

Development of a Data-Driven Selective Assembly Technique for Subcomponents

OKUBO Eiichi : AI Transformation Promotion Department, Intelligent Information Management Division, IHI Corporation

KAWAGUCHI Akihiro : AI Transformation Promotion Department, Intelligent Information Management Division, IHI Corporation

In the product assembly process in manufacturing, slight variations in the shape or characteristics of subcomponents can sometimes affect the quality after assembly. In this paper, we propose a method that predicts the probability of failure for subcomponent combinations using a machine learning model trained on historical data, and searches for the optimal combinations using the Hungarian algorithm. Through simulations, we demonstrated that the proposed method can reduce the defect rate compared to random matching, suggesting its potential effectiveness for improving manufacturing efficiency and reliability.

1. Introduction

Many of the IHI Group's products are composed of multiple subcomponents. Due to increasingly stringent product requirements in recent years, even if individual subcomponents meet their respective requirements, subtle compatibility issues can sometimes prevent the assembled product from satisfying those requirements. On the other hand, there are limits to the machining precision of subcomponents, and a certain degree of manufacturing variation in shape or characteristics must be tolerated. Against this backdrop, there is a strong demand for technologies that can reliably produce products that satisfy functional requirements at low cost.

In recent years, a method called selective assembly, which accounts for variations in subcomponents within a lot to improve the quality of assembled products, has been widely studied to address these problems. For example, Yamada⁽¹⁾ proposes a selective assembly method that refers to the characteristic information of subcomponents within a lot to reduce the occurrence of leftover components while minimizing variations in differences in their characteristic information of subcomponents. However, much of the prior research focuses solely on the variations in subcomponent characteristics, such as dimensional differences, and cannot directly address how these characteristics affect the quality evaluation of the assembled product.

Therefore, this study proposes to quantify the risk directly related to the actual acceptance/rejection decision when combining subcomponents within a lot, using a failure probability model learned from past manufacturing data, and to incorporate this risk as the cost of solving the optimal assignment problem using the Hungarian algorithm⁽²⁾. Specifically, the probability of failure is estimated from the combination features of subcomponents A and B, and this is reflected in the cost matrix to efficiently search for the

matching that maximizes the number of acceptable products. Furthermore, by applying the consideration of uncertainty used in Bayesian optimization⁽³⁾, a cost function is defined that can flexibly handle model errors and unknown variation factors. This method aims to reduce the risk of rejection in increasingly diverse manufacturing processes by incorporating into the combination strategy a perspective on quality evaluation that cannot be captured solely by conventional optimization based on dimensional differences.

2. Problem setting

This paper examines an optimal matching algorithm for producing products by combining two different subcomponents, A and B.

Specifically, we consider a situation in which a lot contains n units of subcomponent A and m units of subcomponent B, each given a feature vector $\mathbf{a}_i$ ($i = 1, \dots, n$) and $\mathbf{b}_j$ ($j = 1, \dots, m$) respectively. These feature vectors quantify the quality and characteristics of each subcomponent.

The goal is to use these feature vectors to optimize the combination of subcomponents and maximize the number of acceptable products produced from the lot, that is, to minimize the number of defective products. **Figure 1** shows the problem setting in this study.

Specifically, we aim to minimize the expected number of defective products in a lot, which is given by the following equation:

$$\sum_{i=1}^n \sum_{j=1}^m \delta_{i,j} \times P_{i,j} \dots\dots\dots (1)$$

where $\delta_{i,j}$ is a binary variable that takes the value 1 when subcomponents A_i and B_j are combined, and 0 otherwise, and $P_{i,j}$ represents the unknown probability of failure for the combination of subcomponents A_i and B_j .

The following constraints are considered in the problem setting:

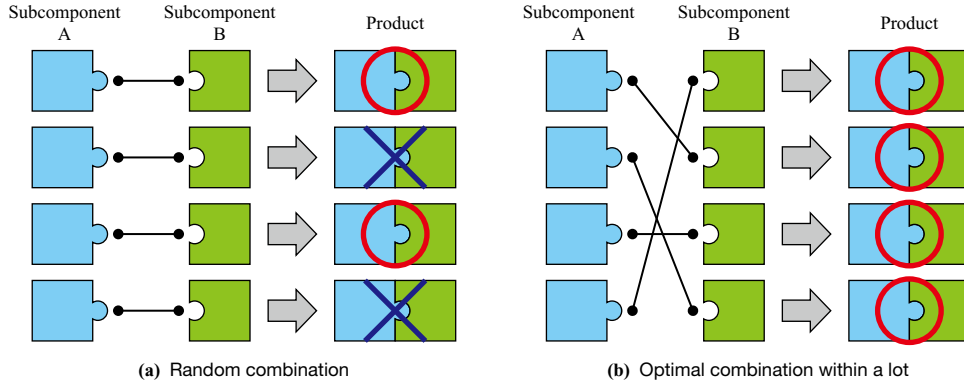


Fig. 1 Problem setting in this study

- Each subcomponent is used only once.
- A total of $\min(n, m)$ products are produced.

Under these conditions, we aim to find the optimal combination. **Chapter 3** proposes a specific method for solving this problem.

3. Proposed method

This chapter describes a method for optimizing the combination of subcomponents within a lot.

3.1 Matching optimization

The problem described in **Chapter 2** can be formulated as an optimal assignment problem. In particular, it is known that the Hungarian algorithm can efficiently search for the optimal matching in an optimal assignment problem using a cost matrix C . Given a $k \times k$ combination of subcomponents and a suitable cost matrix C , applying the Hungarian algorithm allows us to search for the minimum-cost matching between rows and columns in $O(k^3)$. Here each element $C_{i,j}$ of the cost matrix C reflects the failure probability $P_{i,j}$ for the combination of subcomponents A_i and B_j .

Although the Hungarian algorithm is typically applied to square matrices, it can also be applied when the numbers of subcomponents A and B in a lot differ ($n \neq m$). This can be achieved by adding dummy subcomponents to the smaller set, each with a sufficiently large, uniform cost relative to any subcomponent in the other set.

3.2 Building a classification model and setting costs

To apply the Hungarian algorithm to optimize the matching of subcomponents, appropriate costs $C_{i,j}$ must be assigned to all combinations within the lot. When considering the optimal combination while taking into account the product's pass probability, the simplest cost is to construct a binary classification model based on past combinations of subcomponents and their pass/fail results, and to use the predicted probability for each combination. Specifically, this method uses the feature vectors $\mathbf{a}_i$ of subcomponent A_i and $\mathbf{b}_j$ of subcomponent B_j , along with an appropriate function h , to generate the feature vector $\mathbf{x}_{i,j} = h(\mathbf{a}_i, \mathbf{b}_j)$ for each combination, and uses the predicted probability of failure $\hat{P}_{i,j}$ from the classification model as the cost $C_{i,j}$. Here, the combination features can be created simply by concatenating vectors or by applying specific functions.

In addition, defining costs that account for prediction

uncertainty allows us to avoid combinations that cause large fluctuations in the probability of failure, and is expected to contribute not only to improved yield but also to enhanced reliability of the overall manufacturing process. Furthermore, when the predictive model is not perfectly accurate, for example, due to insufficient data or unknown factors, appropriately evaluating uncertainty may help mitigate problems arising from the limitations of the model. From this perspective, this study considers multiple costs $C_{i,j}$ for each combination to account for prediction uncertainty, and compares and analyzes them.

- Expected cost

$$C_{i,j} = \mathbb{E}[f(\mathbf{x}_{i,j})] \dots\dots\dots (2)$$

- UCB (upper confidence bound) cost

$$C_{i,j} = \mathbb{E}[f(\mathbf{x}_{i,j})] + \kappa \times \text{std}\{f(\mathbf{x}_{i,j})\} \dots\dots\dots (3)$$

- EI (expected improvement) cost

$$C_{i,j} = 1 - \mathbb{E}[\max\{0, (y^{\text{base}} - f(\mathbf{x}_{i,j}))\}] \dots\dots\dots (4)$$

- PI (probability of improvement) cost

$$C_{i,j} = \text{Pr}[f(\mathbf{x}_{i,j}) \geq y^{\text{base}}] \dots\dots\dots (5)$$

where f is a binary classification model that can account for prediction uncertainty. In this study, Gaussian process classification (GPC) is used. The UCB, EI, and PI costs are inspired by the acquisition functions of the same names in Bayesian optimization. However, in the GPC used in this study, the probability of failure is obtained by applying a probit function as the link function to the latent function; therefore, the UCB cost is defined using the posterior mean and standard deviation in the probability space. In UCB, κ is the weighting coefficient for the standard deviation relative to the mean and adjusts the balance between exploration and utilization. Increasing this value emphasizes uncertainty and encourages exploratory selection, whereas decreasing it emphasizes the mean. Furthermore, the y^{base} used in EI and PI costs differs from the provisional optimal values used in Bayesian optimization for EI and PI; instead, it represents the baseline probability of failure and is set based on past data. Note that Monte Carlo sampling is used to approximate the calculation of UCB and EI costs.

3.3 Overview of the proposed method

Figure 2 shows an overview of the proposed method in this study. As shown in the figure, we first generate the combination feature $x_{i,j}$ from the features a_i and b_j of subcomponents A_i and B_j , respectively, and then use the predictive model f to calculate the cost given in **Section 3.2**. Next, the obtained costs are organized into a cost matrix, and the overall combination is optimized using the Hungarian algorithm.

4. Experimental setup

To verify the effectiveness of the proposed method, we conduct experiments using synthetic data.

4.1 Simulator design

For the experiment, a specific simulator is designed to evaluate the probability of failure for combinations of subcomponents. **Figure 3** shows an illustration of the simulator. Specifically, the following settings are used.

(1) Generation of subcomponents

Five coordinate points are generated for each of subcomponents A and B. Each coordinate point is represented in two dimensions (x, y) , where the x -coordinate is fixed and only the y -coordinate is determined randomly. These coordinate data are used as the feature vectors a_i and b_j for subcomponents A and B, respectively.

(2) Calculation of the probability of failure

The area formed between the line graphs representing the coordinates of subcomponents A and B is defined as the “gap area” (**Fig. 3**), and the failure probability is determined based on this area. Specifically, a sigmoid function is used to model the relationship such that the probability of failure increases with increasing gap area.

4.2 Experimental method

In this experiment, we evaluate the effectiveness of the proposed method using the following procedure.

- Generate training data

Using a simulator, generate N pairs of subcomponents A and B, randomly combine them into products, and assign a pass/fail label to each based on the calculated probability of failure.

- Build a classification model

GPC is applied to the generated training data. In this study, the combination features are constructed by simply concatenating the dimensions of the subcomponents, i.e., $x_{i,j} = \{a_i, b_j\}$.

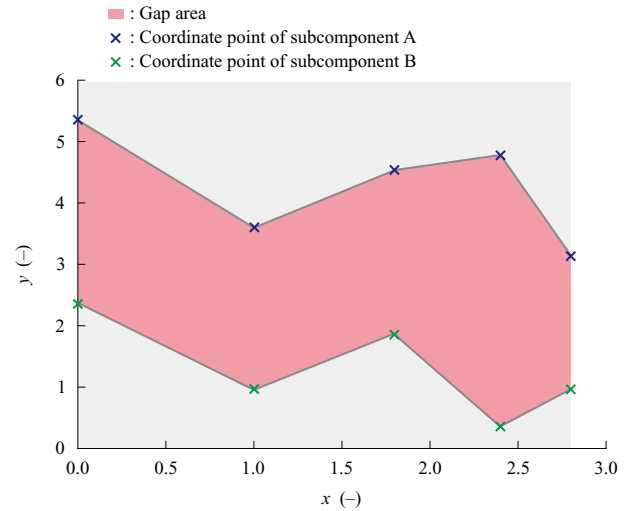


Fig. 3 Illustration of the simulator

- Apply to test data

Then n pairs of subcomponents are prepared as test data, the costs given in **Section 3.2** are calculated for each of the $n \times n$ combinations, and optimization is performed using the Hungarian algorithm, where $\kappa = 1$ for UCB cost, and $y^{base} = 0.5$ for EI and PI costs. Furthermore, the results are compared with those obtained from random combinations to evaluate the effectiveness of the proposed method. The evaluation is performed using Equation (1) with the true probability of failure calculated by the simulator.

5. Experimental results

Figure 4 shows the evaluation of the proposed method for a single lot. The evaluation was performed by applying each method, including the proposed method, to a lot containing 20 pairs of subcomponents. Here, the number of training data points was set to 100. The evaluation index in Equation (1), based on the combinations selected by each method, is shown as “Total cost” for each method; this represents the expected number of defective products out of 20. In addition, the cells in each matrix indicate the true probability of failure for each combination using a color scale; blue indicates values close to 0.0, and red indicates values close to 1.0. The result shown as “True cost matching” (**Fig. 4-(a)**) represents the ideal matching result (3.72/20) obtained using the Hungarian algorithm, assuming that the true probability of failure

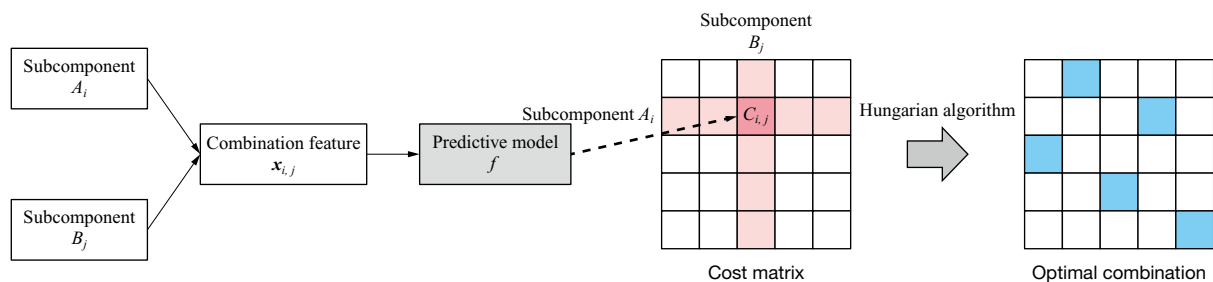


Fig. 2 Overview of the proposed method

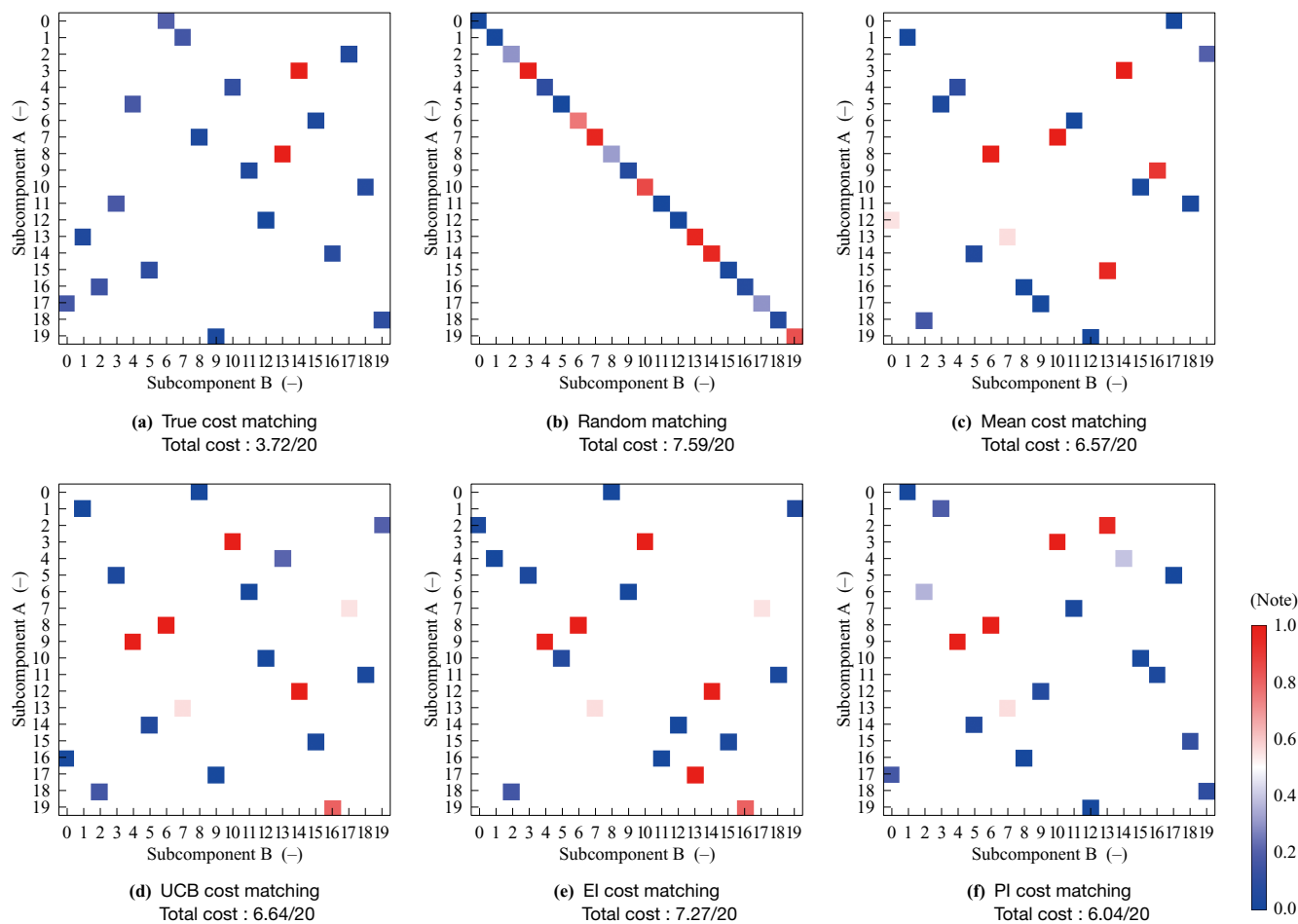


Fig. 4 Evaluation of the proposed method for a single lot

failure is known. On the other hand, “Random matching” in Fig. 4-(b) shows the result obtained by randomly selecting combinations. Since it shows a high value (7.59/20) compared to other methods, it is clear that random selection is inefficient. Comparing the matching results obtained

using different costs, such as expected value, UCB, EI, and PI, the PI-based method achieves the result closest to the ideal (6.04/20).

Figure 5 shows the results when the number of lots is increased. The figure presents the results for the number of

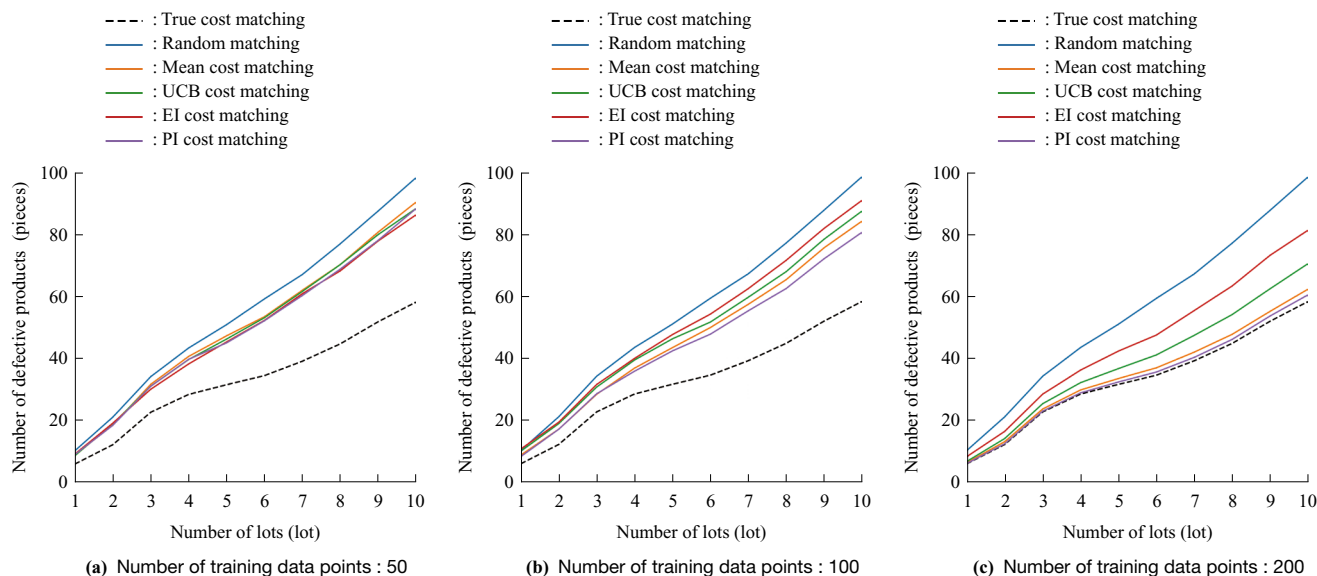


Fig. 5 Evaluation of the proposed method for multiple lots

training data points of 50, 100, and 200, evaluating how much the combinations selected by the proposed method using each cost reduce the average number of defective products compared to random combinations. Regardless of which cost is combined, the proposed method outperforms the random combination (blue curve), in which approximately half of the combinations fail. Furthermore, as the amount of training data increases, the results tend to approach the optimal combination (dashed curve). Under the settings used in this study, the PI-based method yields the results closest to the ideal.

6. Discussion

Bayesian optimization uses acquisition functions that balance exploration and exploitation to select the next point to evaluate. Among these, UCB and EI tend to place a relatively strong emphasis on exploration, actively obtaining information on unknown regions to improve final accuracy. In contrast, PI places greater emphasis on practical application than UCB or EI; therefore, it is considered more suitable for static problem settings like this study, in which the optimal combination is found in a single learning session. In other words, since PI prioritizes combinations with a pass probability exceeding a certain threshold, it does not require sequential learning to explore unknown regions, which may explain why it aligns with a strategy that yields more acceptable products with lower risk than UCB or EI. Furthermore, costs based solely on expected values do not account for uncertainty, leading to concerns about accuracy when training data is limited. However, PI can incorporate the model's variance information and increase the likelihood of exceeding the reference threshold, which may explain why it yields better results in this study.

7. Conclusion

This study proposes an optimal matching algorithm to increase the number of acceptable products when producing products by combining two different subcomponents, A and B. The proposed method constructs a model that predicts the probability of failure of products based on combinations of subcomponents derived from historical manufacturing data, and identifies the optimal combination based on these predictions. In particular, costs that account for prediction uncertainty, inspired by the acquisition function in Bayesian optimization, are designed and applied to combinatorial optimization using the Hungarian algorithm.

Simulation results confirmed that the proposed method

significantly reduces the number of defective products in a lot compared to random combinations. In particular, the matching strategy using PI cost showed the best performance, suggesting that it may also be effective in actual manufacturing processes.

In this study, the problem was formulated as a static one, in which combinations within a lot are optimized using historical data. However, by performing sequential learning as in Bayesian optimization, further improvements in real-time optimization accuracy are expected. Verifying such algorithms remains a topic for future challenge. In addition, from a practical perspective, it is also important to develop methods for feeding back the results of the trained model to the manufacturing process. Specifically, it is necessary to consider how the model's prediction and learning results can be utilized in the manufacturing sites to improve product quality and production efficiency. Furthermore, to handle more complex manufacturing processes, extending the proposed method to the matching of three or more subcomponents is also a topic for future investigation.

The findings of this study contribute to improving efficiency and reliability in the manufacturing industry, and further research and applications are planned for the future.

REFERENCES

- (1) Y. Yamada : Combinatorial Optimization of Selective Assembly in Continuous Production, Transactions of the Japan Society of Mechanical Engineers, Series C, Vol. 60, No. 573, 1994, pp. 1877-1881 (in Japanese)
- (2) H. W. Kuhn : The Hungarian method for the assignment problem, Naval Research Logistics Quarterly, Vol. 2, Iss. 1-2, 1955, pp. 83-97
- (3) B. Shahriari, K. Swersky, Z. Wang, R. P. Adams and N. de Freitas : Taking the Human Out of the Loop: A Review of Bayesian Optimization, Proceedings of the IEEE, Vol. 104, Iss. 1, 2016, pp. 148-175
- (4) E. Okubo, H. Aoyama, A. Kawaguchi, Y. Suzuki and Y. Kawano : Subcomponents Matching Algorithm Considering Product Acceptance Probability, Proceedings of the 39th Annual Conference of the Japanese Society for Artificial Intelligence, Vol. JSAI2025, 2025 (in Japanese)

(Revised and expanded version of the paper⁽⁴⁾ presented at the "2025 Annual Conference of the Japanese Society for Artificial Intelligence")