

Preference-Optimal Multi-Metric Weighting for Parallel Coordinate Plots

Chisa Mori^{* 1}, Shuhei Watanabe^{* 2,3}, Masaki Onishi³, Takayuki Itoh¹

^{*} Equal Contribution

¹*Ochanomizu University, Tokyo, Japan*
{mori.chisa, itot}@is.ocha.ac.jp

²*Preferred Networks Inc., Tokyo, Japan*
shuheiwatanabe@preferred.jp

³*Advanced Industrial Science and Technology (AIST), Tokyo, Japan*
onishi-masaki@aist.go.jp

Abstract—Parallel coordinate plots (PCPs) are a prevalent method to interpret the relationship between the control parameters and metrics. PCPs deliver such an interpretation by color gradation based on a single metric. However, it is challenging to provide such a gradation when multiple metrics are present. Although a naïve approach involves calculating a single metric by linearly weighting each metric, such weighting is unclear for users. To address this problem, we first propose a principled formulation for calculating the optimal weight based on a specific preferred metric combination. Although users can simply select their preference from a two-dimensional (2D) plane for bi-metric problems, multi-metric problems require intuitive visualization to allow them to select their preference. We achieved this using various radar charts to visualize the metric trade-offs on the 2D plane reduced by UMAP. In the analysis using pedestrian flow guidance planning, our method identified unique patterns of control parameter importance for each user preference, highlighting the effectiveness of our method.

Index Terms—Visualization, High-Dimensional Data Visualization, Parallel Coordinate Plots, Real-World Application

I. INTRODUCTION

In the real world, decision makers often need to explain the relationship between the control parameters and their effects on the metrics of interest. For example, human flow simulation, which attempts to mimic human flow in a real environment, has various control parameters that must be adapted by decision makers such that the real human flow becomes idealistic [21]. Decision making is particularly challenging when many control parameters exist.

Parallel coordinate plots (PCPs) [5] are among the most prevalent methods for interpreting the relationship between potentially high-dimensional control parameters and metrics of interest. PCPs draw polyines where each polyline represents the control parameter values and their corresponding metric values. In particular, each polyline is colored based on a metric value such that decision makers can distinguish the pattern in the performant control

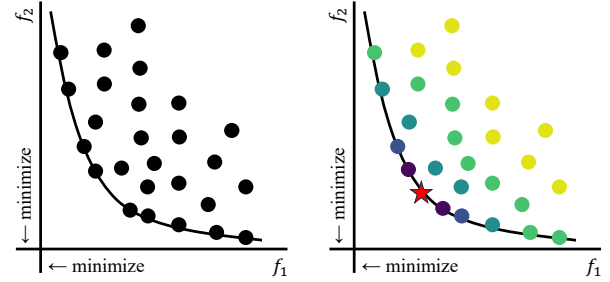


Fig. 1. Conceptual visualization of the preference-optimal weights in a bi-metric problem. For problems with three or more metrics, we provide additional visual support using radar charts as described in Section III. **Left:** A set of bi-metrics in the 2D plane. Black dots represent the observed bi-metrics, whereas the black line represents the approximated Pareto front. In principle, all the observations on the approximated Pareto front are equally performant considering multi-objective optimization. **Right:** Users provide their preference by selecting a point, e.g., the red star, on the approximated Pareto front. Our proposition, cf. Section III, enables calculating the optimal weights given the choice. For example, if the user selects the red star, each observation can be colored as in the right figure, where blue is better and yellow is worse under the user's choice. By doing so, we can rank the Pareto solutions based on the user preference.

parameters. However, practitioners often encounter multi-metric problems, which have multiple metrics to consider, and the coloring of PCPs is unclear owing to the lack of a unique metric, shown in Figure 1.

To this end, we propose a mathematical formulation to color PCPs for multi-metric problems based on a user preference. Our formulation calculates the optimal weights to obtain the weighted average of each metric, allowing us to color PCPs based on a single weighted metric. Although this formulation provides a principled way to calculate a single weighted metric in general problems, selecting an ideal trade-off between each metric is not trivial for users, particularly when more than two metrics are present. To address this problem, we provide visual support combined

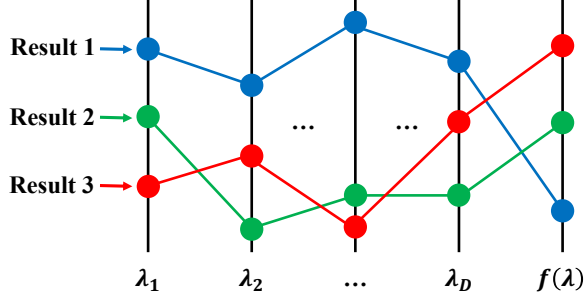


Fig. 2. Example of PCP. Each polyline shows a control parameter vector used in an evaluation and its metric values. Each vertical axis represents either a control parameter or a metric.

with dimension reduction and radar charts.

Through experiments, we demonstrated the effectiveness of the proposed algorithm using pedestrian flow simulation data. Our analysis identified the important control parameters for improving various trade-offs. Importantly, different trade-offs exhibit different patterns of important control parameters, showing the effectiveness of our method.

In summary, our contributions are as follows:

- 1) a novel formulation to calculate the optimal weights for a single weighted metric given user feedback,
- 2) intuitive visual support to allow users to select their preferences even for multi-metric problems using radar charts and UMAP, and
- 3) the demonstration using a pedestrian flow simulator that shows the effectiveness of our proposed method.

II. BACKGROUND

A. Multi-Metric Problems

A multi-metric problem is controlled by the parameter vector $\boldsymbol{\lambda} := [\lambda_1, \dots, \lambda_D] \in \Lambda := \Lambda_1 \times \dots \times \Lambda_D \subseteq \mathbb{R}^D$, where $\Lambda_d \subseteq \mathbb{R}$ denotes the domain of the d -th control parameter for $d \in [D] := \{1, 2, \dots, D\}$. Examples of the control parameters include the ratio of pedestrians to be allocated to a certain route and the start time of pedestrian movement in a pedestrian flow simulation. Each control parameter $\boldsymbol{\lambda}$ is evaluated using the metrics $\mathbf{f}(\boldsymbol{\lambda}) := [f_1, \dots, f_M] \in \mathbb{R}^M$ to minimize, where M denotes the number of metrics. For example, we may evaluate a simulation with the total duration of an evacuation and how crowded each route. This study aims to visualize some patterns of promising control parameters based on evaluated results, i.e., a set of control parameters and the corresponding metrics, $\{(\boldsymbol{\lambda}^{(n)}, \mathbf{f}^{(n)})\}_{n=1}^N$ where N denotes the number of evaluated control parameter vectors and $\mathbf{f}^{(n)} := \mathbf{f}(\boldsymbol{\lambda}^{(n)})$.

B. Parallel Coordinate Plots (PCPs)

Parallel coordinate plots (PCPs) [5] visualize each control parameter vector and the corresponding metric values

as a polyline. Figure 2 illustrates PCPs. Each polyline passes through each vertical axis, representing what value each control parameter and each metric had. Since performant results are often of interest, we color each polyline differently depending on how performant each polyline is. Although such colorization is straightforward for a single-metric problem, as PCPs can be simply colored by the metric, it is not trivial for a multi-metric problem. This motivated our study.

C. Pareto Solutions and Pareto Front

In practice, Pareto optimality defines optimality in multi-metric problems. Given a set of multiple metrics $\mathcal{F} := \{\mathbf{f}^{(n)}\}_{n=1}^N$, the Pareto solutions of the metric set is $\mathcal{P} := \{\mathbf{f} \in \mathcal{F} \mid \forall \mathbf{f}' \in \mathcal{F}, \mathbf{f}' \not\prec \mathbf{f}\}$ where $\mathbf{f}' \prec \mathbf{f}$ represents the dominance of $\mathbf{f}'$ over $\mathbf{f}$, implying $f'_m \leq f_m$ for all $m \in [M]$ and $f'_m < f_m$ for some $m \in [M]$. The Pareto front of $\mathcal{F}$ is the hypercurve where the Pareto solutions are supposed to be distributed. Watanabe [16] provides further details.

III. PARALLEL COORDINATE PLOTS FOR MULTI-METRIC PROBLEMS

In this section, we first discuss PCPs for bi-metric problems and generalize them to higher-dimensional problems. For both setups, we considered coloring PCPs based on user preference feedback and calculated a single weighted metric, i.e., a linear combination of each metric value, using the optimal weights computed from the user feedback. Although user feedback can be easily provided by clicking on a preferred solution on the 2D plane, it is not trivial to achieve this for multi-metric problems, leading to our novel proposition in the combination of UMAP and radar charts.

A. Preference Optimal Weight Calculation for Bi-Metric Problems

In principle, PCPs require a single metric to color each polyline. Therefore, we propose a method for calculating the weight of each metric. Specifically, we color each polyline based on a single weighted metric $\phi_{\mathbf{w}}(\boldsymbol{\lambda}) := \sum_{m=1}^M w_m f_m(\boldsymbol{\lambda})$ where $\mathbf{w} := \{w_m\}_{m=1}^M$ denotes the weight vector subject to $\sum_{m=1}^M w_m = 1$. We assume that the Pareto front $\mathcal{F}$ is represented as a strict convex function $g : [f_1^{\min}, f_1^{\max}] \rightarrow \mathbb{R}$ in the 2D plane where $f_1^{\min}, f_1^{\max}$ are the minimum and maximum of the first metric, and user feedback is provided in the form of $(f_1^u, g(f_1^u))$. Thus, the following holds:

Proposition 1. Assume that the optimal weights are defined such that a single metric $\phi_{\mathbf{w}^*}(\boldsymbol{\lambda})$ subject to $f_2 \geq g(f_1)$ achieves optimality at the user-specified point $(f_1^u, g(f_1^u))$, then the optimal weights are calculated as follows:

$$\mathbf{w}^* = \left(-\frac{g'(f_1^u)}{1 - g'(f_1^u)}, \frac{1}{1 - g'(f_1^u)} \right). \quad (1)$$

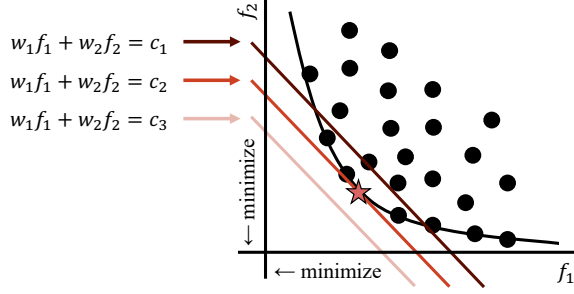


Fig. 3. Conceptual visualization of Proposition 1. The black dots represent each evaluated result, whereas the black line represents the approximated Pareto front $f_2 = g(f_1)$. The three different red lines represent the level sets, each with a different single weighted metric value. As shown by the red star, if the approximated Pareto front is strictly convex, the minimum possible c is achieved when the level set is tangent to the approximated Pareto front. In our example, this is when we take c_2 .

Proof. Let $L_w(c)$ be the level set $\{(f_1, f_2) | w_1 f_1 + w_2 f_2 = c\}$ and assume that g' is the derivative of g . Since $w_1 f_1 + w_2 f_2 = c$ can be transformed into $f_2 = h(f_1) = -\frac{w_1}{w_2} f_1 + \frac{c}{w_2}$ and $g(f_1) - h(f_1)$ is also a strict convex function, the minimum c^* , which achieves $L_w(c) \neq \emptyset$, is obtained when $g'(f_1) - h'(f_1) = 0$, i.e., $g'(f_1) = -\frac{w_1}{w_2}$, and $\min_{f_1} g(f_1) - h(f_1) = 0$ hold. Owing to strict convexity, $g(f_1) - h(f_1)$ becomes minimum at the point where $g'(f_1) - h'(f_1) = 0$ holds. As the minimum must be at $(f_1^u, g(f_1^u))$, $g'(f_1^u) = -\frac{w_1^*}{w_2^*}$ must hold. By solving $g'(f_1^u) = -\frac{w_1^*}{w_2^*}$ and $w_1 + w_2 = 1$, we obtain $\mathbf{w}^* = (-\frac{g'(f_1^u)}{1 - g'(f_1^u)}, \frac{1}{1 - g'(f_1^u)})$. Note that Pareto optimality in 2D guarantees that the Pareto front $g(f_1)$ decreases monotonically, resulting in $g'(f_1) \leq 0 \Rightarrow g'(f_1) - 1 \neq 0$. Since the optimality at the user-specified point, i.e., $L_{\mathbf{w}^*}(w_1^* f_1^u + w_2^* f_2^u) = \{(f_1^u, f_2^u)\}$, holds, this completes the proof. $\square$

An intuitive proof of this is provided in Figure 3. The figure shows three level sets $L_w(c_1)$, which has two intersections with $f_2 \geq g(f_1)$, $L_w(c_2)$, which is tangent to $f_2 \geq g(f_1)$, and $L_w(c_3)$, and does not have any intersection with $f_2 \geq g(f_1)$ where $c_1 < c_2 < c_3$ as can be derived trivially from the hierarchical relationship of the intercepts. The proof states that owing to strict convexity of g , there exists a level set that is tangent to $f_2 \geq g(f_1)$ and contains a unique point.

In this study, we use $g(f_1) = f_2 = \frac{b}{f_1 - a} + c$, where $a > f_1^{\min}$ and $b > 0$, for the approximation and fit the parameters a, b, c by minimizing the squared error.

B. Preference Optimal Weight Calculation for Multi-Metric Problems

Given a user-specified point, the optimal weights can be calculated in a manner similar to that in Proposition 1. Let $g(\mathbf{f}) = b$ be the approximated Pareto front defined on

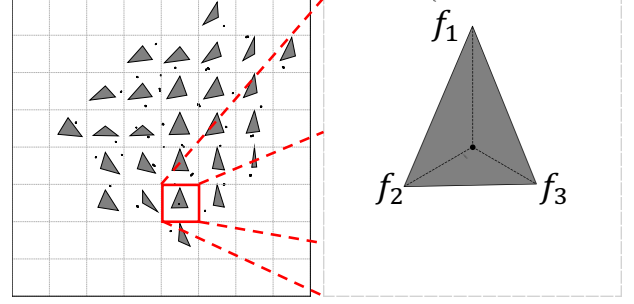


Fig. 4. Visual support to select a multi-metric trade-off. **Left:** Pareto solutions in the space reduced by UMAP. Pareto solutions in each lattice are summarized by a radar chart. **Right:** Radar chart that shows the mean Pareto solutions in each lattice. Each radar chart shows how good each metric is. The larger a radar chart is, the better the Pareto solutions in the lattice are.

$[f_1^{\min}, f_1^{\max}] \times \dots \times [f_M^{\min}, f_M^{\max}]$ and $\mathbf{f}^u \in g^{-1}(b)$ be the user-specified point. Thus, the following holds:

Theorem 1. The optimal weights are calculated as follows:

$$\mathbf{w}^* \propto \nabla g(\mathbf{f}^u) := \left(\frac{\partial g(\mathbf{f}^u)}{\partial f_1}, \dots, \frac{\partial g(\mathbf{f}^u)}{\partial f_M} \right). \quad (2)$$

Proof. Similar to Proposition 1, if $\mathbf{f}^u$ is on the approximated Pareto front, which is guaranteed from $\mathbf{f}^u \in g^{-1}(b)$, the tangential hyperplane at $\mathbf{f}^u$, i.e., $H := \{\mathbf{f} | \nabla g(\mathbf{f}^u)^\top \cdot (\mathbf{f} - \mathbf{f}^u) = 0\}$, satisfies $H \cap \{\mathbf{f} \in g^{-1}(b)\} = \{\mathbf{f}^u\}$ owing to strict convexity. This completes the proof. Notably, the optimal weights are calculated by normalizing each element of $\nabla g(\mathbf{f}^u)$ with its summation. $\square$

In our experiment, we use $g(\mathbf{f}) := \prod_{m=1}^M (f_m - a_m) = b$, which is the general version of $g(f_1) = f_2 = b/(f_1 - a) + c$, leading to the following weights:

$$\mathbf{w}^* \propto \left(\frac{b}{f_1^u - a_1}, \dots, \frac{b}{f_M^u - a_M} \right), \quad (3)$$

where we used the gradients of the approximated Pareto front at $\mathbf{f}^u$ calculated as follows:

$$\frac{\partial g(\mathbf{f}^u)}{\partial f_i} = \prod_{m \neq i} (f_m^u - a_m) = \frac{b}{f_i^u - a_i}. \quad (4)$$

Thus, we can color PCPs based on a single weighted metric, as in bi-metric problems. Recall that although we chose $g(\mathbf{f}) = \prod_{m=1}^M (f_m - a_m) = b$ as the approximated Pareto front, the choice of the function can be arbitrary as long as the function is strictly convex and differentiable in the domain of $\mathbf{f}$. However, it is difficult for users to select $\mathbf{f}^u$, unlike in bi-metric problems. To this end, we propose visual support that enables users to intuitively select a trade-off of interest.

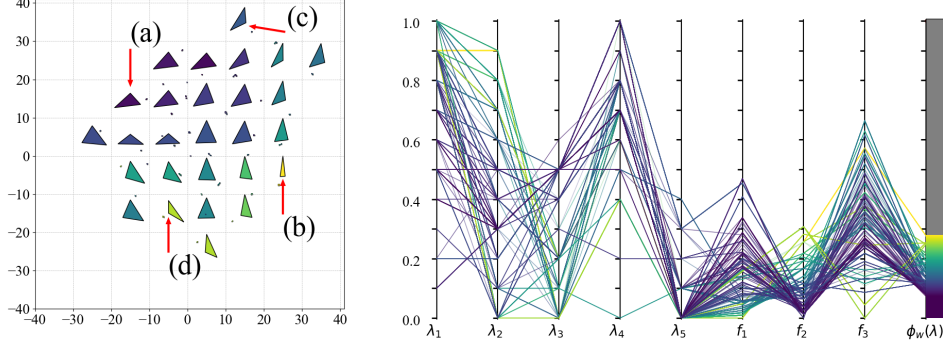


Fig. 5. Illustration of the proposed method. **Left:** Radar charts of mean Pareto solutions on the 2D plane reduced by UMAP. Each radar chart shows the mean Pareto solution in the corresponding lattice, which helps users select their preference intuitively. UMAP enables radar charts with similar patterns to be located close to each other. The optimal weights are calculated based on the selected radar chart. **Right:** PCPs colored based on Radar Chart (a). Blue represents better single weighted metric values, whereas yellow represents worse single weighted metric values.

C. Visual Support to Select Multi-Metric Trade-off

An overview of our proposition is shown in Figure 4. To provide visual support, we first reduce the Pareto solutions $\mathcal{P}$ to a 2D plane, which is divided into lattices at even intervals, using UMAP [10]^{*}. In particular, radar charts with similar patterns are located close to each other, making it easier for users to select a radar chart. We then calculate the mean Pareto solution for each lattice. Finally, we plot a radar chart for each lattice using the following:

$$\bar{f}'_m := 1 - \frac{\bar{f}_m - f_m^{\min}}{f_m^{\max} - f_m^{\min}}. \quad (5)$$

Note that $\bar{f}_m$ is the m -th metric value of the mean Pareto solutions in a lattice, and $f_m^{\min}, f_m^{\max}$ are the minimum and maximum m -th metric values of the Pareto solutions, respectively. The optimal weights are calculated based on the user's choice of a radar chart from the 2D plane, as shown in Figure 4. The pseudocode for the proposed visual support is described in Algorithm 1.

After a radar chart is specified, we first calculate the nearest Pareto solution $\mathbf{f}^u$ on the approximated Pareto front $g(\mathbf{f}) := \prod_{m=1}^M (f_m - a_m) = b$ using the sequential least squares programming in `scipy.minimize`[†]. We assume $\mathbf{f}^r$ is the mean Pareto solution specified by the radar chart. We then find the nearest Pareto solution by solving the following:

$$\mathbf{f}^u \in \arg \min_{\mathbf{f} \in g^{-1}(b)} \|\mathbf{f}^r - \mathbf{f}\|_2. \quad (6)$$

The formulation above is equivalent to $\arg \min_{\mathbf{f} \in \mathbb{R}^M} \|\mathbf{f}^r - \mathbf{f}\|_2$ subject to $g(\mathbf{f}) = b$. Once the nearest Pareto solution is identified, the optimal weights can be computed easily using Eq. (2). Using the optimal weights, we color PCPs as in Figure 5.

^{*}<https://github.com/lmcinnes/umap>

[†]<https://docs.scipy.org/doc/scipy/reference/optimize.minimize-slsqp.html>

Algorithm 1 PCPs for Multi-Metric Problems

Input: evaluated results $\mathcal{D} := \{(\boldsymbol{\lambda}^{(n)}, \mathbf{f}^{(n)})\}_{n=1}^N$

1: (1) Initialization

- 2: Extract the Pareto solutions $\mathcal{P}$ from $\mathcal{D}$
- 3: Approximate the Pareto front by fitting the parameters $\{a_m\}_{m=1}^M, b$ in $g(\mathbf{f}) = \prod_{m=1}^M (f_m - a_m) = b$
- 4: Apply UMAP to $\mathcal{P}$
- 5: **for** each lattice **do**
- 6: **if** No solutions in the lattice **then**
- 7: **continue**
- 8: Collect the solutions in the lattice.
- 9: Calculate the mean Pareto solution $\bar{\mathbf{f}}$
- 10: Transform $\bar{\mathbf{f}}$ into $\bar{\mathbf{f}}'$ using Eq. (5)
- 11: Plot a radar chart in the lattice

12: (2) PCPs with the optimal weights

- 13: Select a radar chart
 - 14: Solve Eq. (6) to find the nearest Pareto solution $\mathbf{f}^u$
 - 15: Calculate the optimal weights $\mathbf{w}^*$ based on Eq. (2)
 - 16: Color PCPs based on the optimal weights $\mathbf{w}^*$
-

IV. REAL-WORLD APPLICATION TO PEDESTRIAN FLOW SIMULATION

A. Problem Setup

In this section, we provide an application example using CrowdWalk [21][‡], a pedestrian flow simulator in evacuation guidance. The pedestrian flow in the simulator is controlled by five control parameters: λ_1 , the time interval of the start of evacuation, λ_2 and λ_3 , the percentages of guidance from the east side to the west side at two different locations, and λ_4 and λ_5 , the percentages of guidance to the underground at two different locations, respectively.

[‡]<https://github.com/crest-cassia/CrowdWalk>

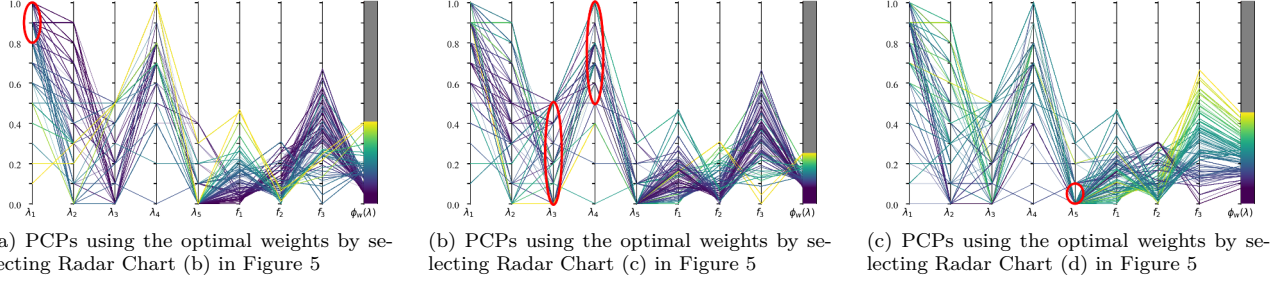


Fig. 6. PCPs based on three different radar charts. The PCPs are colored so that blue represents better single weighted metric values and yellow represents worse single weighted metric values. **Left:** The weights that put emphasis on f_1 . A larger λ_1 improves f_1 as can be seen in the red circle. **Center:** The weights that put emphasis on f_2 . A smaller λ_3 or a larger λ_4 improves f_2 as can be seen in the red circles. **Right:** The weights that put emphasis on f_3 . A smaller λ_5 improves f_3 as can be seen in the red circle.

Each control parameter was normalized to $[0, 1]$ and we discretized the domain of each control parameter into $\{k/10\}_{k=0}^{10}$, leading to 11^5 control parameter combinations overall. We consider three metrics in this problem: f_1 , the degree of congestion, f_2 , the overall evacuation completion time, and f_3 , the total traveling distance of pedestrians, where all the metrics were normalized to $[0, 1]$ for convenience. Note that PCPs visualize the Pareto solutions and the top-30 observations with respect to the single weighted metric $\phi_w(\lambda)$, and we sampled 1000 control parameter vectors using `TPESampler`[§] in Optuna [1] and performed the simulator with these control parameters to collect the results. We defer the details of the TPE algorithm to Watanabe [17] and those of the multi-objective version to Ozaki *et al.* [11], [12].

B. Visualization Results

Figure 6 visualizes the PCPs colored based on different metric trade-offs, which are indicated by (b), (c), and (d) in Figure 5. Recall that a concentration of PCP lines with a similar color on a control parameter axis indicates that the control parameter is the key factor to achieve the single weighted metric values close to those specified by the color [20].

The PCPs in Figure 6a put more emphasis on congestion, f_1 , by selecting Radar Chart (b) in Figure 5. As marked by the red circle, high values in λ_1 attract purple lines. It implies that a longer time interval is important at the start of evacuation. By doing so, sufficient capacity will be preserved in the flow line, reducing congestion. The PCPs in Figure 6b focus more on the overall evacuation completion time, f_2 , by selecting Radar Chart (c) in Figure 5. As shown by the red circles, purple lines concentrate at low values in λ_3 and high values in λ_4 . It indicates that east side guidance in λ_3 and underground guidance in λ_4 are essential to reduce the overall evacuation completion time. Finally, we consider the single weighted metric with

a higher weight on the total traveling distance of pedestrians, f_3 , by selecting Radar Chart (d) in Figure 5. As can be seen in the PCPs in Figure 6c, low λ_5 is the key factor of this scenario. It means that the underground route should be avoided as much as possible at the location controlled by λ_5 if the total traveling distance of pedestrians is the primary objective.

V. RELATED WORK & PRACTICAL CONSIDERATION

The interpretation methods for the relationship between the control parameters and metrics can be roughly divided into two main categories: (1) Importance quantification of each control parameter, and (2) visualization of high-performance control parameter distribution.

Importance quantification is typically used to reduce the complexity of the analysis via the control parameter selection. For example, f-ANOVA [6], PED-ANOVA [20], and SHAP [9] quantify the contribution of each control parameter to the variations of the metric of interest. Since these methods are intended for single-metric problems, they cannot be immediately applied to multi-metric problems. To the best of our knowledge, few methods address importance quantification for multi-metric problems. Theodorakopoulos *et al.* [15] extended f-ANOVA to multi-metric setups. Although they also focused on the linear weighting of each metric, cf. Section III, their focus was to compute the importance of each control parameter across different weight pairs, requiring users to select the desirable weights manually. Meanwhile, we focused on deriving the optimal weights based on the user feedback. The disadvantage of importance quantification is information loss regarding the locations of promising control parameters. More specifically, this approach compresses the distribution information of control parameters and metric values into numerical values, making it difficult to perform a more accurate analysis. This motivates the use of PCPs to interpret the patterns in high-performance control parameters more precisely.

Another approach involves visualizing the distribution of promising control parameters. In practice, scatter plot

[§]<https://optuna.readthedocs.io/en/stable/reference/samplers/generated/optuna.samplers.TPESampler.html>

matrices [4] are widely used in this category. However, as the number of figures increases quadratically, PCPs, which allow us to view high-dimensional data in a figure, become more advantageous than scatter plot matrices for high-dimensional problems. Another example is landscape visualization using dimension reduction to a two-dimensional (2D) plane and a predictive model [8], [13]. PCPs, including variants such as density-based parallel coordinates [5], are also classified into this category. PCPs are frequently used in multi-metric problems [2], [7]. However, these studies did not discuss how to color each polyline to enhance interpretability regarding the trade-off between metrics. The goal of this study is to provide a mathematical formulation to color PCPs based on user preferences and visual support to allow users to intuitively provide their preferences.

Practically speaking, these interpretation methods assist in reducing experiment design or black-box optimization problems to easier ones. The concrete examples are control parameter selection by checking the importance and making some metrics into constraints. The effectiveness of the former approach is reported by Watanabe *et al.* [18], [19], who used PED-ANOVA for this, and Florea and Andonie [3], who used f-ANOVA for this. The latter approach is particularly essential for many-objective problems ($M \geq 4$) because most evaluations achieve Pareto optimality in such a case owing to the curse of dimensionality. Although this problem is tackled independently in the context of ϵ constrained-based approach [14], our method is another effective approach to make some metrics into inequality constraints, as we can visually define appropriate thresholds for each metric in PCPs colored by our method. By relaxing the problems, finding desirable trade-offs becomes viable in an efficient manner.

VI. CONCLUSION

This paper proposes a method to color PCPs for multi-metric problems given user feedback. Our method facilitates the user's selection of metric preferences by displaying the Pareto solutions in the 2D plane reduced by UMAP and metric radar charts at each lattice. By coloring PCPs based on the user preference, a more intuitive interpretation of promising control parameters for the specified metric trade-off becomes possible. To verify this, we conducted a visualization and analysis using simulation results of evacuation guidance. In the analysis, we considered three different trade-offs. Interestingly, different control parameters were important for each scenario. Although there have not been any methods to color PCPs effectively in such a case, we could successfully address this problem and demonstrate the effectiveness through our application. As discussed in Section V, our approach has the potential to contribute to relaxing many-objective problems into fewer-metric problems with some inequality constraints. Such a study is a possible future research topic.

REFERENCES

- [1] T. Akiba, S. Sano, T. Yanase, T. Ohta, and M. Koyama. Optuna: A next-generation hyperparameter optimization framework. In *International Conference on Knowledge Discovery & Data Mining*, 2019.
- [2] P.J. Fleming, R.C. Purshouse, and R.J. Lygoe. Many-objective optimization: An engineering design perspective. In *International conference on evolutionary multi-criterion optimization*, 2005.
- [3] A.C. Florea and R. Andonie. Weighted random search for hyperparameter optimization. *arXiv:2004.01628*, 2020.
- [4] G. Grinstein, M. Trutschl, and U. Cvek. High-dimensional visualizations. In *Visual Data Mining Workshop at KDD*, 2001.
- [5] J. Heinrich and D. Weiskopf. State of the art of parallel coordinates. *Eurographics*, 2013.
- [6] F. Hutter, H. Hoos, and K. Leyton-Brown. An efficient approach for assessing hyperparameter importance. In *International conference on machine learning*, 2014.
- [7] M. Li, L. Zhen, and X. Yao. How to read many-objective solution sets in parallel coordinates. *Computational Intelligence Magazine*, 12, 2017.
- [8] M. Lindauer, M. Feurer, K. Eggensperger, A. Biedenkapp, and F. Hutter. Towards assessing the impact of Bayesian optimization's own hyperparameters. *arXiv:1908.06674*, 2019.
- [9] S.M. Lundberg and S.I. Lee. A unified approach to interpreting model predictions. *Advances in Neural Information Processing Systems*, 2017.
- [10] L. McInnes, J. Healy, and J. Melville. UMAP: Uniform manifold approximation and projection for dimension reduction. *arXiv:1802.03426*, 2018.
- [11] Y. Ozaki, Y. Tanigaki, S. Watanabe, M. Nomura, and M. Onishi. Multiobjective tree-structured Parzen estimator. *Journal of Artificial Intelligence Research*, 73, 2022.
- [12] Y. Ozaki, Y. Tanigaki, S. Watanabe, and M. Onishi. Multi-objective tree-structured Parzen estimator for computationally expensive optimization problems. In *Genetic and Evolutionary Computation Conference*, 2020.
- [13] R. Sass, E. Bergman, A. Biedenkapp, F. Hutter, and M. Lindauer. DeepCAVE: An interactive analysis tool for automated machine learning. *arXiv:2206.03493*, 2022.
- [14] S. Song, K. Zhang, L. Zhang, and N. Wu. A dual-population algorithm based on self-adaptive epsilon method for constrained multi-objective optimization. *Information Sciences*, 655, 2024.
- [15] D. Theodorakopoulos, F. Stahl, and M. Lindauer. Hyperparameter importance analysis for multi-objective AutoML. *arXiv:2405.07640*, 2024.
- [16] S. Watanabe. Python tool for visualizing variability of Pareto fronts over multiple runs. *arXiv:2305.08852*, 2023.
- [17] S. Watanabe. Tree-structured Parzen estimator: Understanding its algorithm components and their roles for better empirical performance. *arXiv:2304.11127*, 2023.
- [18] S. Watanabe, N. Awad, M. Onishi, and F. Hutter. Multi-objective tree-structured Parzen estimator meets meta-learning. *Meta-learning Workshop at Advances in Neural Information Processing Systems*, 2022.
- [19] S. Watanabe, N. Awad, M. Onishi, and F. Hutter. Speeding up multi-objective hyperparameter optimization by task similarity-based meta-learning for the tree-structured Parzen estimator. *arXiv:2212.06751*, 2023.
- [20] S. Watanabe, A. Bansal, and F. Hutter. PED-ANOVA: Efficiently quantifying hyperparameter importance in arbitrary subspaces. *arXiv:2304.10255*, 2023.
- [21] T. Yamashita, T. Okada, and I. Noda. Implementation of simulation environment for exhaustive analysis of huge-scale pedestrian flow. *Journal of Control, Measurement, and System Integration*, 6, 2013.