMP symmetry varies strongly. For the tested parameters and input instances, the metrics correlate only weakly. The reason for the weak agreement of the metrics might be manifold, beginning with poorly chosen parameters.

To better evaluate the metrics's disagreement, the correlation between symmetry and other graph properties is measured (see Fig. 4 for scatter plots and TABLE II for Pearson's correlation factors). In total, six different measures are compared, which are defined as follows.

- *Nodes*: This measure is the number of nodes in the graph, i.e., n.
- *Edges*: The defines the total number of edges in the graph, i.e., m.
- *Density*: The density is defined as the percentage of edges present compared to the maximum amount of possible edges, i.e., $\frac{m}{n^2}$.
- *Bounding box*: This defines the size of the smallest axes-aligned bounding box containing the graph.
- *Convex hull*: This defines the area of the convex hull of the graph.
- *Concentration*: The concentration of a graph as defined by Taylor and Rodgers [1] measures how evenly distributed nodes are in the plane.

With a correlation factor of 0.70 (compared to 0.0008 for the KMP metric), the Purchase metric tends to rate denser graphs higher. This can be explained by the increasing number of symmetric subgraphs detected exceeding the threshold number of nodes. Further, the Purchase metric negatively correlates with both the size of an axis-aligned bounding box as well as the size of the convex hull of the graph. This is expected as the metric is scaled by the size of the graph (see Eq. 3).

[3]See https://networkx.org/ for details on networkX.

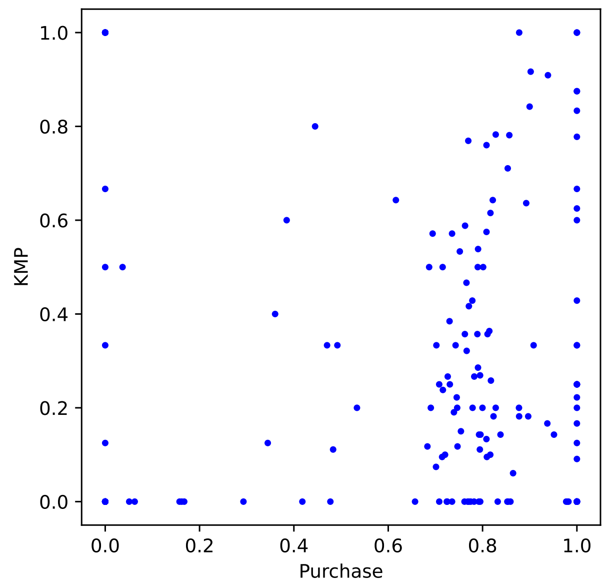


Fig. 3. Correlation between the Purchase and the KMP metric depicted in a scatter plot.

TABLE II
PEARSON'S CORRELATION FACTORS BETWEEN DIFFERENT METRICS AND THE PURCHASE SYMMETRY, THE KMP SYMMETRY AS WELL AS THE RATIO OVER BOTH SYMMETRY METRICS.

Metric	Purchase	KMP	Ratio
Nodes	0.0639	-0.0071	0.0784
Edges	0.4969	0.0787	0.2249
Density	0.6982	0.0008	0.1343
Bounding box	-0.4223	0.1424	0.0915
Convex Hull	-0.4041	0.1168	0.0230
Concentration	-0.1598	-0.1400	0.0248
Crossings	0.1957	0.1857	0.2342

The KMP metric exhibits no strong correlation between any of the investigated graph measures. Only a slight correlation is detectable for the size of the bounding box and the convex hull, with a Pearson's factor of 0.14 and 0.12 respectively. No clear correlation is visible for the rest of the graph measures. Similarly, the number of nodes and the number of crossings only correlate weakly with the measured reflective symmetry. When looking at the ratio of the Purchase metric over the KMP metric, no strong correlations are detectable.

C. Accuracy and robustness

The metrics have previously been compared in the context of a user study by Welch and Kobourov [15]. They concluded

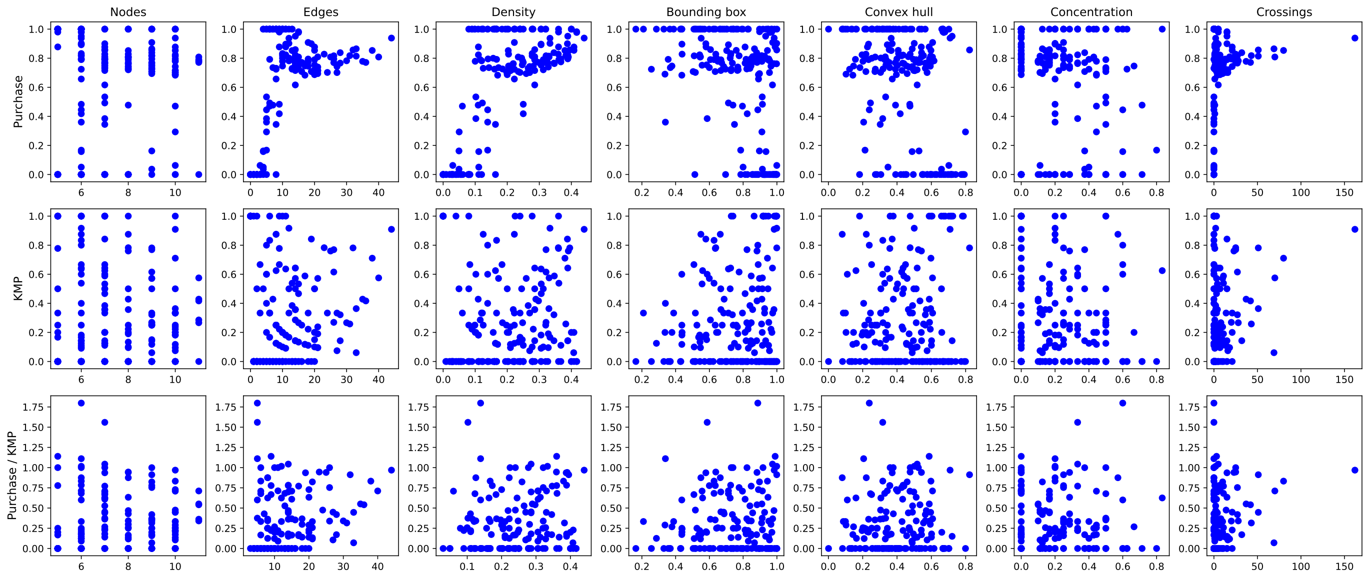


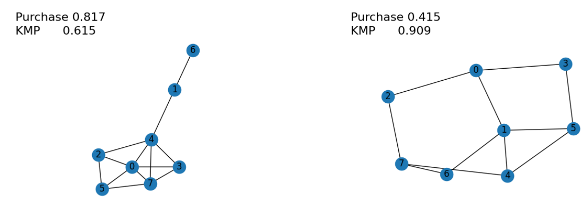
Fig. 4. Correlation between the symmetry metrics and other graph properties.

that participants agree more with the judgements of the Purchase metric than the KMP metric. In addition, they claim the following flaws:

- The Purchase metric generates a large number of axes, more than a human is likely to perceive.
- Both metrics are not robust to scale, meaning that rescaling a graph changes the symmetry metric. While rescaling can be offset by the threshold parameter for the Purchase metric, this is not immediately possible for the KMP metric.
- Neither metric is resilient to noise, meaning small adjustments might change the predicted symmetry vastly. This can be explained by the hard cutoff when two points are considered close enough to be equal. The authors suggest using a smooth loss function instead.

The claims are extended by the following statements:

- When investigating graphs where the two metrics strongly disagree, it can be observed that the Purchase metric tends to rate graphs with a higher node-to-convex-hull proportion higher (see Fig. 5). This is likely due to a smaller area leading to a smaller denominator in Eq. 3, resulting in an overall higher score, as discussed in Section IV-B.
- The KMP metric does not provide a solution to resolve conflicts between axes with equal scores. As the approach generates m axes with the maximum score of 1, equal scores might occur often – especially if little to none of the axes are grouped. This leads to varying results for the same input instance depending on the chosen axis.
- As crossings are ignored in the KMP metric, overlayed symmetric structures are still considered symmetric (see Fig. 7). This issue is resolved by the Purchase metric as crossings are converted to nodes beforehand.



(a) (b)

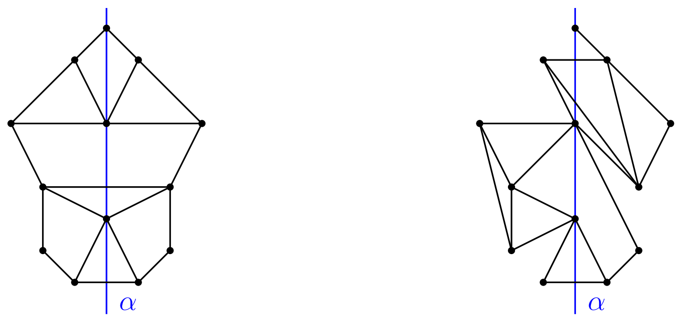
Fig. 5. Two graphs, (a) one where the Purchase metric is higher than the KMP metric, and (b) one where the KMP metric is higher than the Purchase metric.

- Per definition, the Purchase metric is mostly concerned with node positions. It only asserts if edges are present in a set of reflectively symmetric nodes, not considering between what specific nodes they are present. This leads to scenarios where completely different subgraphs – only sharing the same node positions and number of edges – lead to the same axis score as both have the same convex hull area and axis-specific symmetry value s_α. See Fig. 6 for an example.

V. Evaluation

Both metrics can be considered rather expensive regarding their asymptotic runtime. Purchase claims an asymptotic runtime of $O(n^7)$ for their metric. The KMP metric can be implemented in $O(m^2 \cdot N)$ time. The differences in theoretic runtime are observable in the runtime experiments. Even for dense graphs, the KMP metric is considerably faster.

The presented metrics have their unique set of advantages and disadvantages. Neither metric is stable in regard to the chosen parameter values, making the selection of appropriate



(a) (b)

Fig. 6. Two symmetric subgraphs along an axis G_α with the same vertex positions and the same number of edges.

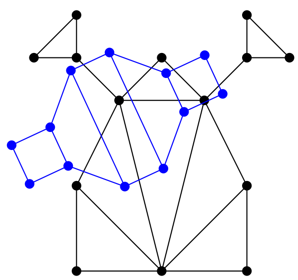


Fig. 7. Two reflectively symmetric graphs overlayed to build a non-symmetric combined drawing.

values a non-trivial task. The Purchase metric incorporates a multitude of axes into the final score than any human is likely to perceive. The KMP metric, on the other hand, is not robust to scale.

To try to compensate for the disadvantages of each approach and begin to develop an improved metric, it is beneficial that both metrics are similar in their modularisation into distinct steps. First, a set of potential axes is calculated, after which an axis-specific score is obtained. Then, axes considered irrelevant are dropped. As a final step, a global symmetry score is obtained from the axes that are left. Both metrics have their unique advantages and drawbacks at each step. The modularisation makes it possible to choose ideas from the approach best suited to a specific use case at any given step. For example, pre-processing the crossings can be done for the KMP metric as well. Further, the Purchase metric might benefit from borrowing the idea of merging close axes or only picking the top N axes at the end.

A study by De Luca et al. suggests that vertical reflective axes are most discernible, followed by horizontal reflective axes and then translational symmetries [5]. A symmetry metric aimed to mimic human perception should prioritise these specific symmetries accordingly. However, neither of the presented metrics incorporates the angle, position or type of an axis into the score. Additionally, a human is unlikely to perceive the three types of symmetries independently, suggesting a need for a methodology that integrates all symmetry types into a unified metric.

VI. Conclusion

Both metrics are improvable in their accuracy in detecting symmetry. In the randomised dataset used, the two metrics did not agree on symmetry, suggesting that they are not capable of predicting symmetry reliably. While the Purchase metric outperformed the KMP metric in a user study [15], it is only capable of detecting reflective symmetry. To further investigate the unique challenges of both metrics, larger datasets and a defined approach to obtain metric parameters are needed.

In conclusion, further research is needed to investigate and improve upon the discussed metrics. Both metrics are adaptable in each step of the process to try to improve upon the identified flaws.

References

[1] M. Taylor and P. Rodgers, "Applying Graphical Design Techniques to Graph Visualisation," in *Ninth International Conference on Information Visualisation (IV'05)*, London, England: IEEE, 2005, pp. 651–656. doi: 10.1109/IV.2005.19.

[2] C. Bennett, J. Ryall, L. Spalteholz, and A. Gooch, "The aesthetics of graph visualization," in *Computational Aesthetics in Graphics '07*, 2007, pp. 57–64. doi: 10.2312/COMPAESTH/COMPAESTH07/057-064.

[3] H. Purchase, "Metrics for Graph Drawing Aesthetics," in *Journal of Visual Languages & Computing*, vol. 13, no. 5, pp. 501–516, Oct. 2002. doi: 10.1006/ jvlc.2002.0232.

[4] H. Purchase, "Which aesthetic has the greatest effect on human understanding?" in *Graph Drawing*, G. Goos, J. Hartmanis, J. van Leeuwen, and G. DiBattista, Eds., vol. 1353, Series Title: Lecture Notes in Computer Science, Berlin, Heidelberg: Springer Berlin Heidelberg, 1997, pp. 248–261, doi: 10.1007/3-540-63938-1_67.

[5] F. De Luca, S. Kobourov, and H. Purchase, "Perception of Symmetries in Drawings of Graphs," in *Graph Drawing and Network Visualization*, vol. 11282, pp. 433–446, 2018. doi: 10.1007/978-3-030-04414-5_31.

[6] S. Kieffer, T. Dwyer, K. Marriott, and M. Wybrow, "HOLA: Human-like orthogonal network layout," in *IEEE Transactions on Visualization and Computer Graphics*, 22(1), pp. 349-358, 2015. doi: 10.1109/TVCG.2015.2467451

[7] K. Marriott, H. Purchase, M. Wybrow, and C. Goncu, "Memorability of visual features in network diagrams," in *IEEE Transactions on Visualization and Computer Graphics*, 18(12), pp. 2477-2485, 2012. doi: 10.1109/TVCG.2012.245.

[8] F. Ball and A. Geyer-Schulz, "How Symmetric Are Real-World Graphs? A Large-Scale Study," Symmetry-Basel, vol. 10, no. 1, Jan. 2018. doi: 10.3390/sym10010029.

[9] T. Xu, J. Yang, and G. Gou, "A Force-Directed Algorithm for Drawing Directed Graphs Sym- metrically," *Mathematical Problems in Engineering*, vol. 2018, 2018. doi: 10. 1155/2018/6208509.

[10] A. Meidiana, S.- H. Hong, P. Eades, and D. Keim, "Automorphism Faithfulness Metrics for Symmetric Graph Drawings," in *IEEE Transactions on Visualization and Computer Graphics*, pp. 1–15, 2022. doi: 10.1109/TVCG.2022.3229354.

[11] G. Sabidussi, "On the minimum order of graphs with given automorphism group," in *Monatshefte für Mathematik*, 63(2), pp. 124-127, 1959. doi: 10.1007/BF01299094.

[12] R.J. Lipton, S.C. North, J.S. Sandberg, "A Method for Drawing Graphs," in *1st ACM Symposium on Computational geometry (SoCG)*, pp. 153–160, 1985. doi: 10.1145/323233.323254.

[13] R. Klapaukh, S. Marshall, and D. Pearce, "A Symmetry Metric for Graphs and Line Diagrams," in *Diagrammatic Representation and Inference*, vol. 10871, Series Title: Lecture Notes in Computer Science, Cham: Springer International Publishing, 2018, pp. 739–742. doi: 10.1007/978-3-319-91376-6_71.

[14] G. Loy and J.-O. Eklundh, "Detecting Symmetry and Symmetric Constellations of Features," in *Computer Vision – ECCV 2006*, vol. 3952, Series Title: Lecture Notes in Computer Science, Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, pp. 508–521. doi: 10.1007/11744047_39.

[15] E. Welch and S. Kobourov, “Measuring Symmetry in Drawings of Graphs,” in *Computer Graphics Forum*, vol. 36, no. 3, pp. 341–351, Jun. 2017. doi: 10.1111/cgf.13192.
[16] F. De Luca, M. I. Hossain, and S. Kobourov, “Symmetry Detection and Classification in Drawings of Graphs,”, in *Graph Drawing and Network Visualization*, vol. 11904, Series Title: Lecture Notes in Computer Science, Cham: Springer International Publishing, pp. 499–513, 2019. doi: 10.1007/978-3-030-35802-0_38.