Evolutionary Edge Bundling as a Large-Scale Multi-Objective Optimization Problem: A Comparative Study

Chihiro Noda and Ryosuke Saga
Graduate School of Informatics, Osaka Metropolitan University
Osaka, Japan
Email: sp25491p@st.omu.ac.jp, r.saga@omu.ac.jp

***Abstract*—Evolutionary edge bundling reduces visual clutter in network visualization by optimizing edge geometries under multiple objectives, yet the optimizer's contribution is still poorly understood. We present a controlled comparison of five representative multi-objective evolutionary algorithms—MOEA/D, SMPSO, NSGA-II, NSGA-III, and SPEA2—within a unified edge-bundling pipeline under an identical computational budget. Fixing the control-point representation, objective functions, datasets, and evaluation protocol allows observed differences to be attributed mainly to optimization behavior. Experiments on one synthetic network and two real-world airline networks highlight optimizer-dependent trade-offs among convergence speed, Pareto-set coverage, and visual diversity, offering practical guidance for LSGO-type edge-bundling tasks.**

***Index Terms*—Evolutionary Edge Bundling, Large-Scale Global Optimization, Multiobjective Evolutionary Algorithms, Network Visualization, Benchmarking.**

I. Introduction

Large-Scale Global Optimization (LSGO) is a core challenge in evolutionary computation, where algorithmic scalability under thousands of decision variables becomes a primary concern. In such settings, scalable optimizers and fair, controlled evaluation across algorithms are essential.

Network visualization provides a practical LSGO-motivated application domain. It is widely used in areas such as social science, social network analysis, and geographic information science [1] [2] [3]. However, as networks grow in size and density, overlapping edges lead to severe visual clutter, degrading both aesthetics and the user's ability to interpret meaningful patterns.

Edge bundling is an effective post-processing technique to mitigate edge clutter by aggregating spatially or structurally similar edges into shared trajectories while keeping node positions fixed. Several methods have demonstrated strong clutter reduction [4], [5]. Nevertheless, many bundling approaches rely on method-specific parameters and heuristics, which makes objective optimization and systematic evaluation difficult.

To address these limitations, evolutionary approaches to edge bundling have been investigated, where edge geometries are optimized under multiple objectives without relying on manual tuning. Saga et al. proposed a GA-based edge bundling method [6] that achieves effective bundling while maintaining edge-flow independence, and the availability of diverse multi-objective optimizers—including genetic algorithms [7], [8] and particle swarm optimization [9], [10]—motivates the exploration of alternative optimizers for this task.

Importantly, evolutionary edge bundling exhibits an LSGO-type property because the decision-vector dimensionality increases with network size. With cp control points per edge, the decision vector scales as $N = |E| \times cp$, reaching $N = 6303$ on the US airline dataset ($|E| = 2101$, $cp = 3$, Section IV-A3). This large-scale nature, together with the multi-objective formulation, makes optimizer choice and scalability particularly critical.

Despite increasing interest in evolutionary edge bundling, most existing studies present end-to-end methods that tightly combine the optimizer with specific representations, objective formulations, parameter settings, and datasets. Consequently, controlled comparisons across different multi-objective evolutionary algorithms—where non-optimizer factors are fixed to isolate optimizer-dependent behavior—remain scarce.

In this work, we address this gap by conducting a controlled comparison of representative multi-objective evolutionary algorithms for edge bundling under an identical optimization pipeline. Our contributions are as follows: we establish a unified pipeline that fixes non-optimizer factors to enable fair evaluation in an LSGO-type setting; we isolate and analyze optimizer-dependent characteristics using both quantitative objective metrics and qualitative inspection of the resulting visualizations; and we provide reproducible results on synthetic and real-world networks. The source code, output visualizations, and experimental data are publicly available at https://doi.org/10.5281/zenodo.19571473 to support reproducibility and future comparative studies.

II. Related works

A. Benchmarking perspective in large-scale multi-objective optimization (LSGO)

Large-scale optimization is commonly evaluated through benchmarking, where algorithms are compared under standardized suites and fixed evaluation budgets [11], [12]. This is important in large-scale multi-objective optimization, because differences in representation or evaluation protocols can confound optimizer comparisons [13].

From this viewpoint, evolutionary edge bundling is an LSGO-type optimization problem. When edges are parameterized by cp control points, the decision space scales as $|E| \times cp$, readily reaching thousands of variables in real-world networks.

However, unlike the LSGO community where benchmark suites and comparison protocols are explicitly defined [11], [12], a standardized benchmarking framework for evolutionary edge bundling under such large-scale conditions has not been established, leaving its optimizer-level characteristics insufficiently clarified.

B. Evolutionary edge bundling and the lack of controlled cross-optimizer comparisons

Evolutionary edge bundling formulates edge bundling as an optimization problem and searches for suitable edge geometries through evolutionary computation, offering flexibility when multiple objectives must be balanced.

Saga et al. proposed a GA-based evolutionary edge bundling method (GABEB), introducing quantitative evaluation metrics and demonstrating that effective edge bundling can be achieved by optimizing control point displacements along edges [6]. This direction was extended to simultaneously optimize node layout and edge bundling within a unified GA-based approach [14], [15]. GP-GPU computation was further incorporated to accelerate evolutionary edge bundling, enabling larger-scale optimization [16]. A Bézier-curve representation was also integrated into the evolutionary process to alleviate zigzag artifacts in bundled edge geometries[17].

Despite these advances, prior studies often change several components simultaneously, making it difficult to isolate optimizer-dependent behavior. A controlled study that fixes the representation, objectives, datasets, and evaluation protocol while varying only the optimizer is therefore needed.

C. Positioning of this work

To address this gap, we conduct a controlled comparison of representative multi-objective evolutionary algorithms under an identical edge-bundling pipeline, attributing observed differences to optimizer behavior rather than method-specific design choices. Because edge bundling is a visualization task, we complement quantitative metrics with qualitative inspection of the resulting layouts.

III. Algorithm

Figure 1 shows the overall procedure of evolutionary edge bundling used in this study, which follows the general pipeline adopted in GABEB. Control-point displacements on each edge are optimized by evolutionary multi-objective algorithms under quantitative evaluation metrics.

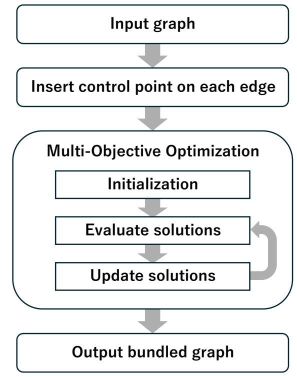

Fig. 1. Overall algorithm pipeline

A. Multi-objective optimization

Edge bundling involves multiple conflicting objectives, such as preserving original edge geometry while enhancing visual compactness and structural clarity. Therefore, this study formulates edge bundling as a multi-objective optimization problem and applies representative evolutionary multi-objective algorithms under a common pipeline. We selected five representative algorithms spanning four paradigms: MOEA/D (decomposition-based), SMPSO (particle swarm-based), NSGA-II and NSGA-III (non-dominated sorting-based), and SPEA2 (strength/indicator-based).

1) MOEA/D: MOEA/D, proposed by Zhang and Li, is a decomposition-based multi-objective evolutionary algorithm that solves a multi-objective optimization problem by decomposing it into a set of scalar subproblems [18]. Each subproblem is defined by a weight vector and is responsible for a specific region of the Pareto front.

2) SMPSO: SMPSO is a particle swarm optimization (PSO)-based multi-objective optimization algorithm proposed by Nebro et al. [19]. Each particle explores the solution space by updating its velocity and position, while the set of non-dominated solutions is maintained in an external archive.

3) NSGA-II: NSGA-II is a widely used Pareto-based multi-objective evolutionary algorithm proposed by Deb et al. [20]. It employs non-dominated sorting to rank solutions and crowding distance to maintain diversity within each front.

4) NSGA-III: NSGA-III is an extension of NSGA-II proposed by Deb et al., which introduces reference points to control solution diversity [21]. This mechanism is designed to distribute solutions across the entire Pareto front, particularly when the number of objectives is large.

5) SPEA2: SPEA2 is an indicator-based multi-objective evolutionary algorithm proposed by Zitzler et al. [22]. It evaluates solutions using a fitness assignment that combines dominance relationships with local density information and maintains non-dominated solutions in an external archive.

B. Representation

In order to perform edge bundling on multi-objective optimization methods, we inserted dividing points, referred to as control points, on each edge. As illustrated in Figure 2, these control points divide each edge into a fixed number of segments. The number of inserted control points is determined by a parameter cp. This divides each edge into $cp+1$ segments. Each control point is assigned a displacement vector d which

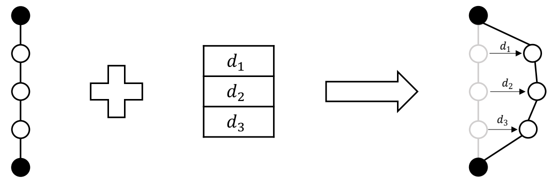

Fig. 2. Concept of the control points insertion

controls its movement perpendicular to the original edge. By optimizing these displacements, we simulate edge deformation and achieve edge bundling.

C. Fitness functions

Within the evolutionary edge bundling pipeline used in this study, four fitness functions introduced by Saga [6], [23] are employed.

1) Mean Edge Length Difference(MELD): MELD measures the average change in edge length before and after bundling. This metric quantifies how much each edge's length changes from its original state. MELD is defined by the following equation:

$$MELD = \frac{1}{|E|}\sum_{e\in E}|L'_e - L_e| \tag{1}$$

where E is the set of all edges, L_e is the original length of edge e, and $L'_e$is its length after bundling. MELD is a minimization objective; smaller values indicate that the change in edge lengths is limited, thereby maintaining the original structure of the network.

2) Mean of Occupation Area(MOA): MOA measures the average area occupied by edges after bundling. When edge bundling is effectively performed, the total occupied area is reduced, resulting in a more compact and efficient visualization. MOA is defined by the following equation:

$$MOA = \frac{1}{|E|}\left|\bigcup_{e\in E} O(e)\right| \tag{2}$$

where E is the set of all edges, *O(e)* is the set of unit area through which edge e passes. MOA is a minimization objective; lower values indicate that the edges are bundled more efficiently and that the overall visual clarity is improved.

3) Edge Density Distribution(EDD): EDD measures the distribution of edge density after bundling. A higher EDD value indicates a more uneven distribution of edge density, highlighting the contrast between dense and sparse area. EDD is defined by the following equation:

$$EDD = \frac{1}{|P|}\sum_{p\in P}(H(p) - H)^2 \tag{3}$$

where P is the set of all pixels, $H(p)$ is the number of edges passing through pixel p, and His the average value of $H(p)$ over all pixels. EDD is a maximization objective.

4) Path Quality(PQ): PQ measures the degree of zigzagging in edges after bundling. A higher PQ value indicates that the bundled edges show more zigzag patterns, whereas a lower value indicates that the edges curve smoothly in a consistent direction. PQ is defined by the following equation:

$$PQ = \sum_{e\in E}(-\sum_{i=3}^{m}\gamma_i|\Delta_i|) \tag{4}$$

$$\Delta_i = \begin{cases} A_i - A_{i-1} & \text{if } -\pi < |A_i - A_{i-1}| < \pi \\ |A_i - A_{i-1}| - 2\pi & \text{if } |A_i - A_{i-1}| > \pi \\ 2\pi + |A_i - A_{i-1}| & \text{if } |A_i - A_{i-1}| < -\pi \end{cases} \tag{5}$$

$$\gamma_i = \begin{cases} 0 & \text{if } \mathrm{sgn}(\Delta_i) = \mathrm{sgn}(\Delta_{i-1}) \\ 1 & \text{if } \mathrm{sgn}(\Delta_i) \neq \mathrm{sgn}(\Delta_{i-1}) \end{cases} \tag{6}$$

where E is the set of all edges, m is the number of segments divided by the control point, and A_i is the angle between the original edge and its i-th segment. PQ is a minimization objective.

IV. Experiments

All algorithms share the same edge representation, control-point insertion scheme, and objective functions, ensuring that observed differences arise solely from the optimization strategies. Experiments are conducted on three network datasets of varying sizes, with 20 independent runs per setting. The datasets, parameter configurations, and evaluation metrics are detailed below.

A. Dataset

The original graphs of the datasets used in the experiments before bundling are shown in Figure 3.

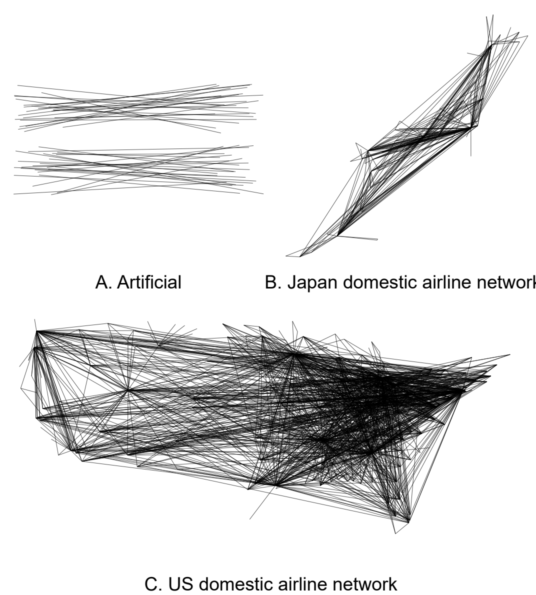

Fig. 3. Original layouts of the artificial network, Japan domestic airline network, and US domestic airline network.

1) Artificial dataset: The artificial network is generated for this study and has a bipartite-like structure, in which the node set is divided into two groups and each edge connects a node in one group to the other. The network consists of 100 nodes and 50 edges.

2) Japan domestic airline network: The Japan domestic airline network is constructed from domestic airline routes in Japan. In this network, nodes represent airports and edges represent scheduled flight routes between airports. The network consists of 79 nodes and 233 edges, forming a real-world flight network with geographically constrained routes.

3) US domestic airline network: The US domestic airline network consists of 235 nodes and 2101 edges, making it the largest and densest network in our experiments.

B. Parameters

Parameter settings follow standard values from the literature, summarized in Table I. The mutation probability was set to $1/N$, where N denotes the number of decision variables.

C. Performance Evaluation

Performance is evaluated from three aspects: qualitative inspection of the resulting visualizations, convergence behavior (tracked as the best objective value per generation), and final Pareto set quality measured by the Hypervolume (HV) indicator [24] computed from non-dominated solutions at the final generation.

V. Results

A. Qualitative observation

Representative bundling results on the US domestic airline network are shown in Figure 4; output images for one run per configuration are publicly available at https://doi.org/10.5281/zenodo.19571473. We observe that NSGA-II, NSGA-III, and SPEA2 tend to produce visually coherent bundles, while their solution sets show limited variation in appearance.

In contrast, MOEA/D and SMPSO generate a wider variety of solutions; however, many of them exhibit either weak bundling effects or edge layouts close to the original straight-line drawings. These observations suggest a trade-off between visual coherence and solution diversity among the compared methods.

TABLE I
PARAMETER SETTINGS FOR EACH OPTIMIZATION METHOD

Parameter	MOEA/D	NS-II	NS-III	SMPSO	SPEA2
Pop./Swarm size	500	500	500	500	500
Offspring size	–	500	500	–	500
Iterations	750	750	750	750	750
Max evals	375k	375k	375k	375k	375k
Objectives	4	4	4	4	4
Control points	3	3	3	3	3
Crossover	DE	SBX	SBX	–	SBX
Crossover prob.	CR=1.0	0.9	0.9	–	0.9
Crossover param.	F=0.5	η_c=20	η_c=20	–	η_c=20
Mutation	Poly.	Poly.	Poly.	Poly.	Poly.
Mutation prob.	$1/N$	$1/N$	$1/N$	$1/N$	$1/N$
Mutation η_m	20	20	20	20	20
Nbr. size	20	–	–	–	–
Nbr. sel. prob.	0.9	–	–	–	–
Max repl.	2	–	–	–	–
Weight seed	42	–	–	–	–
# of ref. direction	–	–	12	–	–
Archive size	–	–	–	20	–

B. Convergence Behavior

Figure 5 shows the convergence behavior of the four objectives for each optimization method on the US domestic airline network. The lines represent the mean values over 20 independent runs, and the semi-transparent bands indicate the standard deviation. Clear differences in optimization dynamics can be observed across the algorithms. MOEA/D and SMPSO exhibit rapid early convergence, with most improvements occurring within the first 50 generations, whereas NSGA-II, NSGA-III, and SPEA2 continue to improve gradually over later generations.

Regarding individual objectives, SMPSO quickly drives MELD and PQ to very small values, while NSGA-II/III and SPEA2 show continued improvements in MOA and EDD during the later stages of optimization. Notably, MOEA/D exhibits an opposite trend in EDD, deteriorating over generations. This is likely due to PQ dominating the scalarized subproblems, as discussed in the Limitations subsection. This indicates that different algorithms tend to prioritize different objectives during the search process. Such differences also imply distinct practical usage profiles: MOEA/D and SMPSO are advantageous in interactive or exploratory settings where short waiting times matter, whereas NSGA-II/III and SPEA2 are preferable when longer runtimes are acceptable to obtain higher-quality final Pareto sets.

When increasing the displacement range from $d = 30$ to $d = 50$, EDD generally improves, while MELD and PQ tend to deteriorate, especially for NSGA-II/III and SPEA2. This suggests that a larger displacement range provides greater flexibility for enhancing edge concentration, but makes it more challenging to preserve edge smoothness and regularity.

C. Final Pareto Quality

Table II summarizes the mean Hypervolume (HV) values of the final Pareto solution sets obtained by each algorithm over 20 independent runs. Prior to the HV calculation, all objective values were normalized to the range [0.0, 1.0] using min–max scaling based on the global minimum and maximum values across all algorithms and runs for each dataset and displacement range d.

To assess the statistical significance of the performance differences, a Wilcoxon rank-sum test with a significance level of $\alpha = 0.05$ was conducted. In each configuration, the algorithm with the highest mean HV was compared against the others, and those performing significantly worse are denoted by a dagger (†) in the table.

Across most datasets and displacement ranges, NSGA-II and NSGA-III achieve higher HV values than the other methods, indicating better overall quality of the final solution sets. This tendency is particularly pronounced on the airline network datasets (Datasets B and C).

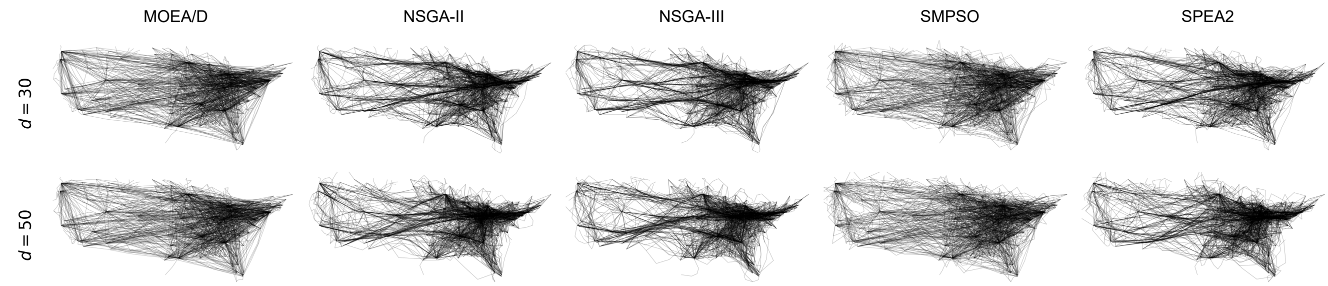

Fig. 4. Qualitative comparison of edge-bundling results produced by five optimization methods

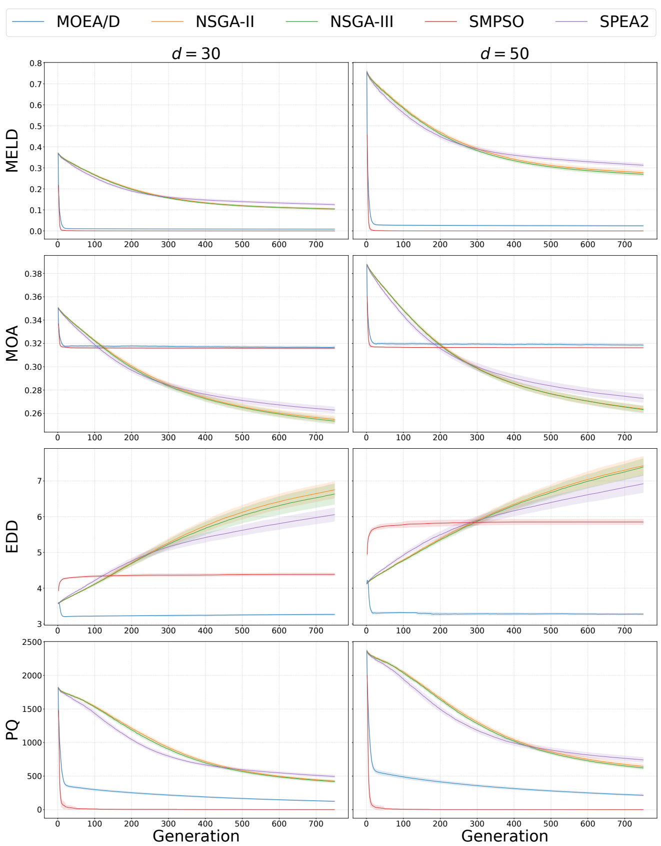

Fig. 5. Convergence behavior of four quantitative edge-bundling objectives (MELD, MOA, EDD, and PQ) over generations on the US domestic airline network.

More specifically, NSGA-III achieves the best HV in four out of six conditions (A30, A50, B30, and B50), while NSGA-II is best in the remaining two conditions (C30 and C50), suggesting that these two algorithms consistently dominate the final Pareto quality across the tested settings.

In contrast, MOEA/D and SMPSO consistently yield lower HV values under all experimental conditions. These results suggest that, despite their fast convergence in individual objectives, the final solution sets produced by these methods exhibit relatively limited coverage of the objective space.

TABLE II
HYPERVOLUME (HV) VALUES OF FINAL PARETO SOLUTION SETS FOR DATASET A (ARTIFICIAL), B (JAPAN DOMESTIC AIRLINE NETWORK), AND C (US DOMESTIC AIRLINE NETWORK)

Data	d	MOEA/D	NSGA-II	NSGA-III	SMPSO	SPEA2
A	30	0.0375 †	0.8591	**0.9431**	0.0465 †	0.7816 †
A	50	0.0259 †	0.7777	**0.7836**	0.0320 †	0.6392 †
B	30	0.0213 †	0.6504	**0.6830**	0.0363 †	0.4786 †
B	50	0.1062 †	1.2176	**1.2626**	0.1270 †	1.0184 †
C	30	0.0991 †	**0.9527**	0.9356	0.1244 †	0.7134 †
C	50	0.1215 †	0.9038	**0.9094**	0.1877 †	0.7365 †

D. Limitations

This study has two notable limitations. First, the experiments are restricted to geographic graphs in which node positions are fixed by real-world coordinates (airline routes). Because the optimization pipeline does not handle node placement, it cannot be directly applied to graphs with algorithmically determined layouts. Second, MOEA/D exhibited anomalous behavior in which EDD deteriorated over generations while PQ reached near-zero values rapidly. This is likely a consequence of PQ being a non-scalable metric: its magnitude is not commensurate with the other objectives, causing it to dominate the scalarized subproblems in MOEA/D and redirecting search pressure away from EDD. The issue highlights a broader challenge when applying decomposition-based algorithms to problems with heterogeneous objective scales: careful normalization or scale-aware scalarization is required to prevent any single objective from monopolizing the search.

VI. CONCLUSION

We conducted a controlled comparison of multi-objective evolutionary algorithms for evolutionary edge bundling under a unified optimization pipeline. Fixing the control-point representation ($cp = 3$), using four objectives (MELD, MOA, EDD, and PQ), and testing two displacement ranges ($d = 30$ and $d = 50$), we evaluated five optimizers (MOEA/D, SMPSO, NSGA-II, NSGA-III, and SPEA2) on three datasets, from an artificial bipartite-like network to real-world airline networks.

Across conditions, the optimizers showed distinct behavior. MOEA/D and SMPSO improved quickly in the early stage (typically within the first 50 generations) and often advanced particular objectives such as MELD and PQ sooner. However, their final Pareto sets tended to have smaller hypervolume, suggesting that the obtained solutions covered the objective space less broadly. In contrast, NSGA-II and NSGA-III delivered stronger final Pareto sets in most settings, especially on the airline datasets. For example, NSGA-III achieved the best HV on the artificial and Japan domestic airline networks, while NSGA-II led on the US domestic airline network. SPEA2 was also competitive, though it generally fell slightly behind NSGA-II/III.

Qualitatively, we observed a trade-off between visual coherence and variation. NSGA-II/III and SPEA2 often produced bundled layouts that were visually consistent but relatively similar to one another. MOEA/D and SMPSO, meanwhile, generated a broader range of solutions, including many with weak bundling or near straight-line layouts.

We also found that increasing the displacement range can strengthen edge concentration (EDD), but it frequently comes at the cost of geometry preservation and smoothness (MELD and PQ). This highlights the need to choose d based on the intended visualization goal. Overall, Pareto-based methods—particularly NSGA-II and NSGA-III—appear to be a reliable default when the priority is consistently high-quality Pareto sets, whereas MOEA/D and SMPSO may be preferable when early objective-wise progress or greater solution diversity is desired.

As future work, we will explore ways to better balance visual coherence and diversity, including parameter sensitivity studies, alternative archiving/diversity mechanisms, and metrics that more directly reflect perceptual readability. In particular, replacing PQ with a scalable alternative that maintains commensurate magnitude with the other objectives would help avoid objective-scale imbalance in decomposition-based methods. We also plan to extend the framework to graphs with algorithmically determined layouts, where node positions are not fixed a priori. In addition, we plan to extend the comparison to more optimizers and larger datasets.

Acknowledgments

The authors acknowledge the use of an AI-based language model for improving the clarity and grammar of the manuscript. The AI tool was used solely for language refinement and did not contribute to the research design, analysis, or conclusions. This work was supported by JSPS KAKENHI Grant Number JP22K12116.

References

[1] R. Saga, N. Ohkusa, T. Yamashita, and N. Maki, "Visualization of customer expectations from web text using co-occurrence graph and autolabeling in the service market," in *Proceedings of the 12th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications - Volume 3: IVAPP (VISIGRAPP 2017)*, 2017, pp. 317–321.

[2] R. Saga and T. Yagi, "Network visualization of customer expectation using web in coffee service," *Artificial Life and Robotics*, vol. 23, no. 2, pp. 213–217, Jun. 2018.

[3] R. Saga, M. Terachi, Z. Sheng, and H. Tsuji, "Fact-graph: Trend visualization by frequency and co-occurrence," in *Proceedings of the 31st Annual German Conference on Advances in Artificial Intelligence (KI'08)*. Kaiserslautern, Germany: Springer, 2008, pp. 308–315.

[4] D. Holten, "Hierarchical edge bundles: Visualization of adjacency relations in hierarchical data," *IEEE Transactions on Visualization and Computer Graphics*, vol. 12, no. 5, pp. 741–748, 2006.

[5] W. Cui, H. Zhou, H. Qu, P. C. Wong, and X. Li, "Geometry-based edge clustering for graph visualization," *IEEE Transactions on Visualization and Computer Graphics*, vol. 14, no. 6, pp. 1277–1284, 2008.

[6] R. Saga, T. Yoshikawa, K. Wakita, K. Sakamoto, G. Schaefer, and T. Nakashima, "A genetic algorithm optimising control point placement for edge bundling," in *Proceedings of the 15th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP 2020)*, 2020, pp. 217–222.

[7] D. E. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison-Wesley, 1989.

[8] C. M. Fonseca and P. J. Fleming, "An overview of evolutionary algorithms in multiobjective optimization," *Evolutionary Computation*, 1995.

[9] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of ICNN'95 - International Conference on Neural Networks*, vol. 4, 1995, pp. 1942–1948.

[10] C. A. Coello Coello and M. S. Lechuga, "Mopso: a proposal for multiple objective particle swarm optimization," in *Proceedings of the 2002 Congress on Evolutionary Computation (CEC'02)*, vol. 2, 2002, pp. 1051–1056.

[11] K. Tang, X. Li, P. N. Suganthan, Z. Yang, and T. Weise, "Benchmark functions for the cec'2010 special session and competition on large-scale global optimization," Jan. 2010.

[12] X. Li, K. Tang, M. N. Omidvar, Z. Yang, and K. Qin, "Benchmark functions for the cec'2013 special session and competition on large-scale global optimization," Jan. 2013.

[13] A. LaTorre, S. Muelas, and J.-M. Peña, "A comprehensive comparison of large scale global optimizers," *Information Sciences*, vol. 316, pp. 517–549, 2015.

[14] J. Meikari and R. Saga, "Evolutionary node layout and edge bundling," in *2022 IEEE Congress on Evolutionary Computation (CEC)*, Padua, Italy, 2022, pp. 1–6.

[15] ——, "Simultaneous optimization of edge bundling and node layout using genetic algorithm," in *Proceedings of the 19th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP 2024) - Volume 1: GRAPP, HUCAPP and IVAPP*. SCITEPRESS – Science and Technology Publications, Lda., 2024, pp. 741–748.

[16] N. Hashimoto and R. Saga, "Simultaneous node layout and edge bundling using multi-objective optimization with gpgpu-based ga," in *2024 28th International Conference Information Visualisation (IV)*, 2024, pp. 1–6.

[17] C. Noda and R. Saga, "Ga-based edge bundling with bézier-curve representations," in *Proceedings of the Thirty-first International Symposium on Artificial Life and Robotics 2026 (AROB 31st 2026) and Eleventh International Symposium on Biocomplexity 2026 (ISBC 11th 2026)*, Jan. 2026, pp. 358–363.

[18] Q. Zhang and H. Li, "Moea/d: A multiobjective evolutionary algorithm based on decomposition," *IEEE Transactions on Evolutionary Computation*, vol. 11, no. 6, pp. 712–731, 2007.

[19] A. J. Nebro, J. J. Durillo, J. Garcia-Nieto, C. A. Coello Coello, F. Luna, and E. Alba, "Smpso: A new pso-based metaheuristic for multi-objective optimization," in *2009 IEEE Symposium on Computational Intelligence in Multi-Criteria Decision-Making (MCDM)*, 2009, pp. 66–73.

[20] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: Nsga-ii," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 2, pp. 182–197, 2002.

[21] K. Deb and H. Jain, "An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part i: Solving problems with box constraints," *IEEE Transactions on Evolutionary Computation*, vol. 18, no. 4, pp. 577–601, 2014.

[22] E. Zitzler, M. Laumanns, and L. Thiele, "SPEA2: Improving the strength pareto evolutionary algorithm for multiobjective optimization," in *Evolutionary Methods for Design Optimization and Control with Applications to Industrial Problems*, K. C. Giannakoglou, D. T. Tsahalis, J. Périaux, K. D. Papailiou, and T. Fogarty, Eds. International Center for Numerical Methods in Engineering, 2001, pp. 95–100.

[23] R. Saga, "Quantitative evaluation for edge bundling based on structural aesthetics," in *EuroVis 2016 - Posters*, 2016.

[24] E. Zitzler and L. Thiele, "Multiobjective evolutionary algorithms: A comparative case study and the strength pareto approach," *IEEE Transactions on Evolutionary Computation*, vol. 3, no. 4, pp. 257–271, 1999.