An Interactive Multi-Objective Optimization Method for Deriving Measures to Social Issues while Presenting the Predicted Pareto Frontier

Toshio Ito
Converging Technologies Laboratory
Fujitsu Limited
Kawasaki, Japan
ito.toshio@fujitsu.com

Kota Itakura
Converging Technologies Laboratory
Fujitsu Limited
Kawasaki, Japan
itakura.kota@fujitsu.com

Arika Hakoda
Converging Technologies Laboratory
Fujitsu Limited
Kawasaki, Japan
h.arika@fujitsu.com

Miwa Ueki
Converging Technologies Laboratory
Fujitsu Limited
Kawasaki, Japan
ueki.miwa@fujitsu.com

Koki Ikeda
Department of Computer Science
Institute of Science Tokyo
Yokohama, Japan
ikeda.k@ic.c.titech.ac.jp

Kota Nagakane
Department of Computer Science
Institute of Science Tokyo
Yokohama, Japan
nagakane.k@ic.c.titech.ac.jp

Isao Ono
Department of Computer Science
Institute of Science Tokyo
Yokohama, Japan
isao@c.titech.ac.jp

***Abstract*—Social issues are complex, and solving them from a single perspective has other negative effects. We set up various Key Performance Indicators (KPIs), in which balanced measures need to be implemented to achieve KPIs. However, it is not easy for decision-makers to find measures for issues while considering multiple perspectives. So, we develop an interactive measure search method that simulates the impact of measures on society through various KPIs, interacts with decision-makers, and evaluates numerous measures to identify those aligned with the decision-makers' values. This method become a type of interactive multi-objective optimization method using evolutionary computation. To find the most preferred measure for decision-makers, this method narrows the search area to the decision-makers' preference region within the Pareto Frontier (PF) while interacting with decision-makers. However, in the early stages of interaction, since only a few optimal solutions calculated by evolutionary computation are available, the overall shape of the PF is not well understood, making it difficult for decision-makers to specify their preference region within the PF. Therefore, we propose a method where we predict the PF from the optimal solutions computed by evolutionary computation, present the predicted PF to decision-makers. The proposed method reduces the number of interactions with decision-makers, enabling faster determination of preferred measures for them.**

***Index Terms*—interactive multi-objective optimization, social issues, optimum measures, predicted pareto frontier.**

I. Introduction

Social issues are complex, and solving them from a single perspective has other negative effects. We set up various Key Performance Indicators (KPIs), in which balanced measures need to be implemented to achieve KPIs. However, it is not easy for decision-makers to find measures for issues while considering multiple perspectives. So, we are developing interactive measure search method that simulates the impact of measures on society through various KPIs, interacts with decision-makers, and evaluates numerous measures to identify those aligned with the decision-makers' values. We are also developing a system equipped with our method, and think this method will become a type of interactive multi-objective optimization (IMO) method that finds the most preferred measure for decision-makers.

IMO has been developing in two fields, the multiple criteria decision making (MCDM) (cf. [1]) and the evolutionary multiobjective optimization (EMO) (cf. [2]), which is one of the evolutionary computation. Also, MCDM and EMO have been particularly interested in IMO recently. For MCDM, IMO uses the mathematical programming methodology to find a single preferred Pareto optimal solution. On the other hand, for EMO, IMO uses evolutionary algorithms to obtain solutions for better perspective of decision makers. Please refer to [3] on what IMO methods exist for both MCDM and EMO and how IMO has developed. Also, please refer to [4] for how each method has been evaluated.

Since our method uses evolutionary computation, it can be considered a type of EMO. Our method narrows the search area of evolutionary computation to decision-makers' preference region within a Pareto Frontier (PF) while interacting with decision-makers.

There are many methods [5]–[13] in EMO that use reference points or modified versions of them to establish the target value that decision-makers consider ideal. Also, Decision Support for computationally Demanding Optimization (DESDEO), a modular open-source framework incorporating interactive multi-objective optimization for the first time, was recently proposed in [14], and supports reference points.

However, decision-makers does not always know the ideal target value from the outset; there are times when decision-makers wish to set a reference point only after reviewing several optimal solutions in the PF. Even if decision-makers know the ideal target value from the outset, if there is no optimal solution nearby, repeating the interaction will not yield the desired optimal solution, thereby placing a psychological burden on decision-makers.

If the entire PF can be presented to decision-makers from the outset, they can use it as a reference to correctly set the ideal target value. Furthermore, decision-makers can realize that there is no optimal solution close to the ideal target value, eliminating the need to repeat unnecessary interactions.

Therefore, our contributions are as follows: We propose a method where we predict the PF from the optimal solutions computed by evolutionary computation, present the predicted PF to decision-makers. Unlike methods that cannot present the predicted PF, the proposed method can reduce the number of interactions with decision-makers by presenting the overall shape of the PF to them using the predicted PF. Therefore, decision-makers can more quickly determine the measure which they prefer. Also, the method for presenting the predicted PF is not found in [5]–[14].

The structure of this paper is as follows: This section is the introduction and, in section II, we show the flow of our proposed method. In section III, we describe how to generate and update the predicted PF. In section IV, we show the use case selected as a social issue and the performance evaluation for the proposed method. Also, section IV has subsections as follows: In subsection IV-A, we explain the use case and simulator we select as a social issue. In subsection IV-B, we explain the evolutionary computation we use. In subsection IV-C, we conduct experiments on the number of searches to evaluate the performance of the proposed method. In subsection IV-D, we describe the experiment by humans. Finally, in section V, we conclude our paper.

II. The Flow of Our Proposed Method

In this section, we explain the flow of our proposed method, as shown in Fig. 1.

We take up the following case as an example to explain the flow. We set up several e-stations for shared e-scooters on an island and encourage the use of e-scooters instead of cars. Consequently, we examine the effect on the amount of CO_2, convenience, and profit. As a measure, we consider the initial number of e-scooters installed at each e-station in an early morning and determine the optimal number of installations.

In step 1, several measure points in the measure parameter space are determined for the next simulation evaluation. For example, if there are two e-stations, the measure parameter

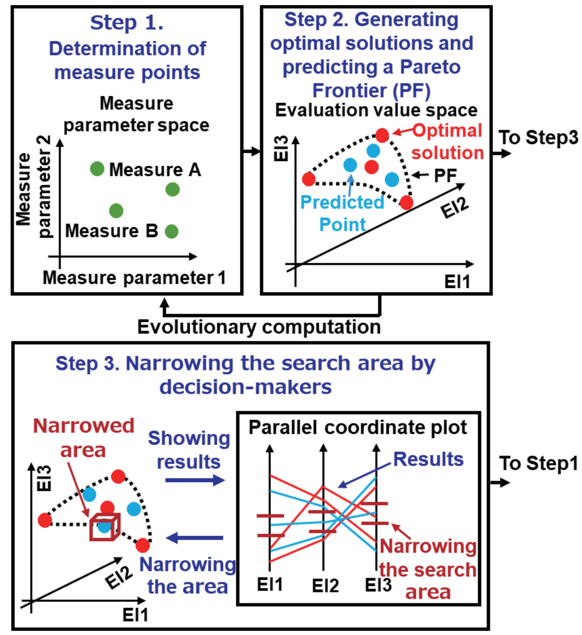

Fig. 1. The flow of our proposed method.

space becomes a two-dimensional space and each measure parameter is the initial number of e-scooters installed at each e-station.

In step 2, the simulation for each measure point is executed and the evaluation values are calculated. As evaluation indexes (EIs), which are the same as KPIs, the total amount of CO_2, the total time taken for travel and the profit of the e-scooter company are set, and their evaluation values are calculated. The evolutionary computation then determines a number of measure points for the next simulation evaluation. The simulations for these measure points are run to calculate the evaluation values, which are repeated to calculating several optimal solutions in the evaluation value space. After the optimal solutions are obtained, the proposed method uses linear interpolation to predict a PF and determine predicted points on the predicted PF.

In step 3, the proposed method presents the optimal solutions and predicted points obtained in step 2 on a parallel coordinate plot (cf. [15]). Each vertical axis of the parallel coordinate plot represents each evaluation index. Each red or blue line segment represents an optimal solution or a predicted point. Decision-makers view the results presented on the parallel coordinate plot and use slider bars on each vertical axis to narrow down their preference region. For example, if decision-makers prioritize the profit, they can use sliders on the axis of profit to narrow down to the higher-value range. The proposed method selects prediction points within the narrowed-down region and returns to step 1.

In step 1, it uses evolutionary computation to find the optimal solution corresponding to each selected prediction point. In step 2, the newly obtained optimal solutions are used to update the predicted PF. In step 3, the obtained optimal

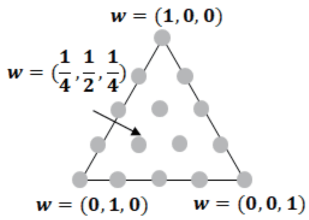

Fig. 2. A weight space and interval sampling.

solutions and the prediction points of the updated predicted PF are presented on the parallel coordinate plot. Decision makers view the results presented on the parallel coordinate plot and use slider bars on each vertical axis to narrow down their preference region. Alternatively, decision-makers determine which measures to adopt from among the optimal solutions. For example, if a more profitable measure is found, decision-makers choose it.

III. The Method to Generate and Update the Predicted PF

In this section, we describe how to generate and update the predicted PF.

Let X be an m-dimensional measure parameter space and $x = (x_1, x_2, \cdots, x_m)$ be a point on X. Let F be an n-dimensional evaluation value space, and $f(x) = (f_1(x), f_2(x), \cdots, f_n(x)) \in F$ represents the evaluation result after executing a simulation. For the value of each evaluation index, we transform the value such that larger values are better. For $f_i(x)(i = 1, \cdots, n)$, if a smaller value of $f_i(x)$ is better, we transform it to $g_i(x) = -f_i(x)$. If a larger value of $f_i(x)$ is better, we set $g_i(x) = f_i(x)$. Next, we normalize $g_i(x)$ within the range $[-1, 1]$ and denote the normalized result as $\tilde{g}_i(x)$.

Let W be an n-dimensional weight space which becomes

$$\{w = (w_1, \cdots, w_n) | 0 \leq w_i \leq 1, \sum_{i=1}^{n} w_i = 1\}.$$

Then, we define the weighted sum $\tilde{g}_w(x) = \sum_{i=1}^{n} w_i \tilde{g}_i(x)$.

We use an evolutionary computation with a single objective function. When the value of the measure parameter space for the optimal solution of $\tilde{g}_w(x)$ is denoted by x_w, the optimal solution is represented by $\hat{g}_w = \tilde{g}_w(x_w)$.

We generate and update the predicted PF as follows. By interval or random sampling, we obtain p weight samples $w^j(j = 1, \cdots, p)$ as shown in Fig. 2. Fig. 3 shows the method for genarating and updating a predicted PF, and Fig. 4 shows the flow of the method. In step 1 of Fig. 3, by using evolutionary computation, we generate the optimal solutions $\hat{g}_{(1,0,\cdots,0)}, \hat{g}_{(0,1,\cdots,0)}, \cdots, \hat{g}_{(0,0,\cdots,1)}$ corresponding to the vertices of the PF. Next, by using the optimal solutions, we generate the predicted points on the predicted PF using linear interpolation as

$$g'_{w^j} = w_1^j \hat{g}_{(1,0,\cdots,0)} + w_2^j \hat{g}_{(0,1,\cdots,0)} + \cdots + w_n^j \hat{g}_{(0,0,\cdots,1)},$$

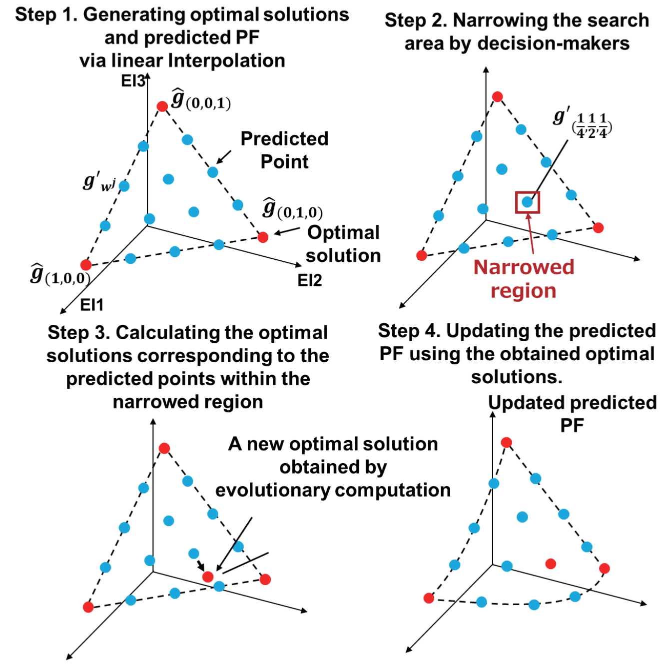

Fig. 3. The method for genarating and updating a predicted PF.

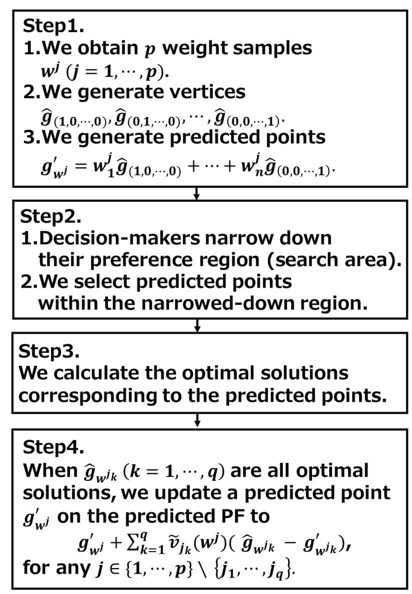

Fig. 4. The flow of the method.

where $j = 1, \cdots, p$. Also, we define the value of the measure parameter space for w^j as

$$x'_{w^j} = w_1^j x_{(1,0,\cdots,0)} + w_2^j x_{(0,1,\cdots,0)} + \cdots + w_n^j x_{(0,0,\cdots,1)}.$$

x'_{w^j} is used as the initial value for evolutionary computation when finding the optimal solution for $\tilde{g}_{w^j}(x)$. In step 2, after decision-makers narrow down their preference region, we select predicted points within the narrowed-down region. In step 3, we calculate the optimal solutions corresponding to the predicted points. In step 4, we update the predicted points

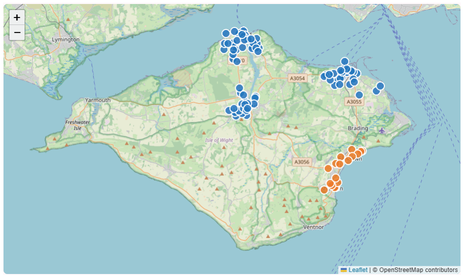

Fig. 5. The installation of e-stations on an island in Europe.

on the predicted PF by using the optimal solutions as follows: Let r or ϵ be a large or small positive real number. Let $\hat{g}_{w^{j_k}}$ $(k = 1, \cdots, q)$ be all optimal solutions already obtained. For any $j \in \{1, \cdots, p\} \setminus \{j_1, \cdots, j_q\}$, we define the Euclidean distance between w^j and w^{j_k} as $d_{j_k}(w^j) = \|w^j - w^{j_k}\|$. When we put

$$v_{j_k}(w^j) = (d_{j_k}(w^j) + \epsilon)^{-r}, \tilde{v}_{j_k}(w^j) = \frac{v_{j_k}(w^j)}{\Sigma_{l=1}^{q} v_{j_l}(w^j)}, \quad (1)$$

we update g'_{w^j} to

$$g'_{w^j} + \Sigma_{k=1}^{q} \tilde{v}_{j_k}(w^j)(\hat{g}_{w^{j_k}} - g'_{w^{j_k}}), \quad (2)$$

using the inverse distance weighting method, and denote this updated value as g'_{w^j}.

IV. The use case selected as a social issue and the performance evaluation for the proposed method

A. The use case and simulator selected as a social issue

In this subsection, we explain the use case and simulator we select as a social issue.

We take up the use case of shared e-scooters and set up several e-stations on an island. We consider the measure about the optimal initial number of e-scooters installed at each e-station in an early morning to change the usage dynamics of e-scooters. The number of e-stations is set to 100, the e-stations are installed as shown in Fig. 5, and the nodes in Fig. 5 represent e-stations. The number of e-scooters at each e-station is set between 0 and 5. Therefore, X is the 100-dimensional measure parameter space and $x = (x_1, x_2, \cdots, x_{100}) \in X$ satisfies $x_i = 0, 1, \cdots, 5$ $(i = 1, \cdots, 100)$.

We use the traffic simulator of [16] in the use case of shared e-scooters. This traffic simulator utilizes data of origin-destination (OD) to determine user travel demand according to human behavioural modelling [17], [18]. It simulates travel modes including car travel, walking, and e-scooter rental/return. The UI of this traffic simulator is shown in Fig. 6.

The simulation time is set from 6:00 to 7:00, and the number of OD is set to 25380. Therefore, from 6:00 to 7:00, 25380 people move. As evaluation indexes, we set the following.

- f_1: The total amount of CO_2
- f_2: The total time taken for all travels
- f_3: The e-scooter usage rate

Fig. 6. The UI of the traffic simulator for shared e-scooters.

- f_4: The revenue of an e-scooter company
- f_5: The opportunity loss in e-scooter usage
- f_6: The ratio of available e-stations
- f_7: The total number of e-scooters

So, F is the 7-dimensional evaluation value space, and $f(x) = (f_1(x), f_2(x), \cdots, f_7(x)) \in F$ represents the value of the above evaluation indexes.

We use the agent-based simulation library SOARS Toolkit [19], [20] for this traffic simulator. It is a library based on the role stage model, which is a kind of agent-based model, and the simulation can be easily implemented.

B. The evolutionary computation

In this subsection, we explain the evolutionary computation we use.

We use DX-NES-ICI [21] as the evolutionary computation, which is used in [22]. DX-NES-ICI is a natural evolution strategy for mixed-integer black box optimization with a single objective function.

Next, we describe the values of parameters to be set in DX-NES-ICI. We normalize $x \in X$ within the range $[-1, 1]$. For each evaluation value, we transform the value such that larger values are better as follows:

$$g_i(x) = f_i(x)(i = 3, 4, 6),$$
$$g_i(x) = -f_i(x)(i = 1, 2, 5, 7).$$

We normalize $g_i(x)$ within the range $[-1, 1]$ and denote the normalized result as $\tilde{g}_i(x)$.

Next, by the following interval sampling, we obtain 3003 weight samples w^j $(j = 1, \cdots, 3003)$:

$$\{w = (w_1, \cdots, w_7) \in W | w_i \in \{0, \frac{1}{8}, \frac{1}{4}, \frac{1}{2}, 1\}, i = 1, \cdots 7\}.$$

The initial mean of the multivariate normal distribution of DX-NES-ICI is set as $(0, \cdots, 0)$ for $w^j = (1, \cdots, 0)$, $\cdots$, $(0, \cdots, 1)$ corresponding to the vertices of the PF. The initial mean is set as x'_{w^j} for a weight w^j other than the vertex. We set the initial standard deviation of the multivariate normal distribution of DX-NES-ICI to 0.5, the number of generations to 10, and the population size per generation to 10.

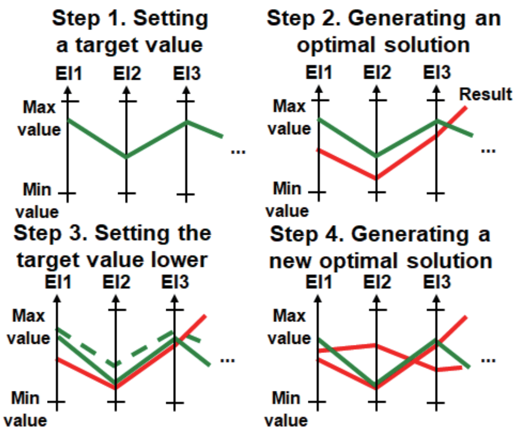

Fig. 7. Searches by the random weight method.

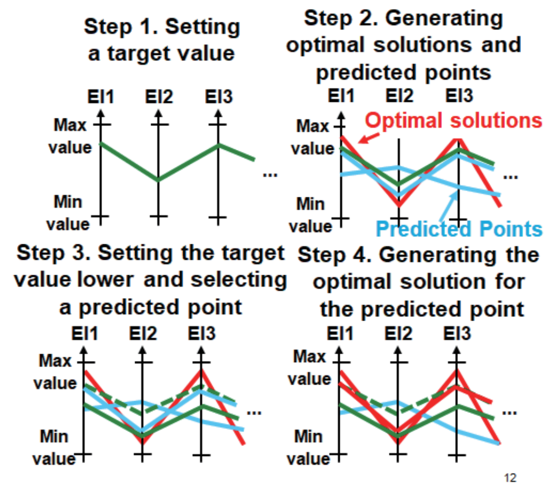

Fig. 8. Searches by the proposed method.

C. The experiment on the number of searches and the performance evaluation

In this subsection, we conduct experiments on the number of searches to evaluate the performance of the proposed method. In particular, we conduct a comparative experiment between the method with the predicted PF and the method without it, and show that the method with it reduces the number of searches.

When the same person repeats the same experiment, the burden increases, and fatigue may prevent accurate evaluation. Therefore, we introduce an evaluation system to replace the judgment by the person. First, for each evaluation index $\tilde{g}_i(x)$ $(i = 1, \cdots, 7)$, the evaluation system sets and fixes a minimum value min_i and a maximum value max_i . The evaluation system randomly set a target value

$$t = (t_1, \cdots, t_7) \in [min_1, max_1] \times \cdots \times [min_7, max_7].$$

If any of the optimal solutions identified through the proposed method or a comparative method exceeds the target value, the evaluation system terminates the search. Otherwise, the evaluation system reset the target value to a lower value as follows:

$$t = (t_1 - 0.01(max_1 - min_1), \cdots, t_7 - 0.01(max_7 - min_7)).$$

And, the evaluation system continues the search using the proposed method or the comparative method. We evaluate how much the proposed method can reduce the number of searches compared to the comparative method.

As the comparative method, we introduce a random weight method that selects weights at random and presents the optimal solutions for those weights. We show the search by the random weight method in Fig, 7. In step 1, the evaluation system randomly set a target value. In step 2, the random weight method selects a weight at random and presents the optimal solution for the weight. If the optimal solution exceeds the target value, the evaluation system terminates the search. Otherwise, in step 3, the evaluation system resets the target value lower. In step 4, the random weight method selects another weight at random and presents the optimal solution for the weight. And, the evaluation system evaluates whether any of the optimal solutions exceeds the target value.

Next, we show the search by the proposed method in Fig, 8. In step 1, the evaluation system randomly set a target value. In step 2, the proposed method generates optimal solutions and predicted points by using the optimal solutions. If any of the optimal solutions exceeds the target value, the evaluation system terminates the search. Otherwise, in step 3, the evaluation system resets the target value lower. And, the evaluation system selects the predicted point closest to the target point. In step 4, the proposed method generates the optimal solution corresponding to the selected predicted point, and updates the predicted points by using the optimal solutions. Then, the evaluation system evaluates whether any of the optimal solutions exceeds the target value.

In this experiment, we randomly generate 100 target values. Each target value is set in the evaluation system, and evaluations are performed for both the proposed method and the random weight method. For each target value, the random weight method is tested 10 times while changing the seed value.

We show the experimental results in Fig. 9 and Fig. 10. Fig. 9 and Fig. 10 are the results by probability values and cumulative probability values, respectively. The horizontal axis represents the number of searches performed, while the vertical axis in Fig. 9 represents the probability of occurrence per 100 trials. Also, the vertical axis in Fig. 10 represents the cumulative probability of occurrence per 100 trials. The red and blue lines are the proposed method and the random weight method, respectively. Each blue line in Fig. 9 represents the average of 10 trials, and the error bar for the blue line represents the standard deviation.

As shown in step 1 of Fig. 3, the 7 optimal solutions corresponding to the vertices of the PF and the predicted points are presented during the first iteration of the proposed method. However, to align the evaluation criteria, we assume the 7

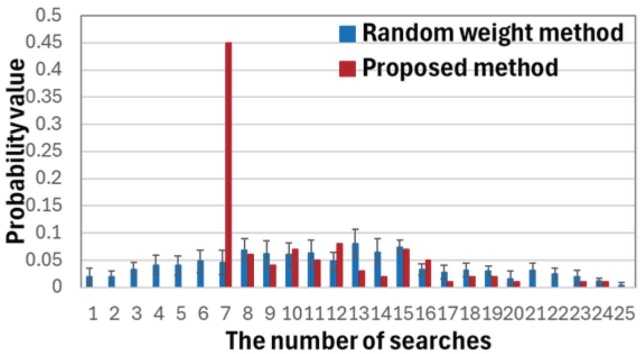

Fig. 9. The result indicated by probability values.

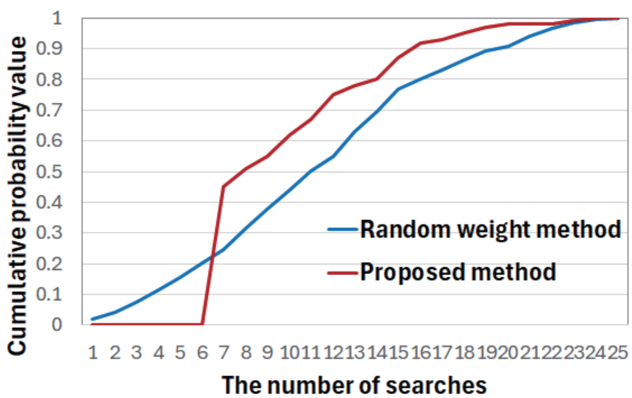

Fig. 10. The result indicated by cumulative probability values.

optimal solutions are found during the seventh search. And, the result for the probability of occurrence is plotted in the seventh search in Figures 9 or 10.

As shown in Fig. 10, we can confirm that after the seventh search, the proposed method reduces the number of searches required to achieve the same cumulative probability value compared to the random weight method. We can also confirm that the reduced number of searches averages 2.7. In practice, the 7 optimal solutions are presented on the first iteration, so we believe we can reduce the number of search by 8.7.

As shown in Fig. 9, the probability of occurrence is extremely high on the seventh search. This result shows that it is important to first present each of the 7 optimal solutions, which is the result of optimizing each evaluation index. Also, it is important to present predicted points. If no predicted point is indicated on the seventh search, decision-makers will be unable to make any choice. And, further search becomes impossible. In this time, we confirm that the proposed method can be executed until the cumulative probability reaches 1 without the search failing at any intermediate stage.

D. The experiment by humans

In this subsection, we describe the experiment by 3 persons.

As shown in step 1 of Fig. 3, the 7 optimal solutions corresponding to the vertices of the PF and the predicted points are presented to each person during the first iteration. The person views the result and terminate the search if the person finds an optimal solution that suits the person's preference. If the person cannot find it, the person selects one predicted point that suits the person's preference. The proposed method then generates the optimal solution corresponding to the prediction point and updates the prediction points as shown in step 4 of Fig. 3. In addition to the previously obtained optimal solutions, the newly obtained optimal solution and updated prediction points are presented to the person. The person views the result and terminate the search if the person finds an optimal solution that suits the person's preference. If the person cannot find it, the person selects one predicted point that suits the person's preference, and the search continues. We evaluate how many iterations it takes to find the optimal solution that suits the person's preferences, and what kind of optimal solution is obtained.

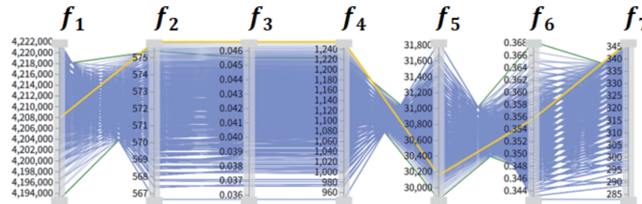

Fig. 11. The optimal solution obtained by the first person.

We assume that each of the persons emphasizes the following: One mainly emphasizes the revenue of an e-scooter company (f_4), the second mainly emphasizes the ratio of available e-stations (f_6) from the perspective for the placement of e-stations, and the third emphasizes balancing the revenue of the e-scooter company (f_4) and the ratio of available e-stations (f_6).

Fig. 11, 12, and 13 show the optimal solutions obtained by the first, second, and third persons, respectively. The yellow line segment represents the obtained optimal solution. The first and second persons can find the optimal solutions during the first iteration. The optimal solution selected for the first person has the highest revenue, and the optimal solution selected for the second person has the highest ratio of available e-stations. The third person can find the optimal solution during the second iteration. After the third person selects a predicted point that balances the revenue and the ratio of available e-stations during the first iteration, the third person obtains the optimal solution during the second iteration. As shown in Fig. 13, the optimal solution is obtained that balances the revenue and the ratio of available e-stations.

Through these experimental results, we confirm that it is possible to select the optimal solution that suits the person's preferences with fewer iterations. By showing the persons the prediction points, they can understand the correlations between evaluation indexes and what values they take. This is important for reducing the number of iterations.

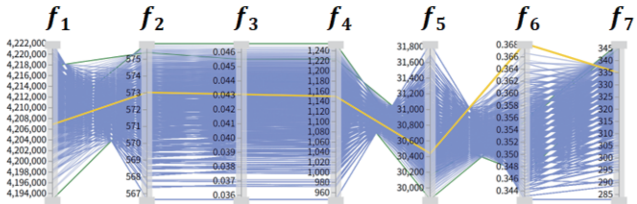

Fig. 12. The optimal solution obtained by the second person.

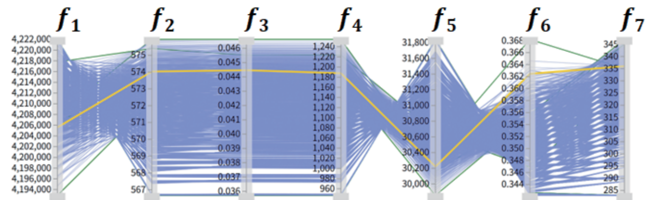

Fig. 13. The optimal solution obtained by the third person.

V. Conclusions

In this paper, we proposed the method to narrow the search area to decision-makers' preference region within the PF while interacting with decision-makers. Especially, we proposed the method where we predict the PF using the optimal solutions and linear interpolation, present the predicted PF to decision-makers, and have them specify the decision-makers' preference region from the predicted PF.

We confirmed that the proposed method reduced the number of searches by the evaluation system. Also, from the experiment by persons, we confirmed that it was possible to select the optimal solution that suits the person's preferences with fewer iterations. However, since the number of persons was 3 this time, we plan to increase the number of persons in the near future.

In the use case of shared e-scooters, we believe the prediction using linear interpolation worked well because the nonlinearity of the PF was weak. We will verify whether predicting the PF using linear interpolation will also work for other use cases. If the nonlinearity of the PF is strong, we believe it is necessary to generate several other optimal solutions in addition to the vertices and then perform linear interpolation.

Also, we are currently developing a system that incorporates the proposed method, and we plan to apply it to mobility and many other use cases. We intend to have actual stakeholders and decision-makers use the system and refine it based on their feedback.

References

[1] C. L. Hwang and K. Yoon, *Multiple attribute decision making: methods and applications: a state-of-the-art survey*. Springer Berlin, Feb. 1981.

[2] K. Deb, *Multi-objective optimization using evolutionary algorithms*. Wiley, Jun. 2001.

[3] B. Xin, L. Chen, J. Chen, H. Ishibuchi, K. Hirota, and B. Liu, "Interactive multiobjective optimization: a review of the state-of-the-art," *IEEE Access*, vol. 6, pp. 41 256–41 279, Jul. 2018.

[4] B. Asfer, K. Miettinen, and F. Ruiz, "Assessing the performance of interactive multiobjective optimization methods: a survey," *ACM Computing Surveys*, vol. 54, no. 4, pp. 1–27, Apr. 2021.

[5] K. Deb and J. Sundar, "Reference point based multi-objective optimization using evolutionary algorithms," *Proceedings of the 8th annual conference on Genetic and evolutionary computation*, pp. 635–642, Jul. 2006.

[6] K. Deb and A. Kumar, "Interactive evolutionary multi-objective optimization and decision-making using reference direction method," *Proceedings of the 9th annual conference on Genetic and evolutionary computation*, pp. 781–788, Jul. 2007.

[7] ——, "Light beam search based multi-objective optimization using evolutionary algorithms," *2007 IEEE congress on evolutionary computation*, pp. 2125–2132, Sep. 2007.

[8] L. Thiele, K. Miettinen, P. J. Korhonen, and J. Molina, "A preference-based evolutionary algorithm for multi-objective optimization," *Evolutionary computation*, vol. 17, no. 3, pp. 411–436, Sep. 2009.

[9] J. Molina, L. V. Santana, A. G. Hernández-Díaz, C. A. C. Coello, and R. Caballero, "g-dominance: Reference point based dominance for multiobjective metaheuristics," *European Journal of Operational Research*, vol. 197, no. 2, pp. 685–692, Sep. 2009.

[10] L. B. Said, S. Bechikh, and K. Ghédira, "The r-dominance: a new dominance relation for interactive evolutionary multicriteria decision making," *IEEE transactions on Evolutionary Computation*, vol. 14, no. 5, pp. 801–818, Apr. 2010.

[11] A. López-Jaimes and C. A. C. Coello, "Including preferences into a multiobjective evolutionary algorithm to deal with many-objective engineering optimization problems," *Information Sciences*, vol. 277, pp. 1–20, Sep. 2014.

[12] A. B. Ruiz, M. Luque, K. Miettinen, and R. Saborido, "An interactive evolutionary multiobjective optimization method: Interactive wasf-ga," *International conference on evolutionary multi-criterion optimization*, pp. 249–263, Jan. 2015.

[13] K. Sindhya, A. B. Ruiz, and K. Miettinen, "A preference based interactive evolutionary algorithm for multi-objective optimization: Pie," *International Conference on Evolutionary Multi-Criterion Optimization*, pp. 212–225, Apr. 2011.

[14] G. Misitano, B. S. Saini, B. Afsar, B. Shavazipour, and K. Miettinen, "Desdeo: The modular and open source framework for interactive multiobjective optimization," *IEEE Access*, vol. 9, pp. 148 277–148 295, Oct. 2021.

[15] R. Mazza, *Introduction to information visualization*. Springer London, Feb. 2009.

[16] K. Itakura and H. Yamaoka, "Development of multi-domain co-simulater for solving social issues," SI2023, The 24th SICE System Integration Division Annual Conference, Tech. Rep., Dec. 2023, in Japanese.

[17] T. Mitomi, F. Makihara, and E. Segawa, "The cognitive bias-informed latent class choice model: A novel approach to predicting human behavior," *SII2024, 2024 IEEE/SICE International Symposium on System Integration*, pp. 559–566, Jan. 2024.

[18] F. Makinoshima, T. Mitomi, F. Makihara, and E. Segawa, "Fully data-driven but interpretable human behavioural modelling with differentiable discrete choice model," *IEEE Access*, vol. 13, pp. 117 420–117 429, Jul. 2025.

[19] I. Ono, M. Ichikawa, and H. Deguchi, "Soars toolkit for large-scale agent-based simulations," SSI2020, The Society of Instrument and Control Engineers, Tech. Rep., Nov. 2020, in Japanese.

[20] I. Ono, Y. Yoshii, K. Nagakane, K. Kojima, M. Ichikawa, and H. Deguchi, "Stage-oriented agent role simulator toolkit and its runtime evaluation by 100,000,000-agent-scale covid-19 simulation," 27th SICE Symposium on Technical Committee on Social Systems, Tech. Rep., Mar. 2022, in Japanese.

[21] K. Ikeda and I. Ono, "Natural evolution strategy for mixed-integer black-box optimization," *Proceedings of The Genetic and Evolutionary Computation Conference*, pp. 831–838, Jul. 2023.

[22] T. Ito, S. Matsuzoe, T. Iwahashi, H. Yamaoka, M. Ueki, K. Nagakane, H. Morozumi, K. Ikeda, and I. Ono, "A method for revealing implicit emphasized criteria for decision makers on social issues," *2024 IEEE Congress on Evolutionary Computation (CEC)*, pp. 1–9, Aug. 2024.