

## Research Article

# Design Mode Analysis of Pareto Solution Set for Decision-Making Support

Satoru Hiwa,<sup>1</sup> Tomoyuki Hiroyasu,<sup>2</sup> and Mitsunori Miki<sup>3</sup>

<sup>1</sup> Graduate School of Life and Medical Sciences, Doshisha University, Kyoto 610-0394, Japan

<sup>2</sup> Faculty of Life and Medical Sciences, Doshisha University, Kyoto 610-0394, Japan

<sup>3</sup> Faculty of Science and Engineering, Doshisha University, Kyoto 610-0394, Japan

Correspondence should be addressed to Satoru Hiwa; [shiwa@mis.doshisha.ac.jp](mailto:shiwa@mis.doshisha.ac.jp)

Received 21 June 2014; Accepted 9 October 2014; Published 20 November 2014

Academic Editor: Kannan Krithivasan

Copyright © 2014 Satoru Hiwa et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Decision making in engineering design problems is challenging because they have multiple and conflicting criteria and complex correlation between design parameters. This study proposes a decision-making support methodology named design mode analysis, which consists of data clustering and principal component analysis (PCA). A design mode is indicated by the eigenvector obtained by PCA and reveals the dominant design parameters in a given dataset. The proposed method is a general framework to obtain the design modes from high-dimensional and large datasets. The effectiveness of the proposed method is verified on the conceptual design problem of the hybrid rocket engine.

## 1. Introduction

Multiple-criteria decision making (MCDM) [1] is used in a class of problems where decisions are made among multiple and conflicting criteria (objectives). Since a single best decision for MCDM problems does not exist, they are solved by seeking a set of available alternative decisions. After the alternatives are obtained, a decision maker (DM) chooses among their preferred solutions. In this procedure, the set of alternatives are represented as a Pareto solution set. This procedure is also known as multiobjective optimization. Nondominated solutions are those where no objective function value can be improved without worsening another objective function value. Multiobjective optimization algorithms derive good approximations for the Pareto optimal solutions, which cannot be further improved.

Once solutions have been properly converged, they can be used to make decisions. However, choosing the best compromise with multiple and conflicting objectives without any information about the target problems to aid decision making is difficult. To solve this problem, many researchers have proposed effective use of optimization results to improve our understanding of the target problems.

Obayashi et al. [2, 3] proposed the multiobjective design exploration framework, which consists of the Kriging Model, multiobjective genetic algorithm, analysis of variance, and a self-organizing map. It can explore a broad part of the decision space and derive many Pareto solutions (alternatives) in a reasonable time through Kriging metamodeling of the objective functions. The information tradeoff between multiple objectives and decision space characteristics is broadly outlined by visualizing the decision and objective spaces using data mining methodologies, such as self-organizing maps.

Oyama et al. [4, 5] used proper orthogonal decomposition (POD) to formulate decision variables involved in designing the airfoil shape and revealed that any design can be decomposed into the mean vector and the fluctuation vector, which is expressed by the linear sum of normalized eigenvectors and orthogonal base vectors. One of the advantages of their approach is that we can understand the representative design types as well as the parameters. After analyzing the fluctuation vector, the parameters are constructed from a large and high-dimensional dataset.

In this paper, inspired by the work of Oyama et al., we extend the concept of the fluctuation vector and define a new

concept—design mode. Based on this concept, we propose “design mode analysis,” which is an analytical framework for finding the design modes and using them effectively. It consists of data clustering and principal component analysis (PCA) and will extract the representative design types along with their characteristics. These information aid the decision making. Note that PCA is same algorithm as POD, but the different name is used depending on the application field. In this paper, we use the name of PCA.

This paper is organized as follows. Section 2 gives the definition of the multiobjective optimization problem. Section 3 introduces the concept and framework of the proposed method, as well as its application to engineering design problems. In Section 4, the proposed method is applied to the conceptual design problem of a hybrid rocket engine. The paper concludes with Section 5.

## 2. Multiobjective Optimization Problem

In many engineering design problems, MCDM is often treated as a multiobjective optimization problem. It can be formulated as

$$\begin{aligned} & \text{minimize } F(\mathbf{x}) = (f_1(\mathbf{x}), \dots, f_m(\mathbf{x}))^T \\ & \text{subject to } \mathbf{x} \in S, \end{aligned} \quad (1)$$

where  $F(\mathbf{x})$  is an objective vector, consisting of  $m$ -objective functions  $f_i : \mathcal{R}^n \rightarrow \mathcal{R}$  for all  $i \in \{1, \dots, m\}$ .  $S \subset \mathcal{R}^n$  is called the decision variable space and is defined as

$$\begin{aligned} S = \{ \mathbf{x} \in \mathcal{R}^n \mid & g_j(\mathbf{x}) \leq 0 \quad (j = 1, \dots, l), \\ & h_k(\mathbf{x}) = 0 \quad (k = 1, \dots, p) \}. \end{aligned} \quad (2)$$

Here,  $g_j(\mathbf{x})$  and  $h_k(\mathbf{x})$  are inequality and equality constraints, respectively. Multiobjective optimization problems consist of two definitions that handle the tradeoff between the multiple criteria (objectives). Notably, these definitions are prescribed for minimization problems.

*Definition 1* (Pareto dominance). For  $\mathbf{x}_1$  and  $\mathbf{x}_2$  ( $\mathbf{x}_1 \neq \mathbf{x}_2$ ),  $\mathbf{x}_1$  is said to *dominate*  $\mathbf{x}_2$  if for at least one  $i = 1, \dots, m$ ,  $f_i(\mathbf{x}_1) \leq f_i(\mathbf{x}_2)$  and  $f_i(\mathbf{x}_1) < f_i(\mathbf{x}_2)$ .

*Definition 2* (Pareto optimality). Let  $\mathbf{x}_0 \in S$ . Then,  $\mathbf{x}_0$  is *Pareto optimal* when there are no other solutions in  $S$  dominating  $\mathbf{x}_0$ .

Based on these definitions, the Pareto solution set or non-dominated solution set is a subset of all the Pareto-optimal solutions. Typically, multiobjective optimization algorithms seek good approximations in the Pareto optimal solutions by evolutionary multiobjective optimization (EMO). EMO is a popular approach because it obtains many nondominated solutions simultaneously in a single run. Recently, a number of EMO algorithms have been developed and improved to derive well-converged and well-spread nondominated solutions sets. One of the most successful frameworks in recent EMO algorithms is the multiobjective evolutionary algorithm based on decomposition (MOEA/D) proposed by Zhang and Li [6]. It has been widely used in many real-world applications [7, 8].

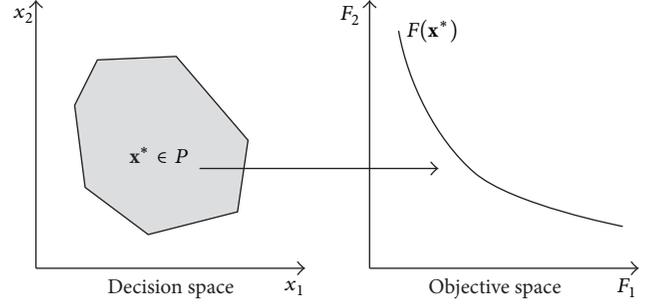


FIGURE 1: Representation of the decision space and the objective space.

## 3. Design Mode Analysis

*3.1. Definition of Design Mode.* Decision variables are defined as a set of parameters that determine the solutions to optimization problems. For example, in product design problems, they frequently represent the size, weight, and shape of the product. Objective functions indicate the goal of the product design, such as the performance and cost of the target product. The decision variables and objective functions form the decision space and the objective space, respectively, as shown in Figure 1. In general, the optimization process explores the decision space with the aim of minimizing or maximizing the objective functions. Many multiobjective evolutionary algorithms aim to improve the diversity and convergence of the non-dominated solutions, especially in the objective space [6, 9].

However, in product design problems, the design is also characterized by the decision variables. For example, if the product mass is successfully minimized and the characteristics of the decision variables of the optimum solutions are analyzed, a weight-saving design strategy may be found for the product.

In this study, we incorporate the “design mode” concept into decision space analysis as an essential perspective. The concept is derived from PCA. These methods extract the dominant characteristics of the target dataset by decomposing high-dimensional data into low-dimensional descriptions using a set of principal component vectors, whose directions correspond to maximum variance among the variables. Oyama et al. [4, 5] applied PCA in their analysis of airfoil shape design and reported its effectiveness. They focused on designing the shape of the product, that is, seeking the principal airfoil shapes that made up the Pareto solution set.

In this study, we generalize and extend the concept to deal with all types of decision parameters during product design. Directions indicated by principal component vectors guide us to change the decision variables for constructing the designs in the given dataset. Furthermore, the contribution ratio of each decision variable to the principal component vectors shows its importance for creating the designs. We assume that the principal component vector gives us important information for the engineering design and define it as “design mode.” An analytical framework based on this design mode is also proposed.

Let  $X^{(i)} = (x_{i1}, \dots, x_{im}) \in \mathcal{R}^m$  be a decision variable of the  $i$ th Pareto solution, and let  $X = (X^{(1)}, \dots, X^{(N)})$  be

a Pareto solution set of size  $N$ . In PCA, the following optimization problem is solved, and the vector  $\mathbf{w}$  that maximizes variation in the decision variables is selected:

$$\max_{\|\mathbf{w}\|=1} \text{Var}[\mathbf{w}^T X] = \max_{\|\mathbf{w}\|=1} \mathbf{w}^T \text{Var}(X) \mathbf{w}, \quad (3)$$

where the matrix  $\text{Var}(X) = C$  is the covariance matrix of the dataset  $X$ . Let  $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_m$  be the eigenvalues for  $C$ , and let  $\mathbf{v}_j$  be their corresponding eigenvectors. Then, the  $k$ th principal component of the data is represented by

$$Y^{(i,k)} = \mathbf{v}_k X^{(i)}. \quad (4)$$

The original dataset can be decomposed into a low-dimensional representation by selecting a certain number of the principal components. The most useful contribution of PCA to the Pareto dataset is that the entire Pareto dataset can be approximated by the mean vector of the dataset and a linear combination of a specified number of eigenvectors:

$$X^{(i)} \cong \mathbf{u} + \sum_{k=1}^p \alpha_k \mathbf{v}_k. \quad (5)$$

A useful criterion for choosing the number of components is the cumulative proportion of the variance  $P$ , defined below. This metric indicates the extent to which each principal component explains the original dataset. Consider

$$P = \frac{\sum_{i=1}^p \lambda_i}{\sum_{i=1}^m \lambda_i}. \quad (6)$$

From the axes associated with each eigenvector, we construct a meaningful new decision space. Here, we define each eigenvector as “design mode.” Along the axes indicated by the design modes, we examine the features of the obtained solutions. To analyze the correlation between the axes of the decision variables in the original decision space and the new axis indicated by the  $k$ th design mode, we calculate the component loading:

$$r_{kj} = \sqrt{\frac{\lambda_k}{s_j}} v_{kj}, \quad (7)$$

where  $s_j$  is the variance of the  $x_j$  and  $v_{kj}$  is the  $j$ th element of eigenvector  $\mathbf{v}_k$ . The component loading specifies the importance of the decision variable  $x_j$  in constructing the  $k$ th principal component. Notably, if PCA is executed on a standardized dataset, the covariance matrix  $C$  is equivalent to the correlation matrix of  $X$ . In this case,  $\mathbf{u} = \mathbf{0}$  and  $s_j = 1$ .

This section has outlined some important aspects of design mode analysis.

- (i) Applying PCA to a Pareto set enables the extraction of dominant designs and decision variables to be extracted.
- (ii) Pareto solutions can be approximated by the mean vector of the dataset and the linear combination of a certain number of eigenvectors.
- (iii) Each eigenvector forms a meaningful new axis in the decision space, and correlations between the original and new axes are quantitatively evaluated by the component loadings.

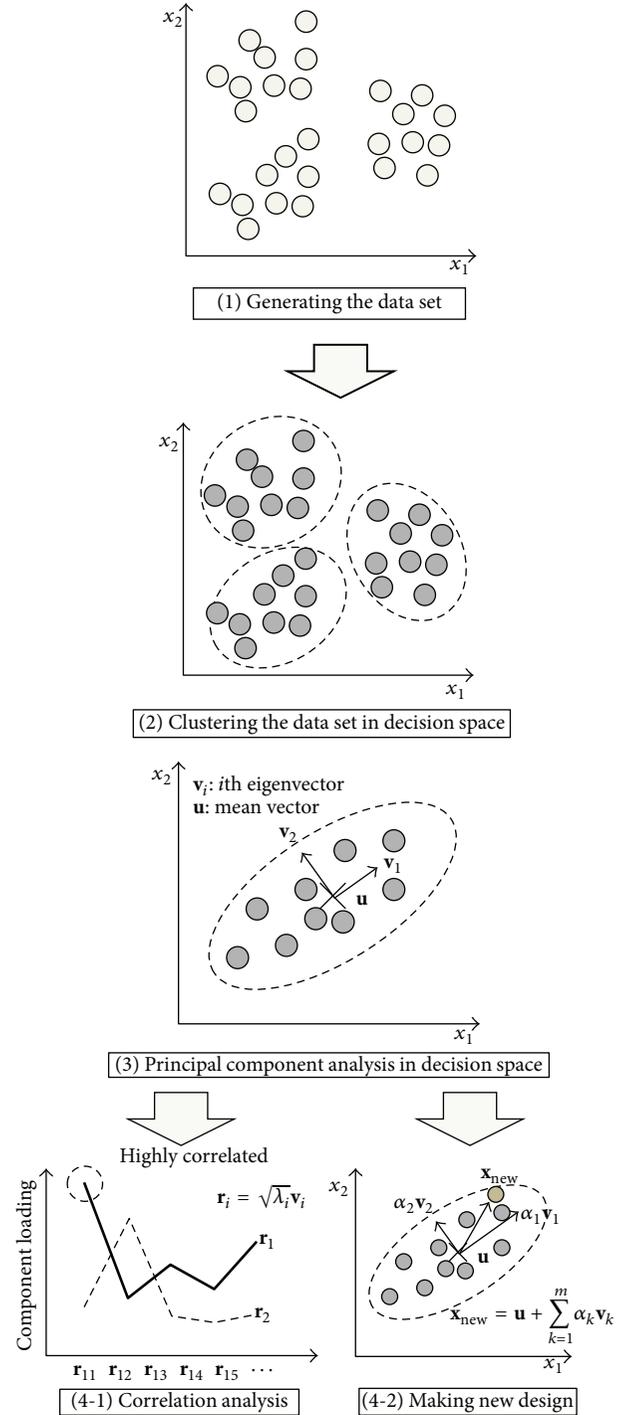


FIGURE 2: Design mode analysis framework.

3.2. Framework of Design Mode Analysis. In the above subsections, we explained the concept of the design mode. Here, we explain the framework of design mode analysis. The main steps of the design mode analysis are as follows: generate the dataset; cluster the dataset; perform PCA; and either perform correlation analysis or construct a new design. The proposed framework is illustrated in Figure 2.

The differences of our proposed approach from the conventional method are as follows:

- (1) data clustering is incorporated into the analysis;
- (2) design mode characterization is achieved by studying the component loading of each design mode.

The following subsections provide detailed descriptions of each step in the framework.

**3.2.1. Generating the Dataset.** The first step in our proposed analysis is to choose how to generate the dataset (design examples). One of the easiest ways to do this is to use statistical sampling methods such as Latin hypercube sampling. An expert engineer can also supply representative designs. The dataset generation is dependent on the objective of the design mode analysis. For example, if the engineer desires a uniform, equal study of the design space characteristics, statistical sampling is preferred. Alternatively, if the aim is to elucidate the design methodology of an expert, then many design examples constructed by the experts must be collected. In this study, we investigate the decision space characteristics around the Pareto solution set and then obtain the solutions by EMO. Notably, our proposed framework does not determine the data generation method. Multiobjective optimization is only one of the tools available to generate characteristic designs.

**3.2.2. Clustering the Dataset.** By incorporating data clustering into data preprocessing, we can obtain reliably distinguished design modes. Data clustering for the Pareto solution set can be applied to either the decision or the objective space. However, since design mode analysis is conducted on the decision space, clustering should ideally be performed on the decision space. Data clustering was not considered as a part of the conventional method proposed by Oyama et al. [4, 5]. However, since data clustering screens all the input designs and divides them into representative designs and their similar counterparts, its inclusion is advantageous. Meanwhile, any of the data clustering methods are suitable. In this study, we adopt  $k$ -means clustering, which divides the dataset into  $k$  clusters. Each datum is then assigned to the cluster with the nearest mean vector.

**3.2.3. Principal Component Analysis on the Dataset.** Once the clustering process is complete, each cluster is subjected to PCA. Different design modes are expected to be derived from each cluster. Moreover, the mean vector of each cluster is the representative design of each cluster, and all solutions in each cluster are approximated by a linear combination of the design modes (eigenvectors), in the coordinate system whose origin is the mean vector.

**3.2.4. Correlation Analysis.** Having obtained the design modes for each cluster, we study each design mode by referring to the component loadings. This process identifies the dominant decision variable in the design mode, thereby revealing the important factors for creating the new designs in each cluster. Decision variables that make low contributions to the design mode can be eliminated from the variables and set as

constants instead. The component loadings characterize the design modes and give us their features. This process was not considered in the conventional method either. It will give us useful information about the target problem.

**3.2.5. Constructing a New Design.** Our proposed method, called design mode analysis, is employed as an analytical tool and a design support tool. Since most designs in each cluster can be approximated in the decision space formed by the design modes, we can easily generate new designs with the same features in this space, using (5). Note that random sampling in the original decision space does not easily create design sharing characteristics for a specific design, especially if that decision space is high-dimensional. We summarize the proposed framework in the pseudocode shown in Algorithm 1.

The following subsection demonstrates the effectiveness of each step in our proposed method through a series of experiments.

**3.3. Case Study: Multiobjective 0/1 Knapsack Problem.** In this subsection, our proposed design mode analysis method is applied to the multiobjective 0/1 knapsack problem (MOKP). The effectiveness of the method is investigated in terms of the following outcomes:

- (i) how well each design mode characterizes the dataset,
- (ii) whether the data clustering process effectively distinguishes the design modes,
- (iii) whether any design in each cluster can be approximated by the mean design and a linear combination of the design modes in the cluster,
- (iv) whether our proposed method can be applied to binary-valued problems.

**3.3.1. Experimental Setup.** The target design problem, MOKP, is a multiobjective extension of the classic 0/1 knapsack problem (KP), a nondeterministic polynomial time complete combinatorial problem. When we try to pack the items, which have their own values and weights, into a knapsack with capacity constraints, choosing the items to maximize the total profit of the items selected is important, as we cannot pick them all. In combinatorial optimization, this is popularly known as the classic KP and often appears in real-world decision-making problems in various fields, such as production scheduling or portfolio management. Figure 3 illustrates the classic KP. In KP, a binary decision variable is used to indicate whether each item is included in the knapsack or not, and the total profit of the items packed into the knapsack is defined as the objective function. The objective is to find a subset of the items with a total weight not exceeding the knapsack capacity, while maximizing the total profit. MOKP is an extended form of KP where the number of the knapsacks is simply increased.

The  $k$ -objective MOKP with  $N$  decision variables is formulated as

$$\begin{aligned} & \text{maximize} && f_i(\mathbf{x}) = \sum_{j=1}^N x_j p_{(i,j)}, \quad x_j \in \{0, 1\} \\ & \text{subject to} && g_i(\mathbf{x}) = \sum_{j=1}^N x_j w_{(i,j)} \leq W_i, \quad 1 \leq i \leq k, \end{aligned} \quad (8)$$

- (1) Generate a design dataset  $X = (X^{(1)}, \dots, X^{(N)})$  of size  $N$ .
- (2) Divide the dataset into  $H$  clusters by using data clustering.
- (3) **for**  $i = 1$  to  $H$  **do**
- (4) Extract the design mode  $\mathbf{v}_k = (v_{k1}, v_{k2}, \dots, v_{km})$  by applying the PCA ( $k = 1, \dots, N$ ) on  $i$ th cluster.
- (5) Calculate and study the component loading  $\mathbf{r}_k = (r_{k1}, r_{k2}, \dots, r_{km})$  for each  $\mathbf{v}_k$  using  $r_{kj} = \sqrt{\lambda_k} / s_j v_{kj}$
- (6) Choose a base design  $\mathbf{b}$  from the  $i$ th cluster, or calculate a mean vector instead of it.
- (7) Choose a number of design modes  $p$  used for generating new designs.
- (8) Generate a new design  $X'$  based on 
$$X' = \mathbf{b} + \sum_{k=1}^p \alpha_k \mathbf{v}_k.$$
 Coefficient  $\alpha_k$  is an arbitrary constant.
- (9) **end for**

ALGORITHM 1: Design mode analysis.

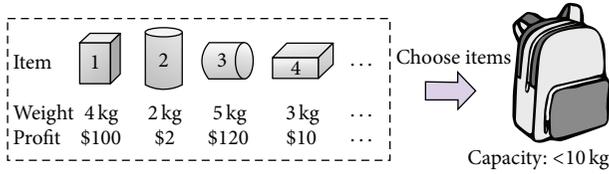


FIGURE 3: Knapsack problem.

where  $p_{(i,j)}$  indicates the profit of the  $j$ th item in calculating the function value for the  $i$ th knapsack. In the constraint function,  $w_{(i,j)}$  is the weight, and  $W_i$  is the upper limit value of  $w_{(i,j)}$ . The test case is the 2KP50-11 dataset selected from MCDMLib [10], a collection of datasets available for testing various multiobjective optimization problems. The study constitutes a biobjective (two-knapsack) problem with 50 decision variables (items). These items in each knapsack are weighted the same, but their profits differ.

The nondominated solution set of 2KP50-11 is obtained by nondominated sorting genetic algorithm II (NSGA-II) [9], one of the most efficient MOEAs: two-point crossover (crossover rate = 1.0) and bit-flip mutation (mutation rate = 1/chromosome length) are used. Population size is set at 120. A single run of NSGA-II is terminated after 1000 generations, and 30 runs of NSGA-II are executed. The crossover and mutation rate used here follow the practice in [6, 9]. Population size and the number of generations are empirically chosen (not optimized) here. Although the choice of the genetic algorithm parameters (population size, number of generations, crossover, and mutation rates) may result in different optimum solutions, it is not the focus of our study to adjust and study the parameter setting. The multiobjective optimization is only the tool to generate the dataset to be analyzed.

Notably, MOKP comprises binary-valued decision variables. Since PCA executes real-valued variables, we must assume that the binary variables are continuous [0, 1] when applying PCA to the solution dataset of MOKP.

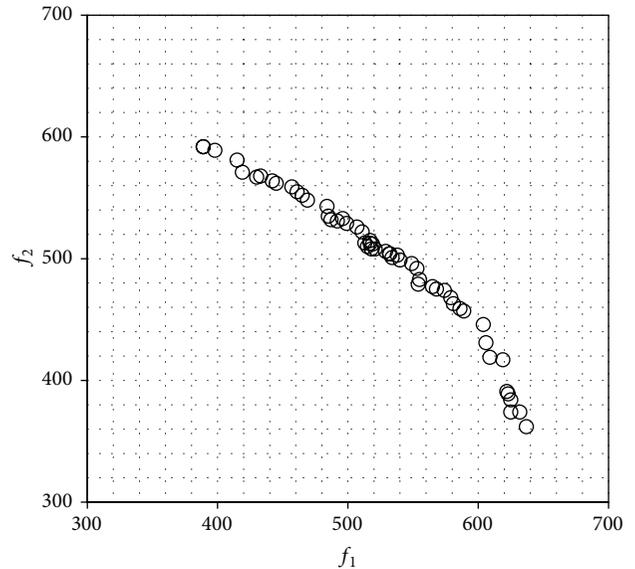


FIGURE 4: Nondominated solutions obtained by NSGA-II.

3.3.2. *Results and Discussion.* Figure 4 shows the nondominated solutions obtained by NSGA-II. After deleting the overlapped data obtained in the 30 runs, we obtained 53 solutions.

PCA was also carried out on the decision variables of the nondominated solutions sets. The results of running PCA on MOKP nondominated solution sets are summarized in Table 1. The original dataset can be explained if the cumulative proportion of explained variance is  $\geq 0.8$ . In this case, the first eight design modes are essential to explain the dataset. To visualize the characteristics of the design modes, the component loadings are calculated and plotted in Figure 5. The component loadings of elements with zero variance are plotted as zero, because when they are equal to zero, (7) is

TABLE 1: Summary of running PCA on a MOKP nondominated solution set.

Design mode ( $i$ )	1	2	3	4	5	6	7	8	9	10
Standard deviation ( $\sqrt{\lambda_i}$ )	0.92	0.76	0.59	0.54	0.48	0.46	0.43	0.41	0.39	0.37
Proportion of variance	0.25	0.17	0.10	0.08	0.07	0.06	0.05	0.05	0.04	0.04
Cumulative proportion	0.25	0.41	0.51	0.60	0.66	0.72	0.78	0.83	0.87	0.91

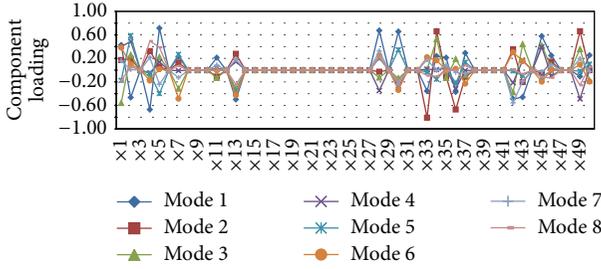


FIGURE 5: Distribution of the component loadings (Modes 1 to 8).

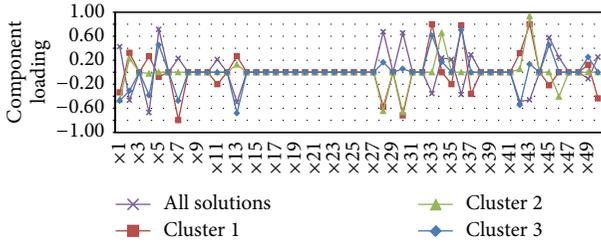
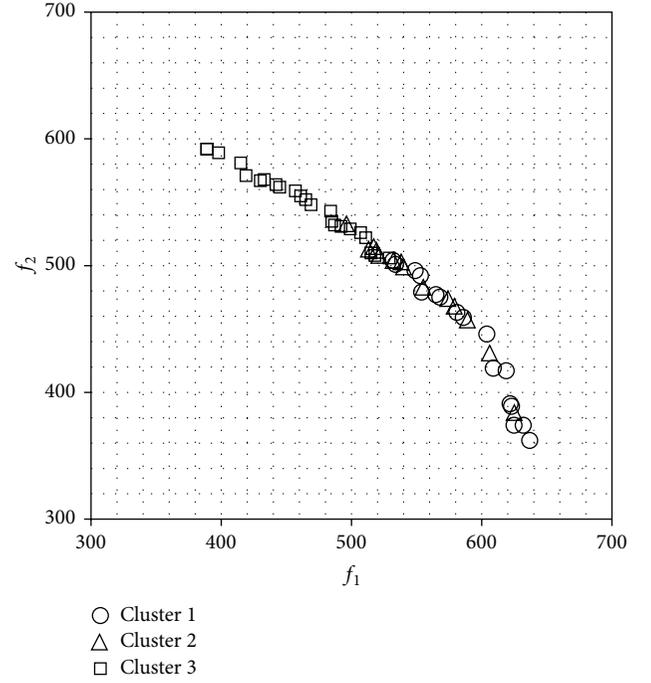


FIGURE 6: Distribution of the component loadings for the first design mode of each cluster.

FIGURE 7: Three clusters obtained by  $k$ -means clustering.

incalculable. To interpret this distribution, we check each element; if the absolute value of the  $i$ th element in  $k$ th design mode is large, then packing or discarding the  $i$ th item strongly affects the  $k$ th design mode. Each design mode has a unique distribution of its component loadings, indicating that several different strategies will successfully pack the items into two knapsacks while maximizing the profits.

Next, to evaluate the effectiveness of the data clustering,  $k$ -means clustering was performed on the dataset, yielding three clusters ( $k = 3$ ). Figure 7 plots these clusters in the objective space.

Here, the first design mode is the design mode corresponding to the eigenvector with the maximum eigenvalue. Following PCA, a different design mode (i.e., the first design mode) was obtained for each cluster and for the entire dataset, as shown in the component loading plots of Figure 6. Thus, apparently, data clustering distinguishes the design modes effectively.

If PCA successfully extracts the design modes of each cluster, any design in any cluster can be approximated by the mean design and a linear combination of the design modes within the cluster. To verify this assumption, we approximated the nondominated solution set by its mean vector and effective design mode vectors (sufficient to achieve a cumulative proportion of the variance  $\geq 0.80$ ). The procedure for approximating the solution set is shown below.

#### Design Approximation Method.

- (1) Choose a target design  $X^{(i)} = (x_{i1}, \dots, x_{im})$  from the dataset  $X = (X^{(1)}, \dots, X^{(N)})$ .
- (2) Choose  $p$  eigenvectors so as to account for a cumulative proportion of the variance  $P \geq 0.80$ .
- (3) Find optimum coefficients  $\alpha_j$  ( $j = 1, \dots, p$ ) of the approximated design calculated by (5) so as to minimize the sum of the squared error (SSE) between the target and approximated design in the decision space:

$$\sum_{i=1}^m (x'_{ik} - x_{ik})^2, \quad (9)$$

where  $x_{ik}$  is the  $k$ th decision variable of the  $i$ th design in the dataset and  $x'_{ik}$  is the approximated design of  $x_{ik}$ . Notably, if there are the constraints in decision variables in the original dataset, they should be added to (9).

As an example, consider a target design in the solution with a maximum  $f_1$  value of  $(f_1, f_2) = (637, 362)$ , and assume that the design belongs in Cluster 1. To obtain the approximation error in the objective space, we must evaluate the objective function value of the approximated design. The

TABLE 2: Summary of design approximation using design modes.

	$f_1$	$f_2$	$g$	SSE	$D_h$	$D_f$	$p$
Target design	<b>637</b>	<b>362</b>	<b>187</b>	—	—	—	—
Mean design (of all the solutions)	510	369	154	$5.02E + 00$	5	127.19	—
Mean design (of Cluster 1)	684	508	212	$3.14E + 00$	5	153.38	—
Approximated design (using design mode of all the solutions)	543	327	156	$9.03E + 00$	1	100.31	8
Approximated design (using design mode of Cluster 1)	<b>637</b>	<b>362</b>	<b>187</b>	<b><math>3.40E - 01</math></b>	<b>0</b>	<b>0.00</b>	<b>7</b>

solution obtained by this approximation method can be real valued. To evaluate the objective function values of MOKP, the approximated decision variables should be converted to binary values. In this experiment, the approximated variable is rounded to the nearest whole number. If the integer lies outside of  $[0, 1]$ , it is assumed to be 0 or 1; that is, if it is smaller than 0, it is assumed to be 0, and if it is larger than 1, it is assumed to be 1.

Results of the design approximation are summarized in Table 2, where  $g$  is the constraint value in (8). SSE is the approximated error in the decision space defined by (9).  $D_h$  is the Hamming distance between the target and the approximated design in the decision space.  $D_f$  is the Euclidean distance between the target and the approximated design in the objective space, and  $p$  is the number of design modes used in the approximation. We tried two different methods of design approximation. In the first, PCA was applied to all solutions, and each design was approximated by the mean vector and the design modes of all solutions. In the second, PCA and design approximation were executed on the Cluster 1 dataset. The results of both trials are listed for comparison in Table 2. The approximated designs in the objective space are plotted in Figure 8.

Table 2 indicates that the target design was successfully approximated from the data in Cluster 1 alone (SSE =  $3.40E - 01$ ,  $D_h = 0$ ). When the approximation was built from all solutions, the SSE was an order of magnitude greater. These results are also evident in the plots of Figure 8.

To ensure that data clustering effectively distinguishes between the design modes, we analyzed the performance of the approximated design. In this analysis, we adjusted the size of each eigenvector before adding it to the mean vector. The upper charts in Figure 9 show the distributions of the elements of the mean and target design elements, while the lower charts show the distributions of the elements of the modal eigenvectors built into the approximation.

To approximate the target design, elements of the mean design with values different from the target values are altered on addition of the eigenvectors. Arrows on the charts indicate the directions of the altered elements. For instance, observe the 37th variable emphasized by the hatched pattern in Figures 9(a) and 9(b). In the design approximation without data clustering (Figure 9(a)), the magnitude of the 37th variable is very much smaller ( $\pm 0.1$ ) than that obtained after data clustering (Figure 9(b)). In this case, a larger coefficient  $\alpha_j$  ( $j = 1, \dots, p$ ) is required to successfully approximate the 37th variable. However, an appropriate coefficient for a single variable is difficult to determine because the coefficient affects all other elements. Conversely, in the design approximated from

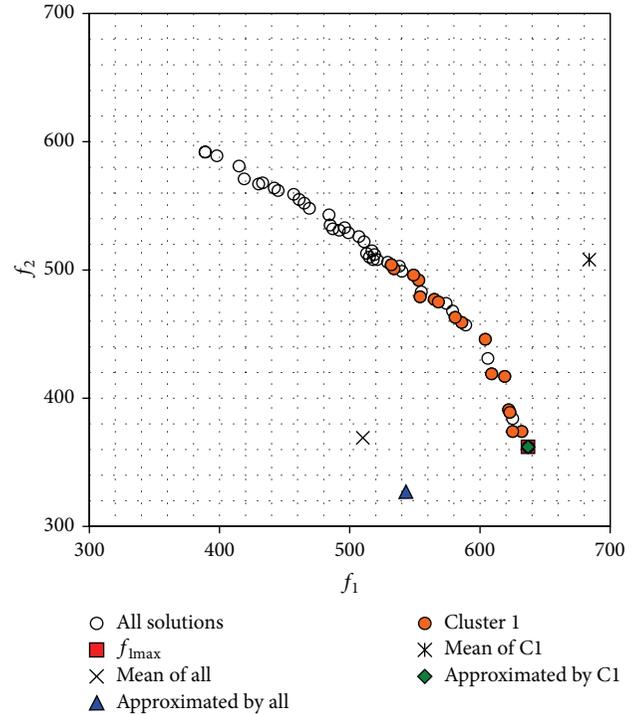


FIGURE 8: Plots of approximated designs in objective space.

the clustered data, the element lacked by the mean design compared to the target design is compensated for the corresponding element of the eigenvectors. These results show that data clustering is effective for extracting precise design modes.

This case study also highlights the importance of granularity in our proposed design mode analysis. In the absence of clustering, we assume that some decision variables contribute negligibly to the design. Thus, granularity exists in the design mode extraction. Data clustering increases the granularity of the design modes. Watanabe et al. [11] proposed an interactive granularity control method, which is applicable to our proposed design mode analysis.

**3.4. Granularity in Design Mode Analysis.** In the above case study, we introduced the concept of “granularity” in design mode analysis. The granularity of the design mode extraction required by DM depends on the situation. For example, if a researcher is interested in the characteristics of specific clusters, he may divide the dataset into several clusters and characterize the clusters by PCA. At the beginning of the

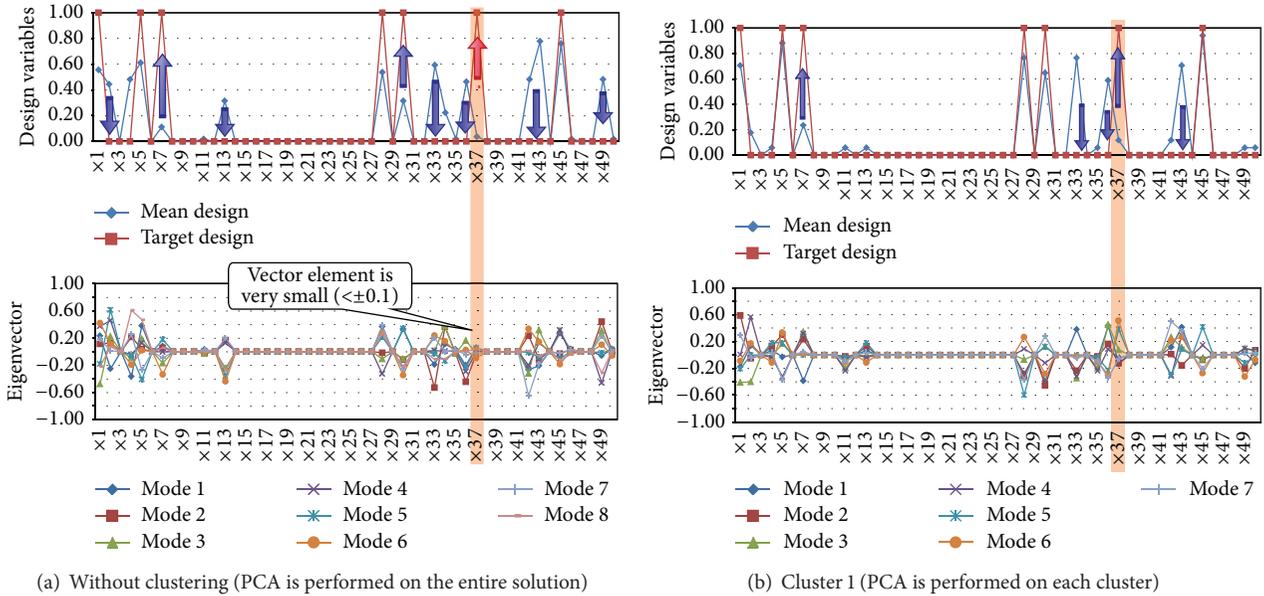


FIGURE 9: Distribution of eigenvectors in design approximation of max solution.

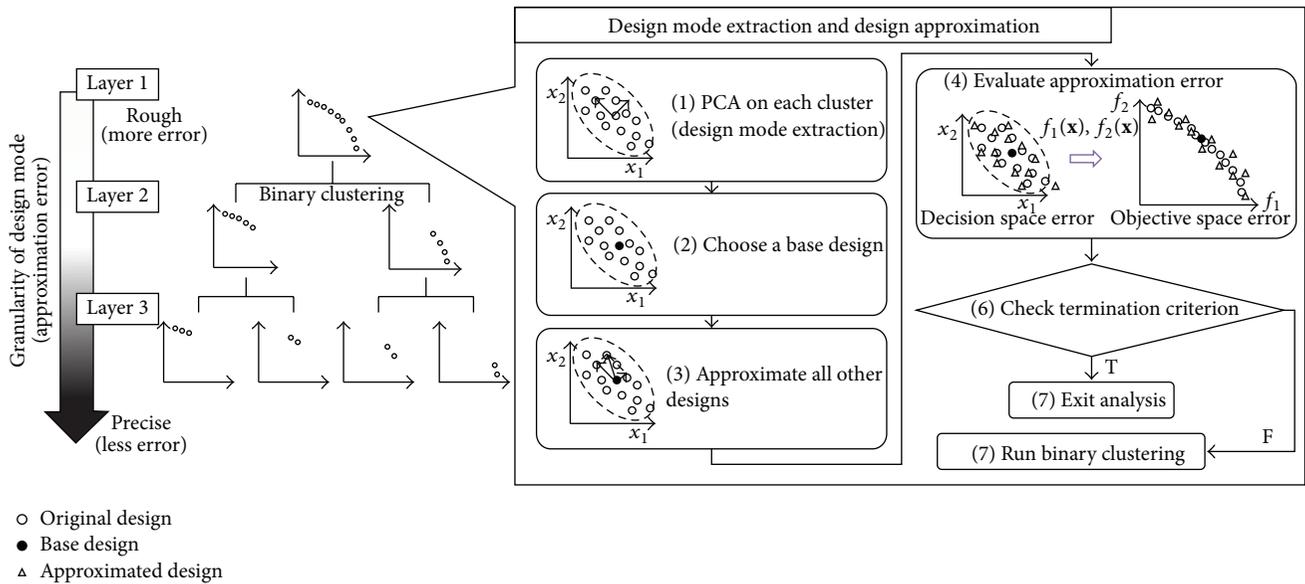


FIGURE 10: Proposed framework of design mode analysis focusing on granularity.

analysis, the characteristics of the decision space can be coarsely determined by imposing a low granularity. Once the design mode has been refined, a high granularity is expected. This subsection focuses on granularity and proposes a more general framework for design mode analysis.

Following Watanabe et al. [11], we adopted a hierarchical approach. The proposed framework iterates binary clustering and PCA and approximates a design for each cluster. Different from Watanabe et al., we controlled the granularity of the design mode analysis by the accuracy of the approximated design. Our proposed framework is illustrated in Figure 10.

The design approximation process is discussed in Section 3.3.2. If the design modes are successfully derived, any design can be approximated by a linear combination of the design modes, as shown in (5). In this equation, the base design is the mean vector of the cluster, but the base design can be any design in the cluster, provided that it retains the average or representative characteristics of its own cluster. The easiest way to choose the representative design is to calculate the mean vector and set it as the base.

Once the design approximation is complete, DM checks whether the analysis has adequately converged or whether

analysis should be continued. The convergence is evaluated by the errors in the decision and objective spaces. DM may specify a threshold for each approximation error. If high quality design modes or design strategies are obtained, DM can terminate the analysis.

Otherwise, the design modes are refined by dividing each cluster into two new clusters and progressing to the next layer of granularity. Notably, PCA and clustering are not performed on clusters of a single datum. In this case, the next layer inherits the cluster.

The above-mentioned procedures yield design modes at any level of granularity.

#### 4. Application to Conceptual Design of Hybrid Rocket Engine

In this section, our proposed design mode analysis method is applied and tested on the conceptual design of a hybrid rocket engine. This problem, which is one of the most useful real-world optimization problems for testing the performance of optimization algorithms [12], was first proposed by Kosugi et al. [13]. The executable software for objective function evaluation is available from the website [14].

**4.1. Problem Definition.** A hybrid rocket engine stores propellant in two different kinds of phases. With the advantages of low environmental impact, flexible thrust control by throttling, and reduced chemical explosion hazard, it is becoming increasingly popular. In the hybrid rocket engine, the thrust and the engine design are strongly correlated because thrust is obtained by combustion in the boundary layer diffusion flame. Thus, designing the solid fuel geometry and the oxidizer supply system is important and difficult.

The hybrid rocket investigated comprises four parts: a payload, an oxidizer tank, a thrust chamber, and a nozzle. The thrust is provided to the hybrid rocket by combustion in a turbulent boundary layer in the thrust chamber. The oxidizer and mass/fuel ratio also affect the thrust. The latter is determined by the fuel parameters, namely, the oxidizer, fuel length, and initial port radius. Hybrid rocket design problems constitute two-objective optimization problems, in which the fuel parameters must be optimized to maximize the altitude gained, while minimizing the gross weight. A schematic of the hybrid rocket is shown in Figure 11.

Here, the six-dimensional decision space comprises the initial mass flow of the oxidizer  $\dot{m}_{\text{oxi}}(0)$  [kg/s], fuel length  $L_{\text{fuel}}$  [m], initial port radius  $r_{\text{port}}(0)$  [m], combustion time  $t_{\text{burn}}$  [s], initial pressure in the combustion chamber  $P_{\text{cc}}(0)$  [MPa], and aperture ratio of nozzle  $\epsilon$ . The two-objective functions aim to maximize the altitude  $H_{\text{max}}$  [km] and simultaneously minimize the gross vehicle weight  $M_{\text{tot}}(0)$  [kg]. The following motion equation is assumed during the flight analysis:

$$a(t) = \frac{T(t) - D(t)}{M_{\text{tot}}(t)} - g, \quad (10)$$

where  $a(t)$  is acceleration at time  $t$ ,  $T(t)$  is the thrust [N],  $D(t)$  is the total drag [N], and  $g$  [m/s<sup>2</sup>] is gravitational acceleration. The following equation relates the thrust  $T(t)$  to the aperture

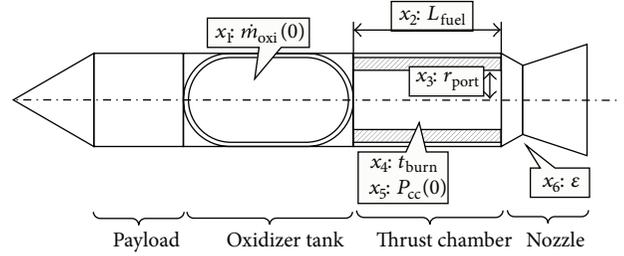


FIGURE 11: Schematic of the hybrid rocket.

ratio of the nozzle  $\epsilon$  and the pressure in the combustion chamber  $P_{\text{cc}}(0)$  [MPa]:

$$T(t) = \eta_T \left[ \lambda \dot{m}_{\text{prop}}(t) u_e + (P_e - P_a) A_e \right], \quad (11)$$

where  $\eta_T$  is the total thrust loss coefficient and  $\lambda$  is the momentum loss coefficient, embodying the effect of friction ( $<1$ ) at the nozzle exit.  $\dot{m}_{\text{prop}}(t)$  is the mass flow of propellant, and  $u_e$  and  $P_e$  denote the velocity and pressure at the nozzle exit, respectively.  $P_a$  denotes the atmospheric pressure at flight altitude, and  $A_e$  is the area of the nozzle exit.

The drag  $D(t)$  is decomposed into the pressure drag  $D_p(t)$  and the friction drag  $D_f(t)$ . The parameters  $\dot{m}_{\text{prop}}(t)$ ,  $D_p(t)$ , and  $D_f(t)$  are not described because of space limitations. For details on these parameters, the reader is referred to [13, 14].

The gross weight  $M_{\text{tot}}(t)$  is estimated by

$$M_{\text{tot}}(t) = M_{\text{en}}(t) + M_{\text{pay}}(t) + M_{\text{ex}}(t),$$

$$M_{\text{en}}(t) = M_{\text{oxi}}(t) + M_{\text{fuel}}(t) + M_{\text{res}}(t) + M_{\text{ch}}(t),$$

$$M_{\text{ex}}(t) = \frac{3}{2} M_{\text{en}}(t),$$

$$M_{\text{oxi}}(t) = \int_0^{t_{\text{burn}}} \dot{m}_{\text{oxi}}(t) dt, \quad (12)$$

$$M_{\text{fuel}}(t) = \int_0^{t_{\text{burn}}} \dot{m}_{\text{fuel}}(t) dt,$$

$$M_{\text{res}} = \rho V_{\text{res}},$$

$$M_{\text{ch}} = \rho V_{\text{ch}},$$

where  $M_{\text{pay}}$  and  $M_{\text{en}}$  are payload and engine weights, respectively.  $M_{\text{oxi}}$  is the total mass of the oxidizer, and  $M_{\text{fuel}}$  is the total fuel mass. The mass of the oxidizer tank, combustion chamber, and other equipment is denoted by  $M_{\text{res}}$ ,  $M_{\text{ch}}$ , and  $M_{\text{ex}}$ , respectively.  $V_{\text{res}}$  and  $V_{\text{ch}}$  are the integrated volumes of a material for the oxidizer tank and the combustion chamber, respectively.

Given (12), we define the multiobjective optimization problem of the hybrid rocket engine design as

$$\begin{aligned}
 & \text{maximize} && f_1 = H_{\max}, \\
 & \text{minimize} && f_2 = M_{\text{tot}}(0), \\
 & \text{subject to:} && 1.0 \leq \dot{m}_{\text{oxi}}(0) \leq 30.0, \\
 & && 1.0 \leq L_{\text{fuel}} \leq 10.0, \\
 & && 10.0 \leq r_{\text{port}}(0) \leq 200.0, \\
 & && 15.0 \leq t_{\text{burn}} \leq 35.0, \\
 & && 3.0 \leq P_{\text{cc}}(0) \leq 4.0, \\
 & && 5.0 \leq \varepsilon \leq 7.0.
 \end{aligned} \tag{13}$$

Here,  $\eta_T$  and  $M_{\text{pay}}$  are set to 1.0 and 50 [kg], respectively.  $\dot{m}_{\text{oxi}}(0)$ ,  $L_{\text{fuel}}$ ,  $P_{\text{ch}}$ , and  $P_{\text{res}}$  are assumed as constants. That is, this design problem seeks the most lightweight rocket that does not compromise the flight altitude.

**4.2. Multiobjective Optimization.** In this subsection, the dataset is the nondominated solution set of the hybrid rocket engine design problem. The solutions are derived by NSGA-II. The population size is set to 120. The analysis assumes a simulated binary crossover (SBX) with a crossover rate of 1.0 and a polynomial mutation with a mutation rate of  $1/(\text{chromosome length})$ . The decision variable vector of a single solution is represented as  $\mathbf{X} = [\dot{m}_{\text{oxi}}(0), L_{\text{fuel}}, r_{\text{port}}(0), t_{\text{burn}}, P_{\text{cc}}(0), \varepsilon]$  in NSGA-II. Each decision variable is binary coded with a length of 20 bits, giving a chromosome length of 120. A single run of NSGA-II is terminated after 188 generations. The objective function was calculated 22680 times, yielding 120 solutions. The crossover and mutation rate used here follows the practice in [6, 9]. Population size, the number of generations, and other genetic parameters are empirically chosen. The parameter study to obtain appropriate parameter setting is dismissed because it is not the focus of our study to improve optimization accuracy. The obtained nondominated solutions are plotted in Figure 12.

Intuitively, more fuel will achieve higher altitude; however, fuel increases the weight of the rocket. The weight-altitude tradeoff is evident in Figure 12 but is difficult to visualize in the six-dimensional decision space. For this reason, our design mode analysis is effective for analyzing the characteristics of high-dimensional decision spaces.

**4.3. Design Mode Analysis.** In this subsection, we extract the decision space characteristics of the nondominated solutions for the hybrid rocket engine design and derive an appropriate design strategy using our proposed design mode analysis (described in Section 3.4). Prior to running the PCA, we first preprocess the decision variables of the hybrid rocket design to normalize the mean and variance of them. Thus, PCA is performed based on the correlations matrix. When the range and scale of variables are different from each other, the dataset should be normalized. Otherwise, as in the case of MOKP, the covariance matrix is used for PCA to preserve variance without normalization of the dataset.

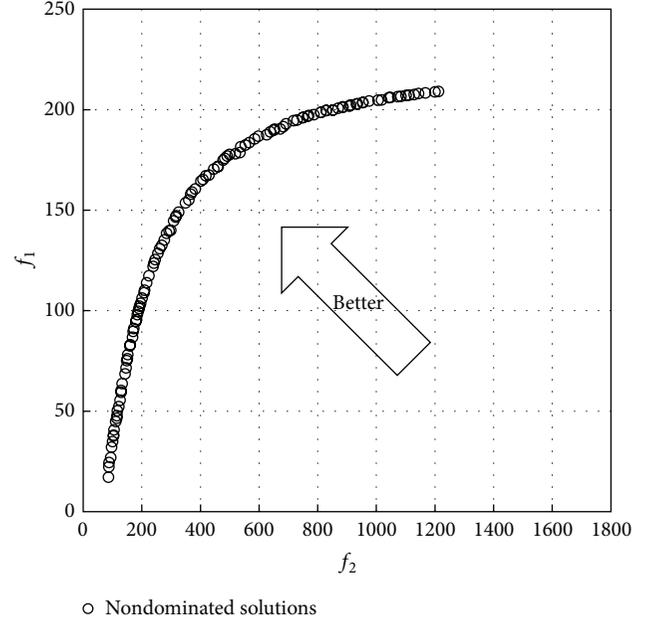


FIGURE 12: Nondominated solution set of hybrid rocket engine design problem.

The base of the approximated design is set to the mean vector of each cluster, where binary clustering is performed by the  $k$ -means method with cluster size 2. The distance metric in  $k$ -means clustering is the Euclidean distance. The error in the design approximated in the decision space is the sum of squares of the relative error (SSRE):

$$\sum_{k=1}^m \frac{(x'_k - x_{ik})^2}{x_{ik}^2}, \tag{14}$$

where  $x_{ik}$  is the  $k$ th decision variable of the  $i$ th design in the dataset and  $x'_k$  is the approximated design of  $x_{ik}$ . Moreover, the PCA is also performed on the correlation matrix. The SSRE is minimized by the optimization algorithm (i.e., sequential least squares programming) [15].

If some decision variables are integer values, we regard them as real-valued variables through the design mode analysis. When the designs based on the design modes are evaluated in the objective space, their variables, which are originally the integer values, should be rounded off to the closest whole number.

The pseudocode of the extended framework of design mode analysis is shown in Algorithm 2.

**4.4. Results and Discussion.** Figure 13 plots the history of the error in the decision space at each layer of the design approximation. Although, relative to the first layer, the accuracy worsens in the second and third layers, it gradually improves, as the layers are refined. This indicates that data clustering contributes to design mode classification and improves the accuracy of the approximated design. This trend is emphasized in the objective space.

- (1) Generate a design dataset  $C_1^{(1)} = (X^{(1)}, \dots, X^{(N)})$  of size  $N$ .
- (2) Scale the dataset such that all decision variables have zero mean and unit variance.
- (3) Initialize total approximation error  $E = \infty$ .
- (4) Set a threshold  $\eta$  for  $E$ .
- (5) Initialize layer counter  $i = 1$ .
- (6) **while**  $E > \eta$  **do**
- (7) Initialize the number of clusters in current layer  
 $H = 2^i - 1$ .
- (8) Initialize the counter of the clusters in new layer  $k = 1$ .
- (9) Initialize  $E = 0$ .
- (10) **for**  $j = 1$  to  $H$  **do**
- (11) Extract the design mode by applying PCA to  $C_j^{(i)}$ .
- (12) Calculate the component loading for each design mode.
- (13) Choose a base design  $C_j^{(i)}$ , or calculate a mean vector of  $C_j^{(i)}$ .
- (14) Choose  $p$  design modes so as to satisfy cumulative proportion of the variance  $P \geq 0.80$ .
- (15) Perform *Design Approximation* (mentioned above) for all the designs in  $C_j^{(i)}$ .
- (16) Add  $E_j^{(i)}$  (approximation error for  $C_j^{(i)}$ ) to the total error:  $E = E + E_j^{(i)}$ .
- (17) Divide the cluster  $C_j^{(i)}$  into two clusters  $C_k^{(i+1)}$  and  $C_{k+1}^{(i+1)}$  by using data clustering.
- (18)  $k = k + 2$
- (19) **end for**
- (20)  $i = i + 1$ .
- (21) **end while**

ALGORITHM 2: Design mode analysis focusing on granularity.

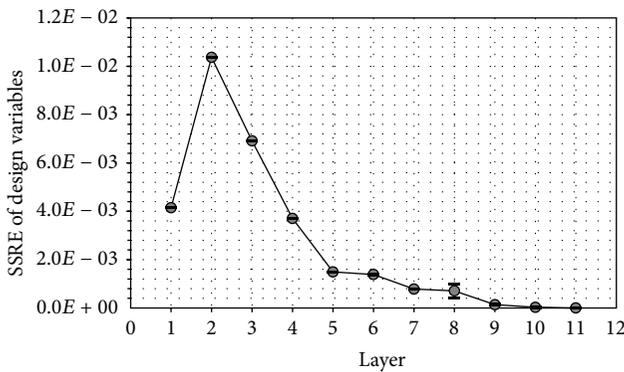


FIGURE 13: History of the accuracy of the approximated decision variables.

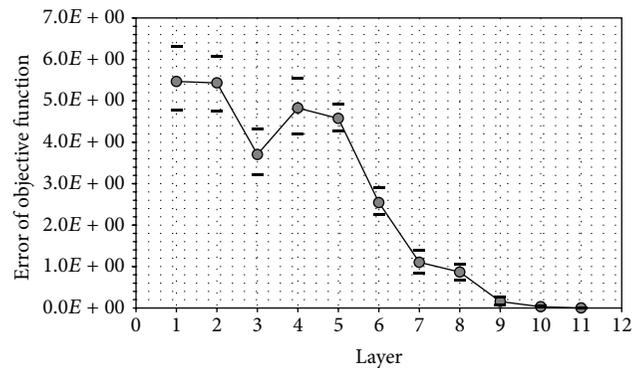


FIGURE 14: History of the accuracy of the approximated objective function values.

Figure 14 plots the history of the error in the objective space at each layer, evaluated as the average Euclidean distance between each real and approximated design.

Figure 15 plots the histories of the average number of clusters (circle-plotted curve) and the average cluster size (average number of designs within each cluster, indicated by the triangle-plotted curve). While the analysis can be continued until the number of clusters equals the dataset

size, such refinement is nonsensical because PCA cannot be performed on a single datum. Instead, we stipulate that our proposed analysis be continued while the dataset size is larger than the dimension of the decision variables. In this case, since the decision space is six-dimensional, the analysis is meaningful up to the 5th layer.

Figure 16 indicates the original and approximated designs of each cluster at each layer. In these plots, “Cx-y” denotes

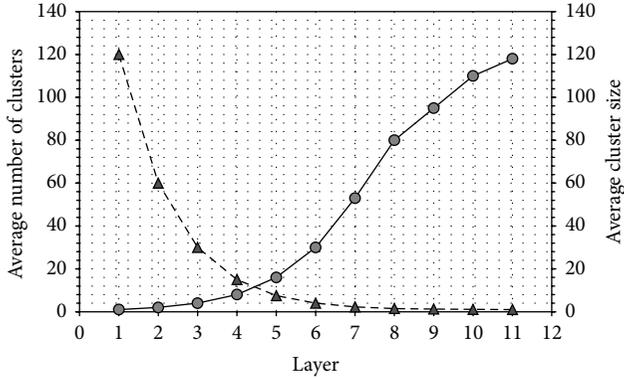


FIGURE 15: History of the number of clusters in each layer.

the  $y$ th cluster at the  $x$ th layer. Notably, possibly because these designs have distinguishable characteristics, around C2-2 in the 2nd layer are almost exactly retained in passing via the 3rd to the 4th layer.

For a detailed characterization of each cluster, we consider the component loadings of each cluster. For illustrative purposes, we investigate the mode 1 component loadings only, although each cluster yielded multiple design modes. The mode 1 component loadings of each cluster at each layer are plotted in Figure 17. The component loading, denoted by  $r_i$ , represents the correlation between the design mode and decision variable. If  $r_i$  is large, the design mode is highly correlated with decision variable  $x_i$ .

$r_1$  and  $r_2$  are excessively high in the first layer. Since  $x_1$  and  $x_2$  denote the initial mass flow of the oxidizer  $\dot{m}_{\text{prop}}(t)$  and the fuel length  $L_{\text{fuel}}$ , respectively, these two parameters are expected to dominate in this problem. To obtain variable designs on nondominated solution sets, we can alter both parameters along the first design mode. Here, the first design mode is the design mode corresponding to the eigenvector with the maximum eigenvalue. Notably, both parameters should be aligned in the same direction because their component loadings have the same sign. This yields the design mode obtained in the first layer. When we create a new design, we first choose the base design and then modify its decision variables along the direction indicated by each design mode (the eigenvector obtained by PCA). The component loadings represent the correlations between the decision variables and each design mode. The sign of the component loading indicates the direction of each decision variable on the axis of the design mode. In the case of the first layer,  $r_1$ ,  $r_2$ ,  $r_5$ , and  $r_6$  are positive values, but  $r_3$  and  $r_4$  are negative values. This indicates that if the decision variables  $x_1$ ,  $x_2$ ,  $x_5$ , and  $x_6$  are changed to the positive direction,  $x_3$  and  $x_4$  should be moved into the negative direction in the first design mode.

In the second layer, each cluster appears to yield different design modes. However, the distribution from  $r_3$  to  $r_6$  is almost identical between the two clusters, and since the component loadings of  $r_1$  and  $r_2$  in the clusters are merely opposite in sign, we can regard the design modes in the clusters as unchanged from the first layer. Thus, when the dataset is divided by  $x_1$  and  $x_2$ , the characteristics of the

resulting clusters are almost identical, suggesting that binary clustering is uninformative at the second layer. This explains the deterioration of the accuracy for design approximation in the second layer.

In the third layer, C3-2 and C3-4 are negatively correlated with  $x_1$  and  $x_2$  but differ in their correlations with  $x_3$  and  $x_4$ . For example, C3-4 yields a new design mode that reverses the sign of  $x_1$ ,  $x_2$ ,  $x_3$ , and  $x_4$  from positive to negative. While the first and second layers only revealed that  $x_1$  and  $x_2$  are dominant, different design modes for each cluster are revealed in the third layer. Moreover, the component loadings of C3-3 and C2-2 appear to be very similar, although the characteristics of C3-3 are expected to dominate over those of C2-2.

In layers 1–3, the component loadings of the higher modes (relative to  $r_1$  and  $r_2$ ) are comparatively low. However, different design modes can be extracted by increasing the granularity. In cluster C4-4 (layer 4), the component loading of  $r_3$  becomes relatively high, while  $r_1$  and  $r_2$  change in sign ( $x_3$  is an initial port radius  $r_{\text{port}}(0)$ ). Thus, C4-4 may provide a design strategy that the decision parameter  $x_1$  can be explored in the negative direction and at the same time  $x_2$  and  $x_3$  should be changed to the positive direction. We conclude that unique design modes are obtained at the 4th layer.

From the experiments in this section, we infer the following:

- (i) the deeper the layer, the better the approximation accuracy (observed in Figures 13 and 14);
- (ii) the design mode is characterized by the component loading distributions of each mode;
- (iii) in the hybrid rocket design problem, the initial mass flow of oxidizer and the fuel length dominate the Pareto set;
- (iv) different design modes are revealed as the granularity is increased; in cluster C4-4, the initial port radius of port exhibits a higher component loading than in other clusters.

A remarkable outcome of this study is that new designs with the same characteristics as a specified design mode are obtained. The design mode provides its own design strategy. The characteristics of each design mode are easily understood by investigating their component loadings. The proposed framework is especially useful when the design problem has a huge number of decision variables because it isolates the important parameters and specifies how their values should be altered.

A priority for our future study is to improve the data clustering process. The current framework adopts binary clustering, which does not always perform to the required standard. To realize more effective clustering, we require a scheme that automatically determines an appropriate number of clusters. Ineffective clustering generates many clusters with identical characteristics. The distance metric of the data clustering should also be reviewed. For example, the Mahalanobis distance, which is based on correlations in the dataset, may improve the data classification.

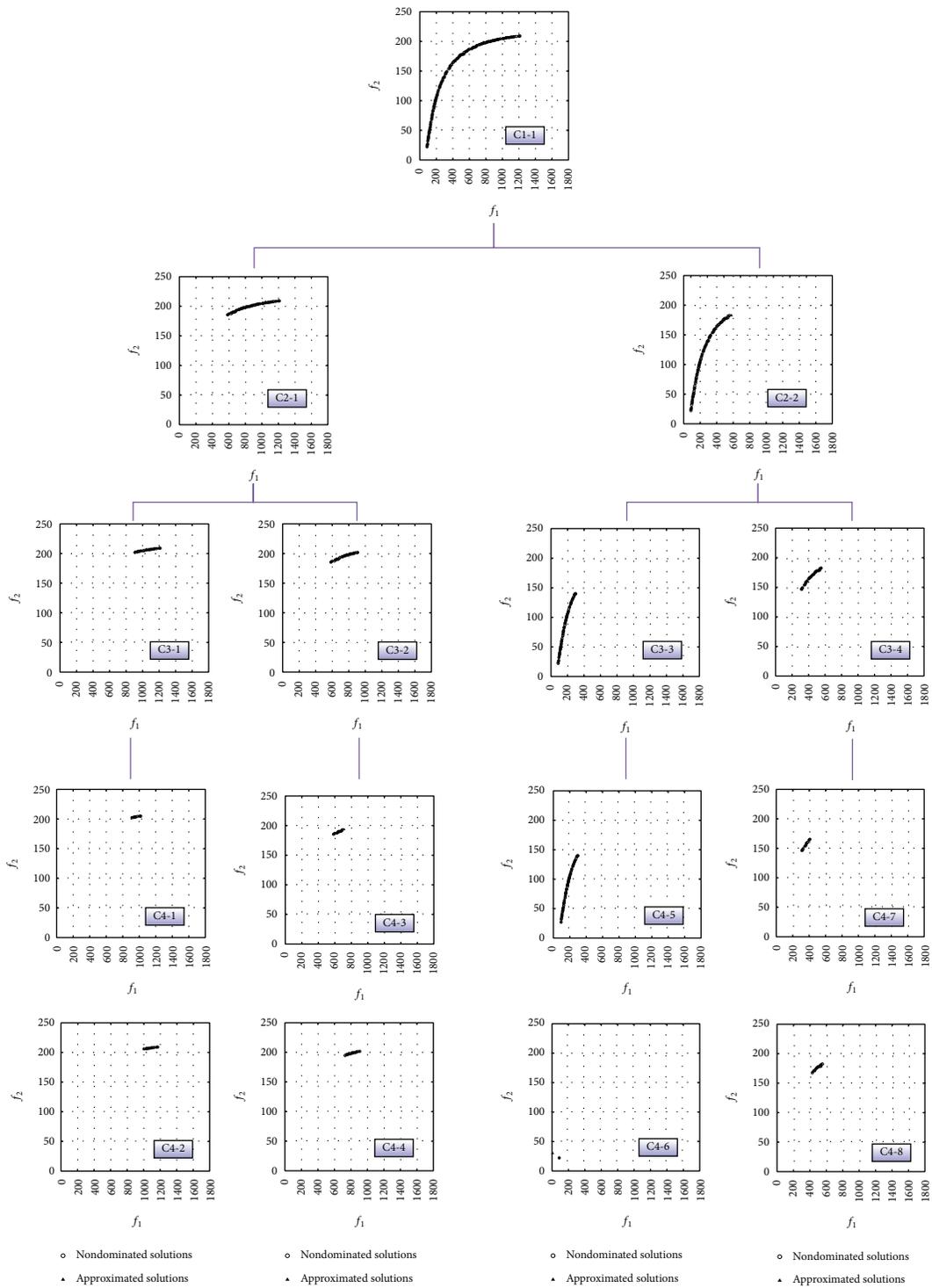


FIGURE 16: Approximated solutions of each cluster in each layer.

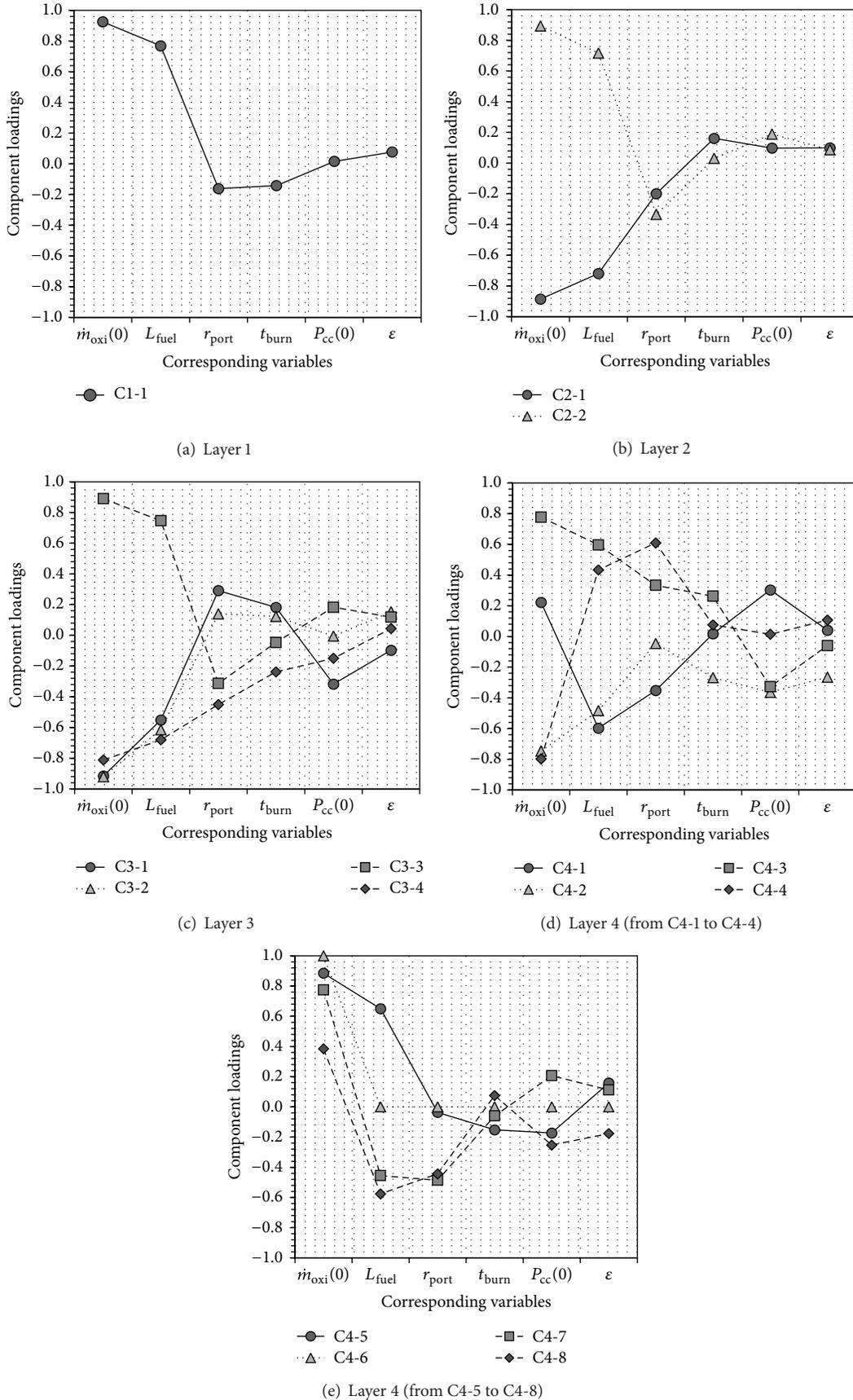


FIGURE 17: Mode 1 component loadings of each cluster at each layer.

## 5. Conclusions

We have proposed a design mode analysis of Pareto solution sets that supports human decision making. The design mode of the Pareto solution set was extracted by PCA. We demonstrated that any design in the Pareto set can be represented by a linear combination of the eigenvectors of the base design. From this finding, we developed a hierarchical framework for design mode analysis, in which the granularity of the extracted design modes determines the accuracy of the approximated design. The effectiveness of the proposed method was tested on the conceptual design problem of the hybrid rocket engine. We found that the extracted design modes depended on the granularity of the analysis. The proposed method will support human decision making in engineering design problems.

## Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

## References

- [1] P. P. Bonissone, R. Subbu, and J. Lizzi, "Multicriteria decision making (MCDM): a framework for research and applications," *IEEE Computational Intelligence Magazine*, vol. 4, no. 3, pp. 48–61, 2009.
- [2] S. Morizawa, K. Shimoyama, S. Obayashi, K. Funamoto, and T. Hayase, "Implementation of visual data mining for unsteady blood flow field in an aortic aneurysm," *Journal of Visualization*, vol. 14, no. 4, pp. 393–398, 2011.
- [3] S. Obayashi, S. Jeong, K. Chiba, and H. Morino, "Multi-objective design exploration and its application to regional-jet wing design," *Transactions of the Japan Society for Aeronautical and Space Sciences*, vol. 50, no. 167, pp. 1–8, 2007.
- [4] A. Oyama, T. Nonomura, and K. Fujii, "Data mining of non-dominated solutions using proper orthogonal decomposition," in *Proceedings of the 11th Annual Genetic and Evolutionary Computation Conference (GECCO '09)*, pp. 1935–1936, Montréal, Canada, July 2009.
- [5] A. Oyama, T. Nonomura, and K. Fujii, "Data mining of Pareto-optimal transonic airfoil shapes using proper orthogonal decomposition," *Journal of Aircraft*, vol. 47, no. 5, pp. 1756–1762, 2010.
- [6] 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.
- [7] A. Zhou, B.-Y. Qu, H. Li, S.-Z. Zhao, P. N. Suganthan, and Q. Zhangd, "Multiobjective evolutionary algorithms: a survey of the state of the art," *Swarm and Evolutionary Computation*, vol. 1, no. 1, pp. 32–49, 2011.
- [8] A. Arias-Montaña, C. A. C. Coello, and E. Mezura-Montes, "Multiobjective evolutionary algorithms in aeronautical and aerospace engineering," *IEEE Transactions on Evolutionary Computation*, vol. 16, no. 5, pp. 662–694, 2012.
- [9] 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.
- [10] The MOCO Numerical Instances Library, <http://xgandibleux.free.fr/MOCOLib/index.html>.
- [11] S. Watanabe, S. Ohe, and R. Minato, "Development of a multi-granularity design support system using hierarchical clustering for non-dominated solutions," in *Proceedings of the World Automation Congress (WAC '10)*, pp. 1–6, September 2010.
- [12] Design optimization problems in space science, [http://flab.eng.isas.jaxa.jp/member/oyama/realproblems\\_j.html](http://flab.eng.isas.jaxa.jp/member/oyama/realproblems_j.html).
- [13] Y. Kosugi, A. Oyama, M. Kanazaki, and K. Fujii, *Multidisciplinary and Multi-Objective Design Exploration Methodology for Conceptual Design of a Hybrid Rocket*. Infotech@Aerospace, American Institute of Aeronautics and Astronautics, 2011.
- [14] K. Chiba, "Performance comparison of evolutionary algorithms applied to hybrid rocket problem," in *Proceedings of the Joint 6th International Conference on Soft Computing and Intelligent Systems (SCIS '12) and 13th International Symposium on Advanced Intelligence Systems (ISIS '12)*, pp. 1673–1678, November 2012.
- [15] D. Kraft, "Algorithm 733: TOMP—fortran modules for optimal control calculations," *ACM Transactions on Mathematical Software*, vol. 20, no. 3, pp. 262–281, 1994.